



Le Lu · Yefeng Zheng
Gustavo Carneiro · Lin Yang *Editors*

Deep Learning and Convolutional Neural Networks for Medical Image Computing

Precision Medicine, High Performance
and Large-Scale Datasets

Advances in Computer Vision and Pattern Recognition

Founding editor

Sameer Singh, Rail Vision, Castle Donington, UK

Series editor

Sing Bing Kang, Microsoft Research, Redmond, WA, USA

Advisory Board

Horst Bischof, Graz University of Technology, Austria

Richard Bowden, University of Surrey, Guildford, UK

Sven Dickinson, University of Toronto, ON, Canada

Jiaya Jia, The Chinese University of Hong Kong, Hong Kong

Kyoung Mu Lee, Seoul National University, South Korea

Yoichi Sato, The University of Tokyo, Japan

Bernt Schiele, Max Planck Institute for Computer Science, Saarbrücken, Germany

Stan Sclaroff, Boston University, MA, USA

More information about this series at <http://www.springer.com/series/4205>

Le Lu · Yefeng Zheng · Gustavo Carneiro
Lin Yang
Editors

Deep Learning and Convolutional Neural Networks for Medical Image Computing

Precision Medicine, High Performance
and Large-Scale Datasets



Springer

Editors

Le Lu

National Institutes of Health Clinical Center
Bethesda, MD
USA

Gustavo Carneiro

University of Adelaide
Adelaide, SA
Australia

Yefeng Zheng

Siemens Healthcare Technology Center
Princeton, NJ
USA

Lin Yang

University of Florida
Gainesville, FL
USA

ISSN 2191-6586

ISSN 2191-6594 (electronic)

Advances in Computer Vision and Pattern Recognition

ISBN 978-3-319-42998-4

ISBN 978-3-319-42999-1 (eBook)

DOI 10.1007/978-3-319-42999-1

Library of Congress Control Number: 2017936345

© Springer International Publishing Switzerland 2017

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature

The registered company is Springer International Publishing AG

The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This book was partially motivated by the recent rapid progress on deep convolutional and recurrent neural network models and the abundance of important applications in computer vision, where quantitative performance has significantly improved in object recognition, detection, and automatic image caption. However publicly available image database with generally well-annotated image or object labels (but with some labeling noise), such as ImageNet (1.2 million images), PASCAL Visual Object Classes (VOC) Dataset and Challenge (VOC) ($\sim 12,000$ images), and Microsoft Common Objects in Context (COCO; $\sim 300,000$ images), have been the essential resource to fuel the learning process of the deep neural networks that work well in practical and challenging scenarios.

These types of large-scale annotated image datasets are, unfortunately, not available in medical image analysis tasks yet, even though improving medical imaging applications through deep neural network models and imaging data is highly valuable. This is partly explained by the fact that labels or annotations for medical image database are much harder or expensive to obtain. The general practice of collecting labels for ImageNet, as for example using Google image search engine for pre-selection, followed by manual label refinement through crowd-sourcing (e.g., Amazon Mechanical Turk), is largely nonfeasible due to the formidable difficulties of medical annotation tasks for those who are not clinically trained.

Indeed, employing deep neural networks, especially convolutional neural network models, requires a large amount of annotated training instances. This concern was reflected in that only four papers (out of 250 total publications) in MICCAI 2014, the 17th International Conference on Medical Image Computing and Computer Assisted Intervention, were based on deep learning models while at least 20% of papers at IEEE Conference on CVPR 2014 were related to deep neural networks. Even after that, this situation has drastically changed. We have had nearly 10% of the publications (23 papers) in MICCAI 2015 that are built upon deep neural network models for a variety of medical imaging problems: fetal ultrasound standard plane detection, vertebrae localization and identification, multi-view mammogram analysis, mass segmentation, glaucoma detection, nucleus localization in microscopy images, lymph node detection and segmentation, organ segmentation

in CT/MRI scans, coronary calcium scoring in cardiac CT angiography, etc. We certainly predict this uprising trend will continue for MICCAI 2016.

To answer the question of how to learn powerful and effective deep neural networks with often sub-hundred of patient scans (where significant quantitative performance gains have been reported), many chapters in this book will precisely address this promising trend by describing detailed technical contents on data preparation, network designs, and evaluation strategies. Moreover, what can be learned from this early success and how to move forward rapidly are the other two main topics to be discussed in details for this book.

Overview and Goals

Deep learning, in particular Convolutional Neural Networks (CNN), is a validated image representation and classification technique for medical image analysis and applications. We have observed many encouraging work that report new and newer state-of-the-art performance on quite challenging problems in this domain. The main reason behind this stream of work, we believe, is that effective task-dependent image features can be directly or intrinsically learned through the hierarchy of convolutional kernels inside CNN. Hand-crafted image feature engineering research was a relatively weak subtopic in medical image analysis, compared to the extensive evaluation and studies on image features for computer vision. This sometimes limits the generality and applicability of well-tested natural image feature descriptors, such as SIFT (scale-invariant feature transform), HOG (histogram of oriented gradients) and others into medical imaging tasks.

On the other hand, CNN models have been proved to have much higher modeling capacity, compared to the previous image recognition mainstream pipelines, e.g., HAAR, SIFT, HOG image features followed by spatial feature encoding, then random forest or support vector classifiers. Given millions of parameters to fit during model training (much more than previous pipelines), CNN representation empowers and enables computerized image recognition models, with a good possibility to be able to handle more challenging imaging problems. The primary risk is overfitting since model capacity is generally high in deep learning but often very limited datasets are available (that are with good quality of labels to facilitate supervised training). The core topics of this book are represented by examples on how to address this task-critical overfitting issue with deep learning model selection, dataset resampling and balancing, and the proper quantitative evaluation protocols or setups.

Furthermore, with deep neural networks (especially CNNs) as *De facto* building blocks for medical imaging applications (just as previous waves of Boosting, Random Forest), we would argue that it is of course important to use them to improve existing problems, which has been widely studied before, but more critically, it is the time to consider exploiting new problems and new experimental, clinical protocols that will foster the development of preventative and precision

medicine tools in imaging to impact modern clinical practices. Without loss of generality, we give the following three examples: *Holistic CT slice based interstitial lung disease (ILD) prediction* via deep multi-label regression and unordered pooling (significantly improving the current status of the mainstream image patch based ILD classification approaches with several built-in prerequisites that nevertheless prevents clinically desirable diagnosis protocols for ILD pre-screening from ultra-low dose CT scans to be a reality); *Precise deep organ/tumor segmentation based volumetric measurements as new imaging bio-markers* (remarkably enabling to provide more precise imaging measurement information to better assist physicians than the current popular RECIST metric, for high-fidelity patient screening, tumor management and patient-adaptive therapy); and *Unsupervised category discovery and joint text-image deep mining using large-scale radiology image database* (opening the door to compute, extract, and mine meaningful clinical imaging semantics via modern hospitals' PACS/RIS database systems that could include millions of patient imaging and text report cases, overcoming the limitation of lack of strong annotations).

Lastly, from our perspective, this is just the beginning of embracing and employing new deep neural network models and representations for many medical image analysis and medical imaging applications. We hope this book will help to get you more prepared and ready to exploit old and new problems. Happy reading and learning!

Organization and Features

This book covers a range of topics from reviews of the recent state-of-the-art progresses, to deep learning for semantic object detection, segmentation and large-scale radiology database mining. In the following, we give a brief overview of the contents of the book.

Chapter 1 describes a brief review of nearly 20 years of research by Dr. Ronald M. Summers (MD/Ph.D.) in medical imaging based computer-aided diagnosis (CAD) where he won the presidential early career award for scientists and engineers in 2000 and his personal take and insights on the recent development of deep learning techniques for medical image interpretation problems?

Chapter 2 lists a relatively comprehensive review of the recent methodological progress and related literature of deep learning for medical image analysis, in the topics of abdominal, chest and cardiology imaging, histopathology cell imaging, and chest X-ray and mammography.

Chapters 3 to 10 cover all various topics using deep learning for object or landmark detection tasks in 2D and 3D medical imaging. Particularly, we present a random view resampling and integration approach for three CAD problems (Chap. 3); 3D volumetric deep neural networks for efficient and robust landmark detection (Chap. 4); a restricted views based pulmonary embolism detection (Chap. 5); a new method on cell detection (Chap. 6); tumor cell anaplasia and multi-nucleation

detection (Chap. 7) in microscopy images; Predicting interstitial lung diseases and segmentation label propagation (Chap. 8); an in-depth study on CNN architectures, dataset characteristics and transfer learning for CAD problems (Chap. 9); followed by a computationally scalable method of accelerated cell detection with sparse convolution kernels in histopathology images (Chap. 10).

Chapters 11 to 16 discuss several representative works in semantic segmentation using deep learning principles in medical imaging. Specifically, this book describes automatic carotid intima-media thickness analysis (Chap. 11); deep segmentation via distance regularized level set and deep-structured inference (Chap. 12); structured prediction for segmenting masses in mammograms (Chap. 13); pathological kidney segmentation in CT via local versus global image context (Chap. 14); skeletal muscle cell segmentation in Microscopy (Chap. 15); and a bottom-up approach for deep pancreas segmentation in contrasted CT images (Chap. 16).

Chapter 17 discusses a novel work on interleaved text/image deep mining on a large-scale radiology image database for automated image interpretation which was not technically feasible in the pre-deep learning era. Finally, we would like to point out the way we organized this book in roughly three blocks of detection, segmentation, and text-image joint learning aligned with the status how computer vision is progressing with deep learning. These three blocks of research topics are probably essential for imaging understanding and interpretation of all imaging problems.

Target Audience

The intended reader of this book is a professional or a graduate student who is able to apply computer science and math principles into problem solving practices. It may be necessary to have some level of familiarity with a number of more advanced subjects: image formation, processing and understanding, computer vision, machine learning, and statistical learning.

Bethesda, USA

Princeton, USA

Adelaide, Australia

Gainesville, USA

Le Lu

Yefeng Zheng

Gustavo Carneiro

Lin Yang

Acknowledgements

We thank the Intramural Research Program of the National Institutes of Health (NIH) Clinical Center, the Extramural Funding office at National Institute of Biomedical Imaging and Bioengineering, National Institute of Arthritis Musculoskeletal and Skin Disease, NIH, Natural Science Foundation, Australian Centre of Excellence for Robotic Vision, Siemens Healthcare, a grant from the KRIBB Research Initiative Program (Korean Visiting Scientist Training Award), Korea Research Institute of Bioscience and Biotechnology, Republic of Korea. Some study utilized the high-performance computational capabilities of the Biowulf Linux cluster at the National Institutes of Health, Bethesda, MD (<http://biowulf.nih.gov>), and we thank NVIDIA for the GPU donation of K40s. The study in this book is funded, in part, by NIH 5R01AR065479-02 for LY.

Contents

Part I Review

- 1 Deep Learning and Computer-Aided Diagnosis
for Medical Image Processing: A Personal Perspective 3
Ronald M. Summers
- 2 Review of Deep Learning Methods in Mammography,
Cardiovascular, and Microscopy Image Analysis 11
Gustavo Carneiro, Yefeng Zheng, Fuyong Xing and Lin Yang

Part II Detection and Localization

- 3 Efficient False Positive Reduction in Computer-Aided
Detection Using Convolutional Neural Networks
and Random View Aggregation 35
Holger R. Roth, Le Lu, Jiamin Liu, Jianhua Yao, Ari Seff,
Kevin Cherry, Lauren Kim and Ronald M. Summers
- 4 Robust Landmark Detection in Volumetric Data
with Efficient 3D Deep Learning 49
Yefeng Zheng, David Liu, Bogdan Georgescu, Hien Nguyen
and Dorin Comaniciu
- 5 A Novel Cell Detection Method Using Deep Convolutional
Neural Network and Maximum-Weight Independent Set 63
Fujun Liu and Lin Yang
- 6 Deep Learning for Histopathological Image Analysis:
Towards Computerized Diagnosis on Cancers 73
Jun Xu, Chao Zhou, Bing Lang and Qingshan Liu

7	Interstitial Lung Diseases via Deep Convolutional Neural Networks: Segmentation Label Propagation, Unordered Pooling and Cross-Dataset Learning	97
	Mingchen Gao, Ziyue Xu and Daniel J. Mollura	
8	Three Aspects on Using Convolutional Neural Networks for Computer-Aided Detection in Medical Imaging	113
	Hoo-Chang Shin, Holger R. Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura and Ronald M. Summers	
9	Cell Detection with Deep Learning Accelerated by Sparse Kernel	137
	Junzhou Huang and Zheng Xu	
10	Fully Convolutional Networks in Medical Imaging: Applications to Image Enhancement and Recognition	159
	Christian F. Baumgartner, Ozan Oktay and Daniel Rueckert	
11	On the Necessity of Fine-Tuned Convolutional Neural Networks for Medical Imaging	181
	Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway and Jianming Liang	

Part III Segmentation

12	Fully Automated Segmentation Using Distance Regularised Level Set and Deep-Structured Learning and Inference	197
	Tuan Anh Ngo and Gustavo Carneiro	
13	Combining Deep Learning and Structured Prediction for Segmenting Masses in Mammograms	225
	Neeraj Dhungel, Gustavo Carneiro and Andrew P. Bradley	
14	Deep Learning Based Automatic Segmentation of Pathological Kidney in CT: Local Versus Global Image Context	241
	Yefeng Zheng, David Liu, Bogdan Georgescu, Daguang Xu and Dorin Comaniciu	
15	Robust Cell Detection and Segmentation in Histopathological Images Using Sparse Reconstruction and Stacked Denoising Autoencoders	257
	Hai Su, Fuyong Xing, Xiangfei Kong, Yuanpu Xie, Shaoting Zhang and Lin Yang	

Contents	xiii
16 Automatic Pancreas Segmentation Using Coarse-to-Fine Superpixel Labeling	279
Amal Farag, Le Lu, Holger R. Roth, Jiamin Liu, Evrim Turkbey and Ronald M. Summers	
Part IV Big Dataset and Text-Image Deep Mining	
17 Interleaved Text/Image Deep Mining on a Large-Scale Radiology Image Database	305
Hoo-Chang Shin, Le Lu, Lauren Kim, Ari Seff, Jianhua Yao and Ronald Summers	
Author Index	323
Subject Index	325

Part I

Review

Chapter 1

Deep Learning and Computer-Aided Diagnosis for Medical Image Processing: A Personal Perspective

Ronald M. Summers

Abstract These are exciting times for medical image processing. Innovations in deep learning and the increasing availability of large annotated medical image datasets are leading to dramatic advances in automated understanding of medical images. From this perspective, I give a personal view of how computer-aided diagnosis of medical images has evolved and how the latest advances are leading to dramatic improvements today. I discuss the impact of deep learning on automated disease detection and organ and lesion segmentation, with particular attention to applications in diagnostic radiology. I provide some examples of how time-intensive and expensive manual annotation of huge medical image datasets by experts can be sidestepped by using weakly supervised learning from routine clinically generated medical reports. Finally, I identify the remaining knowledge gaps that must be overcome to achieve clinician-level performance of automated medical image processing systems.

Computer-aided diagnosis (CAD) in medical imaging has flourished over the past several decades. New advances in computer software and hardware and improved quality of images from scanners have enabled this progress. The main motivations for CAD have been to reduce error and to enable more efficient measurement and interpretation of images. From this perspective, I will describe how deep learning has led to radical changes in how CAD research is conducted and in how well it performs. For brevity, I will include automated disease detection and image processing under the rubric of CAD.

Financial Disclosure The author receives patent royalties from iCAD Medical.

Disclaimer No NIH endorsement of any product or company mentioned in this manuscript should be inferred. The opinions expressed herein are the author's and do not necessarily represent those of NIH.

R.M. Summers (✉)

Imaging Biomarkers and Computer-Aided Diagnosis Laboratory,
Radiology and Imaging Sciences, National Institutes of Health Clinical Center,
Bldg. 10, Room 1C224D MSC 1182, Bethesda, MD 20892-1182, USA
e-mail: rms@nih.gov
URL: http://www.cc.nih.gov/about/SeniorStaff/ronald_summers.html

For medical imaging, CAD has focused predominantly on radiology, cardiology, and pathology. Examples in radiology include the automated detection of microcalcifications and masses on mammography, lung nodules on chest X-rays and CT scans, and colonic polyps on CT colonography [1]. In cardiology, examples include CAD for echocardiography and angiography [2–4]. In digital pathology, examples include detection of cellular components such as nuclei and cells and diseases such as breast, cervical, and prostate cancers [5].

Despite significant research progress in all of these applications, the translation of CAD systems from the bench to the bedside has been difficult and protracted. What have been the impediments to realizing the full potential of medical imaging CAD? There have been a number of difficulties. First, the development of CAD systems is time-consuming and labor-intensive. Researchers must aggregate cases with proven pathology. This is no easy task as the best proof involves surgical excision and histopathologic analysis. It is not always possible to obtain such reference standards of truth. Once the medical data and reference standard have been collected, the data must be annotated. For example, for radiology images, the precise location and extent of the abnormality must be determined by a trained expert. Such hand annotation by an expert is time-consuming and expensive. The best annotations come from multiple trained observers annotating the same abnormality so that a probabilistic assessment of confidence of lesion location can be obtained. Consequently, many CAD systems involve only on the order of tens or hundreds of proven cases. Better evidence usually comes from large datasets, on the order of thousands or more, unattainable for all but the most well-funded studies. For example, the NLST study involved over 53,000 patients and cost over \$250 million [6]. Few studies can achieve these numbers.

These difficulties with data collection have severely hampered CAD research. They have led to the contraction of most CAD research into only a few major problem domains, such as lung nodule detection and mammographic mass detection. But clinicians need to address numerous imaging problems while interpreting medical images. For example, to properly diagnose a chest CT scan, a physician needs to inspect dozens of structures and must be aware of hundreds of potential abnormalities including lesions and normal variants [7]. Most of these numerous imaging problems have been ignored or understudied.

Another difficulty has been the time-consuming task of handcrafting of algorithms for CAD systems. Until recently, it was necessary to develop mathematical algorithms specifically tailored to a particular problem. For example, when I started to develop a CAD system for virtual bronchoscopy in 1997, there were no prior examples on which to build [8]. My lab had to develop shape-based features to distinguish airway polyps from normal airways [9, 10]. When we extended the software to find polyps in the colon on CT colonography, it took about five years to perfect the software and annotate the images to the point where the software could undergo a robust evaluation on over 1000 cases [11]. It took another five years for translation from the bench to the bedside [12]. Other groups found it similarly time-consuming to develop, validate, and refine CAD systems for colonic polyp detection.

Most of our CADs used machine learning classifiers such as committees of support vector machines, which appeared to be superior to other approaches including

conventional neural networks, decision trees and linear discriminants [13–18]. About two years ago, I heard about “deep learning”, a new up-and-coming technology for machine learning [19]. Deep learning was the name given to an improved type of neural network having more layers to permit higher levels of abstraction. Deep learning was finding success at solving hard problems such as recognizing objects in real world images [20, 21]. An aspect of deep learning that caught my attention was its ability to learn the features from the training data. I had long been unhappy with CAD systems that required hand-chosen parameters and hand-crafted features designed for a particular application. I realized early on that such hand-tuning would not be reliable when the CAD software was applied to new data. In addition, the hand-tuning was very time-consuming and fragile. One could easily choose parameters that worked well on one dataset but would fail dramatically on a new dataset. Deep learning had the appeal of avoiding such hand-tuning.

About this time, Dr. Le Lu joined my group. An expert in computer vision, Le brought the passion and knowledge required to apply deep learning to the challenging problems we were investigating. As we learned about the current state of research on deep learning, I was surprised to find that other investigators had used convolutional neural networks, one type of deep learning, in the past [22, 23]. But there seemed to be something different about the most recent crop of deep learning algorithms. They routinely used GPU processing to accelerate training by as much as a factor of 40-fold. They also used multiple convolution layers and multiple data reduction layers. Of great importance, the authors of these new deep learning networks made their software publicly available on the Internet. Soon, an entire zoo of deep learning architectures was available online to download and try, using a variety of different computer programming platforms and languages [24].

In 2013, I was fortunate to attract Dr. Holger Roth to be a postdoctoral fellow in my group. Holger had received his graduate training under David Hawkes at University College London. With this outstanding foundation, Holger was poised to enter the new deep learning field and take it by storm. He published early papers on pancreas and lymph node detection and segmentation, two very challenging problems in radiology image processing. Not only could he develop the software rapidly by leveraging the online resources, but the deep learning networks were able to train on relatively small datasets of under 100 cases to attain state-of-the-art results. With this encouragement, we applied deep learning to many other applications and shifted much of our lab’s focus to deep learning.

Holger and two other postdoctoral fellows in my group, Drs. Hoo-Chang Shin and Xiaosong Wang, have applied deep learning to other challenges. For example, Hoo-Chang showed how deep learning could combine information from radiology reports and their linked images to train the computer to “read” CT and MRI scans and chest X-rays [25, 26]. Xiaosong found a way to automatically create semantic labels for medical images using deep learning [27]. Holger showed how deep learning could markedly improve the sensitivity (by 13–34%) of CAD for a variety of applications, including colonic polyp, spine sclerotic metastasis and lymph node detection [28]. Hoo-Chang showed how the use of different convolutional neural network (CNN) architectures and dataset sizes affected CAD performance [29].

While the use of deep learning by the medical image processing community trailed that of the computer vision community by a couple of years, the uptake of deep learning in medical research has been nothing short of amazing. While in 2014 only five papers by my count used deep learning, in 2015, at least 20 papers used it at the important MICCAI annual meeting, not including the satellite workshops. When my colleagues Hayit Greenspan and Bram van Ginneken sent out a request for deep learning papers in medical imaging for a special issue of the journal IEEE Transactions on Medical Imaging, we anticipated about 15 papers but received 50 submissions in just a few months [30]. These were full journal papers showing the results of detailed experiments and comparisons with existing methods. Clearly, the time was ripe for deep learning in medical image processing.

Interest and advances in deep learning are still growing rapidly in both the computer vision and the medical image processing communities. In the computer vision community, there is a developing sense that the large ImageNet archive used for training the first generation of deep learning systems is nearing exhaustion and new, larger next-generation image databases are needed to take deep learning to the next level, e.g., the Visual Genome project [31]. New CNN architectures are constantly being proposed by large companies and academic research groups that claim improved performance on datasets such as ImageNet. Deep networks with hundreds of layers have been proposed [32]. Researchers are leveraging specialized network elements such as memory and spatial locality to improve performance [33, 34].

In the medical image processing community, there is a great need for false positive (FP) reduction for automated detection tasks. Early work indicates that substantial FP reduction or improved image retrieval is possible with off-the-shelf deep learning networks [28, 35, 36]. Yet even these reductions do not reach the specificity obtained by practicing clinicians. Clearly, further work is required, not just on improved machine learning, but also on the more traditional feature engineering.

Another focus area in the medical image processing community is improved segmentation of organs and lesions. Here again there is evidence that deep learning improves segmentation accuracy and reduces the need for hand-crafted features (Fig. 1.1) [37, 38]. However, segmentation failures still occur and require manual correction. It is likely that for the foreseeable future, semi-automated segmentation with occasional manual correction will be the norm. Consequently, proper user interface design for efficient 3D manual correction will be advantageous. Large datasets will be required to provide representation of more outlier cases. A number of annotated datasets of medical images are available for download [39–42]. The use of crowdsourcing to annotate large medical datasets requires further investigation [43–45].

It is clear that deep learning has already led to improved accuracy of computer-aided detection and segmentation in medical image processing. However, further improvements are needed to reach the accuracy bar set by experienced clinicians. Further integration with other medical image processing techniques such as anatomic atlases and landmarks may help. Deeper network architectures, more efficient training, larger datasets, and faster GPUs are expected to improve performance. Insights from neuroscience may lead to improved deep learning network architectures.

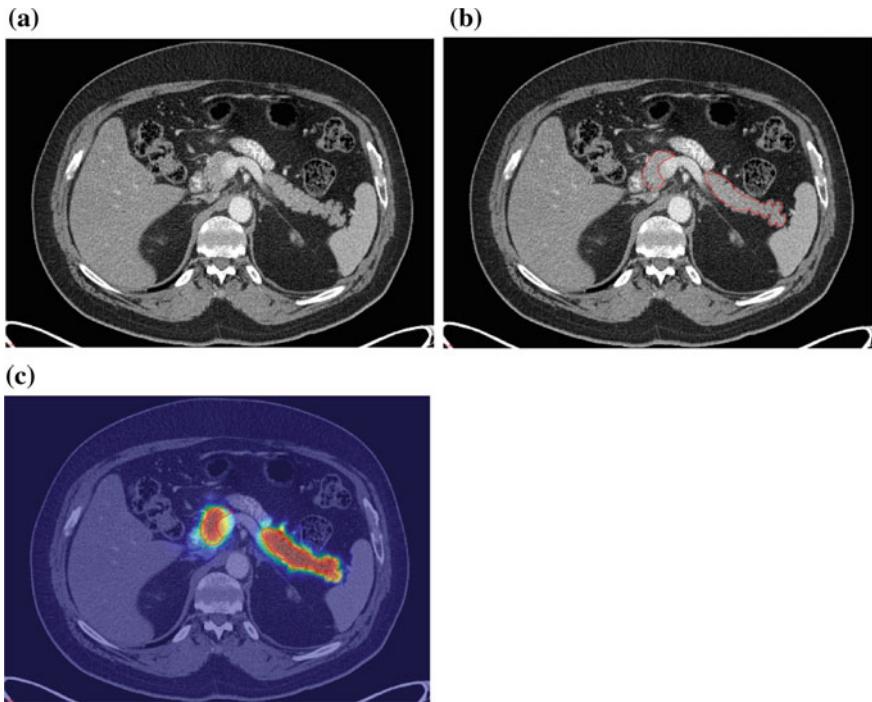


Fig. 1.1 Automated segmentation of the pancreas using deep learning. **a, b** Original contrast-enhanced abdominal CT image (**a**) without and (**b**) with (red contours) manual segmentation of the pancreas. **c** Heat-scale probability map of pancreas computed using deep learning, superimposed over the image in **b**. Reprinted from Ref. [37]

Whether or not it is achieved, the societal implications of clinician-level accuracy of learning systems need to be considered. Such systems may reduce errors, but could also cause disruption in the healthcare industry [46].

Whether performance will reach a plateau is unknown, but current deep learning systems now represent the state of the art. Given its current upward trend, the promise of deep learning in medical image analysis is bright and likely to remain so for the foreseeable future.

Acknowledgements This work was supported by the Intramural Research Program of the National Institutes of Health, Clinical Center.

References

1. Giger ML, Chan HP, Boone J (2008) Anniversary paper: history and status of CAD and quantitative image analysis: the role of Medical Physics and AAPM. *Med Phys* 35(12):5799–5820
2. Willems JL, Abreu-Lima C, Arnaud P, van Bemmel JH, Brohet C, Degani R, Denis B, Gehring J, Graham I, van Herpen G et al (1991) The diagnostic performance of computer programs for the interpretation of electrocardiograms. *N Engl J Med* 325(25):1767–1773
3. Rubin JM, Sayre RE (1978) 1978 memorial award paper: a computer-aided technique for overlaying cerebral angiograms onto computed tomograms. *Invest Radiol* 13(5):362–367
4. Fujita H, Doi K, Fencil LE, Chua KG (1987) Image feature analysis and computer-aided diagnosis in digital radiography. 2. Computerized determination of vessel sizes in digital subtraction angiography. *Med Phys* 14(4):549–556
5. Xing F, Yang L (2016) Robust nucleus/cell detection and segmentation in digital pathology and microscopy images: a comprehensive review. *IEEE Rev Biomed Eng* 9:234–263
6. Harris G (2010) CT scans cut lung cancer deaths, study finds, The New York Times. <http://www.nytimes.com/2010/11/05/health/research/05cancer.html>. 4 Nov 2010
7. Summers RM (2003) Road maps for advancement of radiologic computer-aided detection in the 21st century. *Radiology* 229(1):11–13
8. Summers RM, Selbie WS, Malley JD, Pusanik L, Dwyer AJ, Courcoutsakis N, Kleiner DE, Sneller MC, Langford C, Shelhamer JH (1997) Computer-assisted detection of endobronchial lesions using virtual bronchoscopy: application of concepts from differential geometry. In: Conference on mathematical models in medical and health sciences, Vanderbilt University
9. Summers RM, Pusanik LM, Malley JD (1998) Automatic detection of endobronchial lesions with virtual bronchoscopy: comparison of two methods. *Proc SPIE* 3338:327–335
10. Summers RM, Selbie WS, Malley JD, Pusanik LM, Dwyer AJ, Courcoutsakis N, Shaw DJ, Kleiner DE, Sneller MC, Langford CA, Holland SM, Shelhamer JH (1998) Polypoid lesions of airways: early experience with computer-assisted detection by using virtual bronchoscopy and surface curvature. *Radiology* 208:331–337
11. Summers RM, Yao J, Pickhardt PJ, Franaszek M, Bitter I, Brickman D, Krishna V, Choi JR (2005) Computed tomographic virtual colonoscopy computer-aided polyp detection in a screening population. *Gastroenterology* 129(6):1832–1844
12. Dachman AH, Obuchowski NA, Hoffmeister JW, Hinshaw JL, Frew MI, Winter TC, Van Uitert RL, Periaswamy S, Summers RM, Hillman BJ (2010) Effect of computer-aided detection for CT colonography in a multireader, multicase trial. *Radiology* 256(3):827–835
13. Wang S, Summers RM (2012) Machine learning and radiology. *Med Image Anal* 16(5):933–951
14. Jerebko AK, Malley JD, Franaszek M, Summers RM (2003) Multiple neural network classification scheme for detection of colonic polyps in CT colonography data sets. *Acad Radiol* 10(2):154–160
15. Jerebko AK, Malley JD, Franaszek M, Summers RM (2003) Computer-aided polyp detection in CT colonography using an ensemble of support vector machines. In: Cars 2003: computer assisted radiology and surgery, proceedings, vol 1256, pp 1019–1024
16. Jerebko AK, Summers RM, Malley JD, Franaszek M, Johnson CD (2003) Computer-assisted detection of colonic polyps with CT colonography using neural networks and binary classification trees. *Med Phys* 30(1):52–60
17. Malley JD, Jerebko AK, Summers RM (2003) Committee of support vector machines for detection of colonic polyps from CT scans, pp 570–578
18. Jerebko AK, Malley JD, Franaszek M, Summers RM (2005) Support vector machines committee classification method for computer-aided polyp detection in CT colonography. *Acad Radiol* 12(4):479–486
19. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):436–444
20. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database, 248–255

21. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M (2015) Imagenet large scale visual recognition challenge. *Int J Comput Vis* 115(3):211–252
22. Sahiner B, Chan HP, Petrick N, Wei DT, Helvie MA, Adler DD, Goodsitt MM (1996) Classification of mass and normal breast tissue: a convolution neural network classifier with spatial domain and texture images. *IEEE Trans Med Imaging* 15(5):598–610
23. Chan HP, Lo SC, Sahiner B, Lam KL, Helvie MA (1995) Computer-aided detection of mammographic microcalcifications: pattern recognition with an artificial neural network. *Med Phys* 22(10):1555–1567
24. Gulcehre C (2016) Deep Learning Software Links. http://deeplearning.net/software_links/. Accessed 18 May 2016
25. Shin H-C, Lu L, Kim L, Seff A, Yao J, Summers RM (2015) Interleaved text/image deep mining on a very large-scale radiology database. In: The IEEE conference on computer vision and pattern recognition (CVPR), pp 1090–1099
26. Shin H.-C, Roberts K, Lu L, Demner-Fushman D, Yao J, Summers RM (2016) Learning to read chest X-rays: recurrent neural cascade model for automated image annotation. arXiv preprint [arXiv:1603.08486](https://arxiv.org/abs/1603.08486)
27. Wang X, Lu L, Shin H, Kim L, Bagheri M, Nogues I, Yao J, Summers RM (2017) Unsupervised joint mining of deep features and image labels for large-scale radiology image annotation and scene recognition. IEEE Winter Conference on Applications of Computer Vision (WACV) pp 998–1007. [arXiv:1701.06599](https://arxiv.org/abs/1701.06599)
28. Roth HR, Lu L, Liu J, Yao J, Seff A, Cherry K, Kim L, Summers RM (2016) Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE TMI* 35(5):1170–1181
29. Shin H-C, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans Med Imaging* 35(5):1285–1298
30. Greenspan H, van Ginneken B, Summers RM (2016) Guest editorial deep learning in medical imaging: overview and future promise of an exciting new technique. *IEEE Trans Med Imaging* 35(5):1153–1159
31. Krishna R, Zhu Y, Groth O, Johnson J, Hata K, Kravitz J, Chen S, Kalantidis Y, Li L.-J, Shamma DA (2016) Visual genome: connecting language and vision using crowdsourced dense image annotations. arXiv preprint [arXiv:1602.07322](https://arxiv.org/abs/1602.07322)
32. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. [arXiv:1512.03385](https://arxiv.org/abs/1512.03385)
33. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780
34. Jaderberg M, Simonyan K, Zisserman A (2015) Spatial transformer networks. In: Advances in neural information processing systems pp 2008–2016
35. Ciompi F, de Hoop B, van Riel SJ, Chung K, Scholten ET, Oudkerk M, de Jong PA, Prokop M, van Ginneken B (2015) Automatic classification of pulmonary peri-fissural nodules in computed tomography using an ensemble of 2D views and a convolutional neural network out-of-the-box. *Med Image Anal* 26(1):195–202
36. Anavi Y, Kogan I, Gelbart E, Geva O, Greenspan H (2015) A comparative study for chest radiograph image retrieval using binary, texture and deep learning classification. In: 2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC), pp 2940–2943
37. Roth HR, Lu L, Farag A, Shin H-C, Liu J, Turkbey EB, Summers RM (2015) DeepOrgan: multi-level deep convolutional networks for automated pancreas segmentation. In: Navab N, Hornegger J, Wells WM, Frangi AF (eds) Medical image computing and computer-assisted intervention – MICCAI 2015, Part I, vol 9349. LNCS. Springer, Heidelberg, pp 556–564
38. Prasoon A, Petersen K, Igel C, Lauze F, Dam E, Nielsen M (2013) Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. *Med Image Comput Comput Assist Interv* 16(Pt 2):246–253

39. Grand challenges in biomedical image analysis. <http://grand-challenge.org/>. Accessed 18 May 2016
40. VISCRERAL. <http://www.visceral.eu/>. Accessed 14 Dec 2015
41. Roth HR, Summers RM (2015) CT lymph nodes. <https://wiki.cancerimagingarchive.net/display/Public/CT+Lymph+Nodes>. Accessed 14 Dec 2015
42. Roth HR, Farag A, Turkbey E.B, Lu L, Liu J, Summers RM (2016) Data from Pancreas-CT. The cancer imaging archive. <https://wiki.cancerimagingarchive.net/display/Public/Pancreas-CT>. Accessed 18 May 2016
43. Nguyen TB, Wang SJ, Anugu V, Rose N, McKenna M, Petrick N, Burns JE, Summers RM (2012) Distributed human intelligence for colonic polyp classification in computer-aided detection for CT colonography. Radiology 262(3):824–833
44. McKenna MT, Wang S, Nguyen TB, Burns JE, Petrick N, Summers RM (2012) Strategies for improved interpretation of computer-aided detections for CT colonography utilizing distributed human intelligence. Med Image Anal 16(6):1280–1292
45. Albarqouni S, Baur C, Achilles F, Belagiannis V, Demirci S, Navab N (2016) AggNet: deep learning from crowds for mitosis detection in breast cancer histology images. IEEE Trans Med Imaging 35(5):1313–1321
46. Summers RM (21 April 2016) Progress in fully automated abdominal CT interpretation. AJR Am J Roentgenol, 1–13

Chapter 2

Review of Deep Learning Methods in Mammography, Cardiovascular, and Microscopy Image Analysis

Gustavo Carneiro, Yefeng Zheng, Fuyong Xing and Lin Yang

Abstract Computerized algorithms and solutions in processing and diagnosis mammography X-ray, cardiovascular CT/MRI scans, and microscopy image play an important role in disease detection and computer-aided decision-making. Machine learning techniques have powered many aspects in medical investigations and clinical practice. Recently, deep learning is emerging a leading machine learning tool in computer vision and begins attracting considerable attentions in medical imaging. In this chapter, we provide a snapshot of this fast growing field specifically for mammography, cardiovascular, and microscopy image analysis. We briefly explain the popular deep neural networks and summarize current deep learning achievements in various tasks such as detection, segmentation, and classification in these heterogeneous imaging modalities. In addition, we discuss the challenges and the potential future trends for ongoing work.

2.1 Introduction on Deep Learning Methods in Mammography

Breast cancer is one of the most common types of cancer affecting the lives of women worldwide. Recent statistical data published by the World Health Organisation (WHO) estimates that 23% of cancer-related cases and 14% of cancer-related

G. Carneiro (✉)

Australian Centre for Visual Technologies, The University of Adelaide,
Adelaide, SA, Australia
e-mail: gustavo.carneiro@adelaide.edu.au

Y. Zheng

Medical Imaging Technologies, Siemens Healthcare, Princeton, NJ, USA
e-mail: yefeng.zheng@siemens.com

F. Xing · L. Yang

Department of Electrical and Computer Engineering,
J. Crayton Pruitt Family Department of Biomedical Engineering,
University of Florida, Gainesville, FL 32611, USA
e-mail: f.xing@ufl.edu

deaths among women are due to breast cancer [1]. The most effective tool to reduce the burden associated with breast cancer consists of early detection in asymptomatic women via breast cancer screening programs [2], which commonly use mammography for breast imaging. Breast screening using mammography comprises several steps, which include the detection and analysis of lesions, such as masses and calcifications, that are used in order to estimate the risk that the patient is developing breast cancer. In clinical settings, this analysis is for the most part a manual process, which is susceptible to the subjective assessment of a radiologist, resulting in a potentially large variability in the final estimation. The effectiveness of this manual process can be assessed by recent studies that show that this manual analysis has a sensitivity of 84% and a specificity of 91% [3]. Other studies show evidence that a second reading of the same mammogram either from radiologists or from computer-aided diagnosis (CAD) systems can improve this performance [3]. Therefore, given the potential impact that second reading CAD systems can have in breast screening programs, there is a great deal of interest in the development of such systems.

2.2 Deep Learning Methods in Mammography

A CAD system that can analyze breast lesions from mammograms usually comprises three steps [3]: (1) lesion detection, (2) lesion segmentation, and (3) lesion classification. The main challenges involved in these steps are related to the low signal-to-noise ratio present in the imaging of the lesion, and the lack of a consistent location, shape, and appearance of lesions [4, 5]. Current methodologies for lesion detection involve the identification of a large number of candidate regions, usually based on the use of traditional filters, such as morphological operators or difference of Gaussians [6–13]. These candidates are then processed by a second stage that aims at removing false positives using machine learning approaches (e.g., region classifier) [6–13]. The main challenges faced by lesion detection methods are that they may generate a large number of false positives, while missing a good proportion of true positives [4]; in addition, another issue is the poor alignment of the detected lesion in terms of translation and scale within the candidate regions—this issue has negative consequences for the subsequent lesion segmentation that depends on a relatively precise alignment. Lesion segmentation is then addressed with global/local energy minimisation models on a continuous or discrete space [14–16]. The major roadblock faced by these methods is the limited availability of annotated datasets that can be used in the training of the segmentation models. This is a particularly important problem because, differently from the detection and classification of lesions, the segmentation of lesions is not a common task performed by radiologists, which imposes strong limitations in the annotation process and, as a consequence, in the availability of annotated datasets. In fact, the main reason behind the need for a lesion segmentation is the assumption that the lesion shape is an important feature in the final stage of the analysis: lesion classification. This final stage usually involves the extraction of manually or automatically designed features from the lesion image

and shape and the use of those features with traditional machine learning classifiers [17–19]. In this last stage, the main limitation is with respect to the features being extracted for the classification because these features are usually hand-crafted, which cannot guarantee optimality for this classification stage.

The successful use and development of deep learning methods in computer vision problems (i.e., classification and segmentation) [20–24] have motivated the medical image analysis community to investigate the applicability of such methods in medical imaging segmentation and classification problems. Compared to the more traditional methods presented above (for the problem of mammogram analysis), deep learning methods offer the following clear advantages: automated learning of features estimated based on specific detection/segmentation/classification objective functions; opportunity to build complete “end-to-end” systems that take an image, detect, segment, and classify visual objects (e.g., breast lesion) using a single model and a unified training process. However, the main challenge faced by deep learning methods is the need for large annotated training sets given the scale of the parameter space, usually in the order of 10^6 parameters. This problem is particularly important in medical image analysis applications, where annotated training sets rarely have more than a few thousand samples. Therefore, a great deal of research is focused on the adaptation of deep learning methods to medical image analysis applications that contain relatively small annotated training sets.

There has been an increasing interest in the development of mammogram analysis methodologies based on deep learning. For instance, the problem of breast mass segmentation has been addressed with the use of a structured output model, where several potential functions are based on deep learning models [25–27]. The assumption here is that deep learning models alone cannot produce results that are accurate enough due to the small training set size problem mentioned above, but if these models are combined with a structured output model that makes assumptions about the appearance and shape of masses, then it is possible to have a breast mass segmentation that produces accurate results—in fact this method holds the best results in the field in two publicly available datasets [19, 28]. Segmentation of breast tissue using deep learning alone has been successfully implemented [29], but it is possible that a similar structured output model could improve even more the accuracy obtained. Dhungel et al. [30] also worked on a breast mass detection methodology that consists of a cascade of classifiers based on the Region Convolutional Neural Network (R-CNN) [23] approach. The interesting part is that the candidate regions produced by the R-CNN contain too many false positives, so the authors had to include an additional stage based on a classifier to eliminate those false positives. Alternatively, Ertosun and Rubin [31] propose a deep learning-based mass detection method consisting of a cascade of deep learning models trained with DDSM [28]—the main reason that explains the successful use of deep learning models here is the size of DDSM, which contains thousands of annotated mammograms.

The classification of lesions using deep learning [32–34] has also been successfully implemented in its simplest form: as a simple lesion classifier. Carneiro et al. [35] have proposed a system that can classify the unregistered two views of a mammography exam (cranial-caudal and mediolateral-oblique) and their respective

segmented lesions and produce a classification of the whole exam. The importance of this work lies in its ability to process multi-modal inputs (images and segmentation maps) that are not registered, in its way of performing transfer learning from computer vision datasets to medical image analysis datasets, and also in its capability of producing high-level classification directly from mammograms. A similar high-level classification using deep learning estimates the risk of developing breast cancer by scoring breast density and texture [36, 37]. Another type of high-level classification is the method proposed by Qiu et al. [38] that assesses the short-term risk of developing breast cancer from a normal mammogram.

2.3 Summary on Deep Learning Methods in Mammography

Based on the recent results presented above, it is clear that the use of deep learning is allowing accuracy improvements in terms of mass detection, segmentation, and classification. All the studies above have been able to mitigate the training set size issue with the use of regularization techniques or the combination of different approaches that can compensate the relatively poor generalization of deep learning methods trained with small annotated training sets. More importantly, deep learning is also allowing the implementation of new applications that are more focused on high-level classifications that do not depend on lesion segmentation. The annotation for this higher level tasks is readily available from clinical datasets, which generally contain millions of cases that can be used to train deep learning models in a more robust manner. These new applications are introducing a paradigm shift in how the field analyzes mammograms: from the classical three-stage process (detection, segmentation, and classification of lesions) trained with small annotated datasets to a one-stage process consisting of lesion detection and classification trained with large annotated datasets.

2.4 Introduction on Deep Learning for Cardiological Image Analysis

Cardiovascular disease is the number one cause of death in the developed countries and it claims more lives each year than the next seven leading causes of death combined [39]. The costs for addressing cardiovascular disease in the USA will triple by 2030, from 273 billion to 818 billion (in 2008 dollars) [40]. With the capability of generating images of a patient's inside body non-invasively, medical imaging is ubiquitously present in the current clinical practice. Various imaging modalities, such as computed tomography (CT), magnetic resonance imaging (MRI), ultrasound, and nuclear imaging, are widely available in clinical practice to generate images

of the heart, and different imaging modalities meet different clinical requirements. For example, ultrasound is most widely used for cardiac function analysis (i.e., the pumping of a cardiac chamber) due to its low cost and free of radiation dose; nuclear imaging and MRI are used for myocardial perfusion imaging to measure viability of the myocardium; CT reveals the most detailed cardiac anatomical structures and is routinely used for coronary artery imaging; while fluoroscopy/angiography is the workhorse imaging modality for cardiac interventions.

Physicians review these images to determine the health of the heart and to diagnose disease. Due to the large amount of information captured by the images, it is time consuming for physicians to identify the target anatomy and to perform measurements and quantification. For example, many 3D measurements (such as the volume of a heart chamber, the heart ejection fraction, the thickness and the thickening of the myocardium, or the strain and torsion of the myocardium) are very tedious to calculate without help from an intelligent post-processing software system. Various automatic or semi-automatic cardiac image analysis systems have been developed and demonstrated to reduce the exam time (thereby increase the patient throughput), increase consistency and reproducibility of the exam, and boost diagnosis accuracy of physicians.

Cardiovascular structures are composed of the heart (e.g., cardiac chambers and valves) and vessels (e.g., arteries and veins). A typical cardiac image analysis pipeline is composed of the following tasks: detection, segmentation, motion tracking, quantification, and disease diagnosis. For an anatomical structure, detection means determining the center, orientation, and size of the anatomy; while, for a vessel, it often means extraction of the centerline since a vessel has a tubular shape [41]. Early work on cardiac image analysis usually used non-learning-based data-driven approaches, for example, from simple thresholding and region growing to more advanced methods (like active contours, level sets, graph cuts, and random walker) for image segmentation. In the past decade, machine learning has penetrated into almost all steps of the cardiac image analysis pipeline [42, 43]. The success of a machine learning-based approach is often determined by the effectiveness and efficiency of the image features.

The recent advance of deep learning demonstrates that a deep neural network can automatically learn hierarchical image representations, which often outperform the most effective hand-crafted features developed after years of feature engineering. Encouraged by the great success of deep learning on computer vision, researchers in the medical imaging community quickly started to adapt deep learning for their own tasks. The current applications of deep learning on cardiac image segmentation are mainly focused on two topics: left/right ventricle segmentation [44–52] and retinal vessel segmentation [53–60]. Most of them are working on 2D images as input; while 3D deep learning is still a challenging task. First, evaluating a deep network on a large volume may be too computationally expensive for a real clinical application. Second, a network with a 3D patch as input requires more training data since a 3D patch generates a much bigger input vector than a 2D patch. However, the medical imaging community is often struggling with limited training samples (often in hundreds or

thousands) due to the difficulty to generate and share patients' images. Nevertheless, we started to see a few promising attempts [61–63] to attack the challenging 3D deep learning tasks.

2.5 Deep Learning-Based Methods for Heart Segmentation

Carneiro et al. [44] presented a method using a deep belief network (DBN) to detect an oriented bounding box of the left ventricle (LV) on 2D ultrasound images of the LV long-axis views. One advantage of the DBN is that it can be pre-trained layer by layer using unlabeled data; therefore, good generalization capability can be achieved with a small number of labeled training images. A 2D-oriented bounding box has five pose parameters (two for translation, one for rotation, and two for anisotropic scaling). Since an exhaustive searching in this five-dimensional pose parameter space is time consuming, they proposed an efficient search strategy based on the first- or second-order derivatives of the detection score, which accelerated the detection speed by ten times. Furthermore, the DBN has also been applied to train a boundary detector for segmentation refinement using an active shape model (ASM). The LV detection/segmentation module can also be integrated in a particle filtering framework to track the motion of the LV [44]. This work was later extended to segment the right ventricle (RV) too [46]. In follow-up work [47], the DBN was applied to segment the LV on short-axis cardiac MR images. Similarly, the LV bounding box is detected with a DBN. Furthermore, another DBN was trained to generate a pixel-wise probability map of the LV. Instead of using the ASM as [44], the level set method is applied on the probability map to generate the final segmentation.

Avendi et al. [50] proposed a convolutional network (CNN)-based method to detect an LV bounding box on a short-axis cardiac MR image. Stacked autoencoder was then applied to generate an initial segmentation of the LV, which was used to initialize the level set function. Their level set function combines a length-based energy term, a region-based term, and the prior shape. Instead of running level set on the probability map as [44], it was applied on the initial image.

Different to [44, 50], Chen et al. proposed to use a fully convolutional network (FCN) to segment the LV on 2D long-axis ultrasound images [52]. In [44, 50], deep learning was applied in one or two steps of the whole image analysis pipeline. Differently, the FCN can be trained end-to-end without any preprocessing or post-processing. It can generate a segmentation label for each pixel efficiently since the convolution operation is applied once on the whole image. Due to the limited training samples, a deep network often suffers from the over-fitting issue. There are multiple canonical LV long-axis views, namely apical two-chamber (A2C), three-chamber (A3C), four-chamber (A4C), and five-chamber (A5C) views. Instead of training an LV segmentation network for each task, the problem was formulated as a multi-task learning, where all tasks shared the low-level image representations. At the high level, each task had its own classification layers. The segmentation was refined iteratively by focusing on the LV region detected by the previous iteration. Experiments showed that

the iterative cross-domain deep learning approach outperformed alternative single-domain deep learning, especially for tasks with limited training samples.

Zhen et al. [49] presented an interesting method for direct estimation of a ventricular volume from images without performing segmentation at all. They proposed a new convolutional deep belief network. A DBN is composed of stacked restricted Boltzmann machine (RBM), where each layer is fully connected to the previous layer. Due to the full connectivity, the network has more parameters than a CNN; therefore it is more prone to over-fit. In [49], the first RBM layer was replaced with a multi-scale convolutional layer. The convolutional DBN was trained without supervision on unlabeled data and the trained network was used as an image feature extractor. A random forest regressor was then trained on the DBN image features to directly output an estimate of the LV area on each MR slice. Summing LV areas from all images results in the final volume estimate.

Due to the difficulty of 3D deep learning, all the above-reviewed methods work on 2D images, even though the input may be 3D. A 3D volume contains much richer information than a 2D image. Therefore, an algorithm leveraging 3D image information may be more robust. For heart segmentation, we only found one example using 3D deep learning, namely marginal space deep learning (MSDL) [62]. MSDL is an extension of marginal space learning (MSL), which uses hand-crafted features (i.e., Haar-like features and steerable features) and a boosting classifier. Here, the hand-crafted features are replaced with automatically learned sparse features and a deep network is exploited as the classifier. In [62], Ghesu et al. demonstrated the efficiency and robustness of MSDL on aortic valve detection and segmentation in 3D ultrasound volumes. Without using GPU, the aortic valve can be successfully segmented in less than one second with higher accuracy than the original MSL. MSDL is a generic approach and it can be easily re-trained to detect/segment other anatomies in a 3D volume.

2.6 Deep Learning-Based Methods for Vessel Segmentation

Early work on vessel segmentation used various hand-crafted vesselness measurements to distinguish the tubular structure from background [64]. Recently, we saw more and more work to automatically learn the most effective application-specific vesselness measurement from an expert-annotated dataset [65, 66]. Deep learning has potential to replace those classifiers to achieve better segmentation accuracy. However, the current applications of deep learning on vessel segmentation are mainly focused on retinal vessels in fundus images [53–60]. We only found limited work on other vessels, e.g., the coronary artery [62, 63] and carotid artery [61]. We suspect that the main reason is that a fundus image is 2D; therefore, it is much easier to apply an off-the-shelf deep learning package on this application. Other vessels in a 3D volume (e.g., CT or MR) are tortuous and we have to take the 3D context for a reliable segmentation. With the recent development of 3D deep learning, we expect to see more applications of deep learning on other vessels too.

In most work, pixel-wise classification is performed by a trained deep network to directly output the segmentation mask. For example, Wang et al. [53] applied a CNN to retinal vessel segmentation. To further improve the accuracy, they also used the CNN as a trainable feature extractor: activations of the network at different layers are taken as features to train random forests (RF). State-of-the-art performance has been achieved by an ensemble of RF classifiers on the public DRIVE and STARE datasets. Li et al. [54] presented another method based on an FCN with three layers. They formulated the task as cross-modality data transformation from the input image to vessel map. The first hidden layer was pre-trained using denoising autoencoder, while the other two hidden layers were randomly initialized. Different to [53] (which generates a label of the central pixel of an input patch), Li et al. approach outputs labels for all pixels in the patch. Since overlapping patches are extracted during classification, a pixel appears on multiple patches. The final label of the pixel is determined by majority voting to improve the classification accuracy. Fu et al. [60] adapted a holistically nested edge detection (HED) method for retinal vessel segmentation. HED is motivated by the FCN and deeply supervised network, where the outputs of intermediate layers are also directly connected to the final classification layer. After getting the the vessel probability map using HED, a conditional random field is applied to further improve the segmentation accuracy.

Since pixel-wise classification is time consuming, Wu et al. [58] proposed to combine pixel classification and vessel tracking to accelerate the segmentation speed. Starting from a seed point, a vessel is traced in the generalized particle filtering framework (which is a popular vessel tracing approach), while the weight of each particle is set by the CNN classification score at the corresponding position. Since CNN classification is invoked only on a suspected vessel region during tracing, the segmentation speed was accelerated by a fact of two. Besides retinal vessel segmentation, deep learning has also been exploited to detect retinal vessel microaneurysms [56] and diabetic retinopathy [57] from a fundus image.

Coronary artery analysis is the killer application of cardiac CT. Due to the tiny size of a coronary artery, CT is currently the most widely used noninvasive imaging modality for coronary artery disease diagnosis due to its superior image resolution (around 0.2–0.3 mm for a state-of-the-art CT scanner). Even with a quite amount of published work on coronary artery segmentation in the literature [64], we only found one work using deep learning [62] for coronary artery centerline extraction. Coronary centerline extraction is still challenging task. To achieve a high detection sensitivity, false positives are unavoidable. The false positives mainly happen on coronary veins or other tubular structures; therefore, traditional methods cannot reliably distinguish false positives from true coronary arteries. In [41], a CNN is exploited to train a classifier which can distinguish leakages from good centerlines. Since the initial centerline is given, the image information can be serialized as a 1D signal along the centerline. Here, the input channels consist of various profiles sampled along the vessel such as vessel scale, image intensity, centerline curvature, tubularity, intensity, and gradient statistics (mean, standard deviation) along and inside a cross-sectional circular boundary, and distance to the most proximal point in the branch. Deep learning-based

branch pruning increases the specificity from 50 to 90% with negligible degradation of sensitivity.

Similar to heart segmentation reviewed in Sect. 2.5, almost all previous work on deep learning for vessel segmentation was focused on 2D. Recently, Zheng et al. [61] proposed an efficient 3D deep learning method for vascular landmark detection. A two-step approach is exploited for efficient detection. A shallow network (with one hidden layer) is used for the initial testing of all voxels to obtain a small number of promising candidates, followed by more accurate classification with a deep network. In addition, they proposed several techniques, i.e., separable filter decomposition and network sparsification, to speed up the evaluation of a network. To mitigate the over-fitting issue, thereby increasing detection robustness, small 3D patches from a multi-resolution image pyramid are extracted as network input. The deeply learned image features are further combined with Haar-like features to increase the detection accuracy. The proposed method has been quantitatively evaluated for carotid artery bifurcation detection on a head–neck CT dataset. Compared to the state-of-the-art, the mean error is reduced by more than half, from 5.97 to 2.64 mm, with a detection speed of less than 1 s/volume without using GPU.

Wolterink et al. [63] presented an interesting method using a 2.5D or 3D CNN for coronary calcium scoring in CT angiography. Normally, a standard cardiac CT protocol includes a non-contrasted CT scan for coronary calcium scoring [67] and a contracted scan (called CT angiography) for coronary artery analysis. If calcium scoring can be performed on a contrasted scan, the dedicated non-contrasted scan can be removed from the protocol to save radiation dose to a patient. However, calcium scoring on CT angiography is more challenging due to the reduced intensity gap between contrasted coronary lumen and calcium. In this work voxel-wise classification is performed to identify calcified coronary plaques. For each voxel, three orthogonal 2D patches (the 2.5D approach) or a full 3D patch are used as input. A CNN is trained to distinguish coronary calcium from other tissues.

2.7 Introduction to Microscopy Image Analysis

Microscopy image analysis can provide support for improved characterization of various diseases such as breast cancer, lung cancer, brain tumor, etc. Therefore, it plays a critical role in computer-aided diagnosis in clinical practice and pathology research. Due to the large amount of image data, which continue to increase nowadays, it is inefficient or even impossible to manually evaluate the data. Computerized methods can significantly improve the efficiency and the objectiveness, thereby attracting a great deal of attention. In particular, machine learning techniques have been widely and successfully applied to medical imaging and biology research [68, 69]. Compared with non-learning or knowledge based methods that might not precisely translate knowledge into rules, machine learning acquires their own knowledge from data representations. However, conventional machine learning techniques usually do not

directly deal with raw data but heavily rely on the data representations, which require considerable domain expertise and sophisticated engineering [70].

Deep learning is one type of representation learning methods that directly process raw data (e.g., RGB images) and automatically learns the representations, which can be applied to detection, segmentation, or classification tasks. Compared with hand-crafted features, learned representations require less human intervention and provide much better performance [71]. Nowadays, deep learning techniques have made great advantages in artificial intelligence, and successfully applied to computer vision, natural language processing, image understanding, medical imaging, computational biology, etc. [70, 72]. By automatically discovering hidden data structures, it has beaten records in several computer vision tasks such as image classification [73] and speech recognition [74], and won multiple competitions in medical image analysis such as brain image segmentation [75] and mitosis detection [76]. Meanwhile, it has provided very promising performance in other medical applications [77, 78].

Recently, deep learning is emerging as a powerful tool and will continue to attract considerable interests in microscopy image analysis including nucleus detection, cell segmentation, extraction of regions of interest (ROIs), image classification, etc. A very popular deep architecture is convolutional neural networks (CNNs) [70, 79], which have obtained great success in various tasks in both computer vision [73, 80–82] and medical image analysis [83]. Given images and corresponding annotations (or labels), a CNN model is learned to generate hierarchical data representations, which can be used for robust target classification [84]. On the other hand, unsupervised learning can also be applied to neural networks for representation learning [85–87]. Autoencoder is an unsupervised neural network commonly used in microscopy image analysis, which has provided encouraging performance. One of significant benefits of unsupervised feature learning is that it does not require expensive human annotations, which are not easy to achieve in medical computing.

There exist a number of books and reviews explaining deep learning principles, historical survey, and applications in various research areas. Schmidhuber [88] presents a historical overview of deep artificial neural networks by summarizing relevant work and tracing back the origins of deep learning ideas. LeCun et al. [70] mainly review supervised learning in deep neural networks, especially CNNs and recurrent neural networks, and their successful applications in object detection, recognition, and nature language processing. The book [71] explains several established deep learning algorithms and provides speculative ideas for future research, the monograph [87] surveys general deep learning techniques and their applications (mainly) in speech processing and computer vision, and the paper [83] reviews several recent deep learning applications in medical image computing (very few in microscopy imaging). Due to the emergence of deep learning and its impacts in a wide range of disciplines, there exist many other documents introducing deep learning or relevant concepts [74, 89–92].

In this chapter, we focus on deep learning in microscopy image analysis, which covers various topics such as nucleus/cell/neuron detection, segmentation, and classification. Compared with other imaging modalities (e.g., magnetic resonance imaging, computed tomography, and ultrasound), microscopy images exhibit unique com-

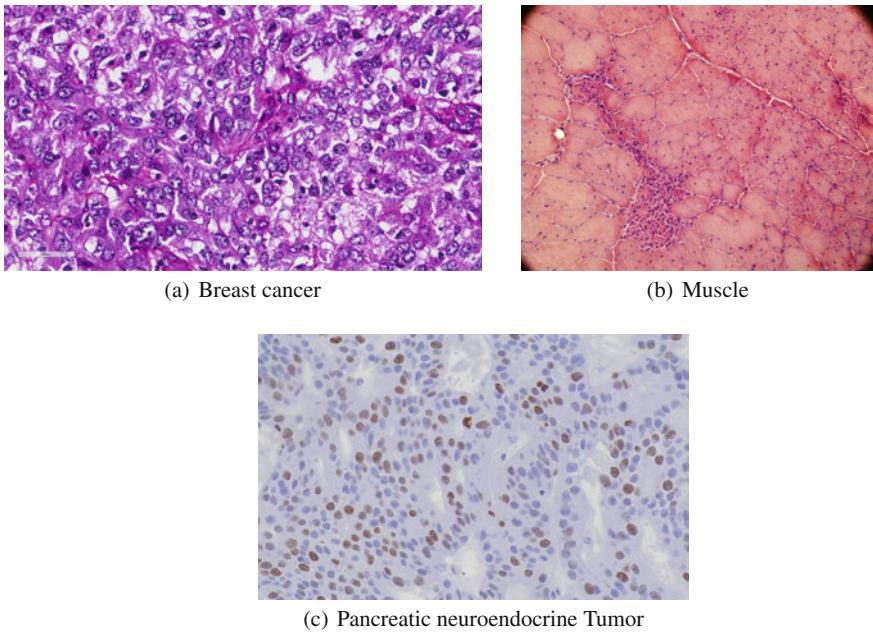


Fig. 2.1 Sample images of breast cancer, muscle, and pancreatic neuroendocrine tumor using different tissues and stain preparations. Hematoxylin and eosin (H&E) staining is used for the first two, while immunohistochemical staining is for the last. These images exhibit significant challenges such as background clutter, touching nuclei, and weak nucleus boundaries, for automated nucleus/cell detection and segmentation

plex characteristics. In digital histopathology, image data are usually generated with a certain chemistry staining and presents significant challenges including background clutter, inhomogeneous intensity, touching or overlapping nuclei/cells, etc. [72, 93–96], as shown in Fig. 2.1. We will not review all deep learning techniques in this chapter, but instead introduce and interpret those deep learning-based methods specifically designed for microscopy image analysis. We will explain the principles of those approaches and discuss their advantages and disadvantages, and finally conclude with some potential directions for future research at deep learning in microscopy image analysis.

2.8 Deep Learning Methods

Deep learning is a kind of machine learning methods involving multi-level representation learning, which starts from raw data input and gradually moves to more abstract levels via nonlinear transformations. With enough training data and sufficiently deep architectures, neural networks can learn very complex functions and

discover intricate structures in the data [70]. One significant advantage is that deep learning does not require much engineering work, which is not easy to achieve in some specific domains. Deep learning has been successfully applied to pattern recognition and prediction, and outperforms traditional machine learning methods in many domains including medical image computing [83]. More specifically, deep learning exhibits its great power in microscopy image analysis. To our knowledge, up to now there are mainly four commonly used deep networks in microscopy image analysis: CNNs, fully convolutional networks (FCNs), recurrent neural networks (RNNs), and stacked autoencoders (SAEs). More details related to optimization and algorithms can be found in [71, 89].

2.9 Microscopy Image Analysis Applications

In microscopy image analysis, deep neural networks are often used as classifiers or feature extractors to resolve various tasks in microscopy image analysis, such as target detection, segmentation, and classification. For the usage of a classifier, a deep neural network assigns a hard or soft label to each pixel of the input image in pixel-wise classification or a single label to the entire input image in image-level classification. CNNs are the most popular networks in this type of applications and their last layers are usually chosen as a multi-way softmax function corresponding to the number of target classes. For the usage of a feature extractor, a network generates a transformed representation of each input image, which can be applied to subsequent data analysis, such as feature selection or target classification. In supervised learning, usually the representation before the last layer of a CNN is extracted, but those from middle layers or even lower layers are also helpful to object recognition [111, 112]. To deal with limited data in medical imaging, it might be necessary to apply pretrain and fine-tune to the neural network. Tables 2.1 and 2.2 summarize the current deep learning achievements in microscopy image analysis.

2.10 Discussions and Conclusion on Deep Learning for Microscopy Image Analysis

Deep learning is a rapidly growing field and is emerging as a leading machine learning tool in computer vision and image analysis. It has exhibited great power in medical image computing with producing improved accuracy of detection, segmentation, or recognition tasks [83]. Most of works presented in this paper use CNNs or one type of the variants, FCNs, to solve problems in microscopy image analysis. Our conjecture is that CNNs provide consistent improved performance across a large variety of computer vision tasks and thus it might be straightforward to apply convolutional networks to microscopy image computing. More recently, FCNs have attracted a great

Table 2.1 Summary of current deep learning achievements in microscopy image analysis. SSAE = stacked sparse autoencoder, P = precision, R = recall, F₁ = F₁-score, AUC = area under curve, and ROC = Receiver operating characteristic

	Network	Usage	Topic	Data	Evaluation metric
[97]	CNN	Pixel classification	Mitosis detection	Breast cancer images	P, R, F ₁
[98]	CNN	Pixel classification	Nucleus detection	Brain tumor, NET, breast cancer images	P, R, F ₁
[99]	CNN	Pixel classification	Cell detection	Breast cancer images	P, R
[100]	CNN	Pixel classification, feature extraction	Neutrophils identification	Human squamous cell carcinoma images	P, R
[101]	CNN	Pixel classification	Cell detection	Larval zebrafish brain images	P, R, F ₁
[102]	CNN	Patch classification	Mitosis detection	NIH3T3 scratch assay culture images	Sensitivity, specificity, F ₁ , AUC
[103]	CNN	Patch scoring	Cell detection	NET, lung cancer images	P, R, F ₁
[104]	CNN	Regression	Cell, nucleus detection	Breast cancer, NET, HeLa images	P, R, F ₁
[105]	CNN	Regression	Nucleus detection, classification	Colon cancer images	P, R, F ₁ , AUC
[106]	FCN	Regression	Cell counting	Retinal pigment epithelial and precursor T Cell lymphoblastic lymphoma images	Counting difference
[107]	CNN	Voting	Nucleus detection	NET images	P, R, F ₁
[108]	CNN	Pixel classification	Mitosis detection	Breast cancer images	P, R, F ₁ , AUC, ROC, relative changes
[109]	CNN	Pixel classification	Hemorrhage detection	Color fundus images	ROC
[110]	SSAE	Feature extraction	Nucleus detection	Breast cancer images	P, R, F ₁ , average precision

Table 2.2 Summary of current deep learning achievements in microscopy image analysis. FCNN = fully connected neural network, DSC = dice similarity coefficient, PPV = positive predictive value, NPV = negative predictive value, IOU = intersection over union, MCA = mean class accuracy, ACA = average classification accuracy, and BAC = balanced accuracy

	Network	Usage	Topic	Data	Evaluation metric
[113]	CNN	Pixel classification	Neuronal membrane segmentation	Ventral nerve cord images of a <i>Drosophila</i> larva	Rand, warping, pixel errors
[114]	CNN	Pixel classification	Neuronal membrane segmentation	Ventral nerve cord images of a <i>Drosophila</i> larva	Rand, warping, pixel errors
[115]	CNN	Pixel classification	Nucleus, cell segmentation	Developing <i>C. elegans</i> embryos images	Pixel-wise error rate
[116]	CNN	Pixel classification	Nucleus, cytoplasm segmentation	Cervical images	DSC, PPV, NPV, overlapping ratio, pixel error
[117]	FCN	Pixel classification	Neuronal membrane and cell segmentation	Ventral nerve cord images of a <i>Drosophila</i> larva, Glioblastoma-astrocytoma U373 cell and HeLa cell images	Rand, warping, pixel errors, IOU
[118]	FCN	Pixel classification	Neuronal membrane segmentation	Ventral nerve cord images of a <i>Drosophila</i> larva	Rand, warping, pixel errors
[119]	RNN	Pixel classification	Neuronal membrane segmentation	Ventral nerve cord images of a <i>Drosophila</i> larva	Rand, warping, pixel errors
[120]	SDAE	Patch reconstitution	Nucleus segmentation	Brain tumor, lung cancer images	P, R, F ₁
[121]	CNN	Image classification	Image classification	Human Epithelial-2 (HEp-2) cell images	MCA, ACA
[122]	FCNN	Cell classification	Cell classification	Optical phase and loss images	ROC
[123]	CNN	Feature extraction	Image classification	Glioblastoma multiforme and low grade glioma images	F ₁ , accuracy
[124]	CNN	Feature extraction	Image classification	Colon cancer images	Accuracy

(continued)

Table 2.2 (continued)

	Network	Usage	Topic	Data	Evaluation metric
[125]	Autoencoder	Feature extraction	Image classification	Basal-cell carcinoma cancer images	Accuracy, P, R, F ₁ , specificity, BAC
[126]	SPSD	Feature extraction	Image classification	Glioblastoma multiforme and kidney clear cell carcinoma, tumorigenic breast cancer, and control cell line images	Accuracy

deal of interest due to the end-to-end training design and efficient fully convolutional inference for image semantic segmentation. FCNs begin to enter in microscopy imaging and are expected to become more popular in the future.

Model training in deep learning is usually computationally expensive and often needs programming with graphics processing units (GPUs) to reduce running time. There are several publicly available frameworks supporting deep learning. Caffe [127] is mainly written with C++ programming languages and supports command line, Python, and MATLAB interfaces. It uses Google protocol buffers to serialize data and has powered many aspects of the communities of computer vision and medical imaging. Theano [128] is a Python library that allows efficient definition, optimization, and evaluation of mathematical expressions. It is very flexible and has supported many scientific investigations. TensorFlow [129] uses data flow graphs for numerical computation and allows automatic differentiation, while Torch [130] is developed with Lua language and it is flexible as well. Another commonly used deep learning library in medical imaging is MatConvnet [131], which is a Matlab toolbox for CNNs and FCNs. It is simple and easy to use. There exist some other libraries supporting deep learning, and more information can be found in [132, 133].

Although unsupervised deep learning is applied to microscopy image analysis, the majority of the works are using supervised learning. However, deep learning with supervision usually require a large set of annotated training data, which might be prohibitively expensive in the medical domain [83]. One way to address this problem is to view a pre-trained model that is learned with other datasets, either natural or medical images, as a fixed feature extractor, and use generated features to train a target classifier for pixel-wise or image-level prediction. If the target data size is sufficiently large, it might be beneficial to initialize the network with a pre-trained model and then fine-tune it toward the target task. The initialization can be conducted in the first several or all layers depending on the data size and properties. On the other hand, semi-supervised or unsupervised learning might be a potential alternative if annotated training data are not sufficient or unavailable.

Another potential challenge of applying deep learning to microscopy image computing is to improve the network scalability, thereby adapting to high resolution images. In pathology imaging informatics, usually it is necessary to conduct quantitative analysis on whole-slide images (WSI) [134] instead of manually selected regions, since it can reduce biases of observers and provide complete information that is helpful to decision-making in diagnosis. The resolution of a WSI image is often over 50000×50000 , and has tens of thousands or millions of object of interest (e.g., nuclei or cells). Currently, pixel-wise prediction with CNNs is mainly conducted in a sliding-window manner, and clearly this will be extremely computationally expensive when dealing with WSI images. FCNs are designed for efficient inference and might be a good choice for computation improvement.

This paper provides a survey of deep learning in microscopy image analysis, which is a fast evolving field. Specifically, it briefly introduces the popular deep neural networks in the domain, summarizes current research efforts, and explains the challenges as well as the potential future trends. Deep learning has benefitted the microscopy imaging domain and we expect that it will play a more important role in the future. New learning algorithms in artificial intelligence can accelerate the process of transferring deep learning techniques from natural toward medical images and enhance its achievements.

References

1. Jemal A, Siegel R, Ward E, Hao Y, Xu J, Murray T, Thun MJ (2008) Cancer statistics, 2008. CA Cancer J Clin 58(2):71–96
2. Lauby-Secratan B, Scoccianti C, Loomis D, Benbrahim-Tallaa L, Bouvard V, Bianchini F, Straif K (2015) Breast-cancer screening—viewpoint of the IARC working group. New Engl J Med 372(24):2353–2358
3. Giger ML, Pritzker A (2014) Medical imaging and computers in the diagnosis of breast cancer. In: SPIE optical engineering + applications. International Society for Optics and Photonics, p 918908
4. Oliver A, Freixenet J, Martí J, Perez E, Pont J, Denton ER, Zwigelaar R (2010) A review of automatic mass detection and segmentation in mammographic images. Med Image Anal 14(2):87–110
5. Tang J, Rangayyan RM, Xu J, El Naqa I, Yang Y (2009) Computer-aided detection and diagnosis of breast cancer with mammography: recent advances. IEEE Trans Inf Technol Biomed 13(2):236–251
6. Kozegar E, Soryani M, Minaei B, Domingues I et al (2013) Assessment of a novel mass detection algorithm in mammograms. J Cancer Res Ther 9(4):592
7. Beller M, Stotzka R, Müller TO, Gemmeke H (2005) An example-based system to support the segmentation of stellate lesions. In: Bildverarbeitung für die Medizin 2005. Springer, pp 475–479
8. te Brake GM, Karssemeijer N, Hendriks JH (2000) An automatic method to discriminate malignant masses from normal tissue in digital mammograms. Phys Med Biol 45(10):2843
9. Campanini R, Dongiovanni D, Iampieri E, Lanconelli N, Masotti M, Palermo G, Riccardi A, Roffilli M (2004) A novel featureless approach to mass detection in digital mammograms based on support vector machines. Phys Med Biol 49(6):961
10. Eltonsy NH, Tourassi GD, Elmaghhraby AS (2007) A concentric morphology model for the detection of masses in mammography. IEEE Trans Med Imaging 26(6):880–889

11. Sampat MP, Bovik AC, Whitman GJ, Markey MK (2008) A model-based framework for the detection of spiculated masses on mammography. *Med Phys* 35(5):2110–2123
12. Bellotti R, De Carlo F, Tangaro S, Gargano G, Maggipinto G, Castellano M, Massafra R, Cascio D, Fauci F, Magro R et al (2006) A completely automated cad system for mass detection in a large mammographic database. *Med Phys* 33(8):3066–3075
13. Wei J, Sahiner B, Hadjiiski LM, Chan H-P, Petrick N, Helvie MA, Roubidoux MA, Ge J, Zhou C (2005) Computer-aided detection of breast masses on full field digital mammograms. *Med Phys* 32(9):2827–2838
14. Ball JE, Bruce LM (2007) Digital mammographic computer aided diagnosis (cad) using adaptive level set segmentation. In: 29th annual international conference of the IEEE engineering in medicine and biology society, 2007. EMBS 2007. IEEE, pp 4973–4978
15. Rahmati P, Adler A, Hamarneh G (2012) Mammography segmentation with maximum likelihood active contours. *Med Image Anal* 16(6):1167–1186
16. Cardoso JS, Domingues I, Oliveira HP (2014) Closed shortest path in the original coordinates with an application to breast cancer. *Int J Pattern Recognit Artif Intell* 29:1555002
17. Varela C, Timp S, Karssemeijer N (2006) Use of border information in the classification of mammographic masses. *Phys Med Biol* 51(2):425
18. Shi J, Sahiner B, Chan H-P, Ge J, Hadjiiski L, Helvie MA, Nees A, Wu Y-T, Wei J, Zhou C et al (2008) Characterization of mammographic masses based on level set segmentation with new image features and patient information. *Med Phys* 35(1):280–290
19. Domingues I, Sales E, Cardoso J, Pereira W (2012) Inbreast-database masses characterization. In: XXIII CBEB
20. LeCun Y, Bengio Y (1995) Convolutional networks for images, speech, and time series. In: The handbook of brain theory and neural networks, vol 3361
21. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: NIPS, vol 1, p 4
22. Farabet C, Courcier C, Najman L, LeCun Y (2013) Learning hierarchical features for scene labeling. *IEEE Trans Pattern Anal Mach Intell* 35(8):1915–1929
23. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 580–587
24. Zhang Y, Sohn K, Villegas R, Pan G, Lee H (2015) Improving object detection with deep convolutional networks via bayesian optimization and structured prediction. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 249–258
25. Dhungel N, Carneiro G, Bradley AP (2015) Deep learning and structured prediction for the segmentation of mass in mammograms. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, pp 605–612
26. Dhungel N, Carneiro G, Bradley AP (2015) Tree re-weighted belief propagation using deep learning potentials for mass segmentation from mammograms. In: 2015 IEEE 12th international symposium on biomedical imaging (ISBI), pp 760–763
27. Dhungel N, Carneiro G, Bradley AP (2015) Deep structured learning for mass segmentation from mammograms. In: 2015 IEEE international conference on image processing (ICIP), pp 2950–2954
28. Heath M, Bowyer K, Kopans D, Moore R, Kegelmeyer P (2000) The digital database for screening mammography. In: Proceedings of the 5th international workshop on digital mammography, pp 212–218
29. Dubrovina A, Kisilev P, Ginsburg B, Hashoul S, Kimmel R (2016) Computational mammography using deep neural networks. In: Workshop on deep learning in medical image analysis (DLMIA)
30. Dhungel N, Carneiro G, Bradley A (2015) Automated mass detection in mammograms using cascaded deep learning and random forests. In: 2015 international conference on digital image computing: techniques and applications (DICTA), pp 1–8
31. Ertosun MG, Rubin DL (2015) Probabilistic visual search for masses within mammography images using deep learning. In: 2015 IEEE international conference on bioinformatics and biomedicine (BIBM). IEEE, pp 1310–1315

32. Arevalo J, González FA, Ramos-Pollán R, Oliveira JL, Lopez MAG (2016) Representation learning for mammography mass lesion classification with convolutional neural networks. *Comput Methods Programs Biomed*
33. Qiu Y, Yan S, Tan M, Cheng S, Liu H, Zheng B (2016) Computer-aided classification of mammographic masses using the deep learning technology: a preliminary study. In: SPIE medical imaging. International Society for Optics and Photonics, p 978520
34. Jiao Z, Gao X, Wang Y, Li J (2016) A deep feature based framework for breast masses classification. *Neurocomputing* 197:221–231
35. Carneiro G, Nascimento J, Bradley AP (2015) Unregistered multiview mammogram analysis with pre-trained deep learning models. In: Medical image computing and computer-assisted intervention – MICCAI 2015. Springer, Berlin, pp. 652–660
36. Kallenberg M, Petersen K, Nielsen M, Ng A, Diao P, Igel C, Vachon C, Holland K, Karssemeijer N, Lillholm M (2016) Unsupervised deep learning applied to breast density segmentation and mammographic risk scoring
37. Petersen K, Nielsen M, Diao P, Karssemeijer N, Lillholm M (2014) Breast tissue segmentation and mammographic risk scoring using deep learning. In: Breast imaging. Springer, Berlin, pp 88–94
38. Qiu Y, Wang Y, Yan S, Tan M, Cheng S, Liu H, Zheng B (2016) An initial investigation on developing a new method to predict short-term breast cancer risk based on deep learning technology. In: SPIE medical imaging. International Society for Optics and Photonics, p 978521
39. Lloyd-Jones D, Adams R, Carnethon M et al (2009) Heart disease and stroke statistics – 2009 update. *Circulation* 119(3):21–181
40. Heidenreich PA, Trogdon JG, Khavjou OA et al (2011) Forecasting the future of cardiovascular disease in the United States: a policy statement from the American Heart Association. *Circulation* 123:933–944
41. Gulsun MA, Funka-Lea G, Sharma P, Rapaka S, Zheng Y (2016) Coronary centerline extraction via optimal flow paths and CNN path pruning. In: Proceedings of international conference on medical image computing and computer assisted intervention
42. Zheng Y, Barbu A, Georgescu B, Scheuering M, Comaniciu D (2008) Four-chamber heart modeling and automatic segmentation for 3D cardiac CT volumes using marginal space learning and steerable features. *IEEE Trans Med Imaging* 27(11):1668–1681
43. Zheng Y (2015) Model based 3D cardiac image segmentation with marginal space learning. In: Medical image recognition, segmentation and parsing: methods, theories and applications. Elsevier, Amsterdam, pp 383–404
44. Carneiro G, Nascimento JC, Freitas A (2012) The segmentation of the left ventricle of the heart from ultrasound data using deep learning architectures and derivative-based search methods. *IEEE Trans Image Process* 21(3):968–982
45. Carneiro G, Nascimento JC (2013) Combining multiple dynamic models and deep learning architectures for tracking the left ventricle endocardium in ultrasound data. *IEEE Trans Pattern Anal Mach Intell* 35(11):2592–2607
46. Ngo TA, Lu Z, Carneiro G (2016) Combining deep learning and level set for the automated segmentation of the left ventricle of the heart from cardiac cine magnetic resonance. *Med Image Anal* 35:159–171
47. Ngo TA, Carneiro G (2014) Fully automated non-rigid segmentation with distance regularized level set evolution initialization and constrained by deep-structured inference. In: Proceedings of IEEE conference computer vision and pattern recognition, pp 1–8
48. Emad O, Yassine IA, Fahmy AS (2015) Automatic localization of the left ventricle in cardiac MRI images using deep learning. In: Proceedings of annual international conference of the IEEE engineering in medicine and biology society, pp 683–686
49. Zhen X, Wang Z, Islam A, Bhaduri M, Chan I, Li S (2016) Multi-scale deep networks and regression forests for direct bi-ventricular volume estimation. *Med Image Anal* 30:120–129
50. Avendi MR, Kheirkhah A, Jafarkhani H (2016) A combined deep-learning and deformable-model approach to fully automatic segmentation of the left ventricle in cardiac MRI. *Med Image Anal* 30:108–119

51. Avendi MR, Kheradvar A, Jafarkhani H (2016) Fully automatic segmentation of heart chambers in cardiac MRI using deep learning. *J Cardiovasc Magn Reson* 18:351–353
52. Chen H, Zheng Y, Park J-H, Heng PA, Zhou SK (2016) Iterative multi-domain regularized deep learning for anatomical structure detection and segmentation from ultrasound images. In: Proceedings of international conference medical image computing and computer assisted intervention
53. Wang S, Yin Y, Cao G, Wei B, Zheng Y, Yang G (2015) Hierarchical retinal blood vessel segmentation based on feature and ensemble learning. *Neruocomputing* 149:708–717
54. Li Q, Feng B, Xie L, Liang P, Zhang H, Wang T (2016) A cross-modality learning approach for vessel segmentation in retinal images. *IEEE Trans Med Imaging* 35(1):109–118
55. Maji D, Santara A, Mitra P, Sheet D (2016) Ensemble of deep convolutional neural networks for learning to detect retinal vessels in fundus images. [arXiv:1603.04833](https://arxiv.org/abs/1603.04833)
56. Haloi M (2015) Improved microaneurysm detection using deep neural networks. [arXiv:1505.04424](https://arxiv.org/abs/1505.04424)
57. Chandrakumar T, Kathirvel R (2016) Classifying diabetic retinopathy using deep learning architecture. *Int J Eng Res Technol* 5(6):19–24
58. Wu A, Xu Z, Gao M, Buty M, Mollura DJ (2016) Deep vessel tracking: a generalized probabilistic approach via deep learning. In: Proceedings of IEEE international symposium on biomedical imaging, pp 1363–1367
59. Melinscak M, Prentasic P, Loncaric S (2015) Retinal vessel segmentation using deep neural networks. In: Proceedings of international conference computer vision theory and application, pp 577–582
60. Fu H, Xu Y, Wong DWK, Liu J (2016) Retinal vessel segmentation via deep learning network and fully-connected conditional random fields. In: Proceedings of IEEE international symposium on biomedical imaging, pp 698–701
61. Zheng Y, Liu D, Georgescu B, Nguyen H, Comaniciu D (2015) 3D deep learning for efficient and robust landmark detection in volumetric data. In: Proceedings of international conference on medical image computing and computer assisted intervention, pp 565–572
62. Ghesu FC, Krubasik E, Georgescu B, Singh V, Zheng Y, Hornegger J, Comaniciu D (2016) Marginal space deep learning: efficient architecture for volumetric image parsing. *IEEE Trans Med Imaging* 35(5):1217–1228
63. Wolterink JM, Leiner T, de Vos BD, van Hamersveld RW, Viergever MA, Isgum I (2016) Automatic coronary artery calcium scoring in cardiac CT angiography using paired convolutional neural networks. *Med Image Anal* 34:123–136
64. Lesage D, Angelini ED, Bloch I, Funka-Lea G (2009) A review of 3D vessel lumen segmentation techniques: models, features and extraction schemes. *Med Image Anal* 13(6):819–845
65. Zheng Y, Loziczonek M, Georgescu B, Zhou SK, Vega-Higuera F, Comaniciu D (2011) Machine learning based vesselness measurement for coronary artery segmentation in cardiac CT volumes. In: Proceedings of SPIE medical imaging, vol 7962, pp 1–12
66. Zheng Y, Tek H, Funka-Lea G (2013) Robust and accurate coronary artery centerline extraction in CTA by combining model-driven and data-driven approaches. In: Proceedings of international conference medical image computing and computer assisted intervention, pp 74–81
67. Wolterink JM, Leiner T, Coatrieux J-L, Kelm BM, Kondo S, Salgado RA, Shahzad R, Shu H, Snoeren M, Takx RA, van Vliet L, de Vos BD, van Walsum T, Willems TP, Yang G, Zheng Y, Viergever MA, Ium I (2016) An evaluation of automatic coronary artery calcium scoring with cardiac CT: the orCaScore challenge. *Med Phys* 43(5):2361–2373
68. Sommer C, Gerlich DW (2013) Machine learning in cell biology teaching computers to recognize phenotypes. *J Cell Sci* 126(24):5529–5539
69. Wernick MN, Yang Y, Brankov JG, Yourganov G, Strother SC (2010) Machine learning in medical imaging. *IEEE Signal Process Mag* 27(4):25–38
70. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436–444
71. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. Book in preparation for MIT Press

72. Xing F, Yang L (2016) Robust nucleus/cell detection and segmentation in digital pathology and microscopy images: a comprehensive review. *IEEE Rev Biomed Eng* 99
73. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: *Advances neural information processing systems*, pp 1097–1105
74. Hinton G, Deng L, Yu D, Dahl GE, Mohamed Ar, Jaitly N, Senior A, Vanhoucke V, Nguyen P, Sainath TN, Kingsbury B (2012) Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process Mag* 29(6):82–97
75. Arganda-Carreras I et al (2015) Crowdsourcing the creation of image segmentation algorithms for connectomics. *Front Neuroanat* 9(142)
76. Veta M et al (2015) Assessment of algorithms for mitosis detection in breast cancer histopathology images. *Med Image Anal* 20(1):237–248
77. Ma J, Sheridan RP, Liaw A, Dahl GE, Svetnik V (2015) Deep neural nets as a method for quantitative structure-activity relationships. *J Chem Inf Model* 55:263274
78. Xiong HY et al (2015) The human splicing code reveals new insights into the genetic determinants of disease. *Science* 347(6218)
79. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86:2278–2324
80. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: *2014 IEEE conference on computer vision and pattern recognition*, pp 580–587
81. Girshick R (2015) Fast r-cnn. In: *2015 IEEE international conference on computer vision*, pp 1440–1448
82. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: *2015 IEEE conference on computer vision and pattern recognition*, pp 3431–3440
83. Greenspan H, van Ginneken B, Summers RM (2016) Guest editorial deep learning in medical imaging: overview and future promise of an exciting new technique. *IEEE Trans Med Imaging* 35(5):1153–1159
84. LeCun Y, Kavukcuoglu K, Farabet C (2010) Convolutional networks and applications in vision. In: *IEEE international symposium on circuits and systems (ISCAS)*, pp 253–256
85. Lee H, Grosse R, Ranganath R, Ng AY (2009) Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: *Proceedings of the 26th international conference on machine learning*, pp 609–616
86. Lee H, Grosse R, Ranganath R, Ng AY (2011) Unsupervised learning of hierarchical representations with convolutional deep belief networks. *Commun ACM* 54(10):95–103
87. Deng L, Yu D (2014) Deep learning: methods and applications. *Found Trends Signal Process* 3(3–4):197–387
88. Schmidhuber J (2015) Deep learning in neural networks: an overview. *Neural Networks* 61:85–117. Published online 2014; based on TR [arXiv:1404.7828](https://arxiv.org/abs/1404.7828) [cs.NE]
89. Nielsen MA (2015) Neural networks and deep learning. Determination Press
90. Arel I, Rose DC, Karnowski TP (2010) Deep machine learning - a new frontier in artificial intelligence research [research frontier]. *IEEE Comput Intell Mag* 5(4):13–18
91. Bengio Y (2009) Learning deep architectures for ai. *Found Trends Mach Learn* 2:1–127
92. Bengio Y, Courville A, Vincent P (2013) Representation learning: a review and new perspectives. *IEEE Trans Pattern Anal Mach Intell* 35(8):1798–1828
93. Gurcan MN, Boucheron LE, Can A, Madabushi A, Rajpoot NM, Yener B (2009) Histopathological image analysis: a review. *IEEE Rev Biomed Eng* 2:147–171
94. McCann MT, Ozolek JA, Castro CA, Parvin B, Kovacevic J (2015) Automated histology analysis: opportunities for signal processing. *IEEE Signal Process Mag* 32:78–87
95. Veta M, Pluim J, van Diest P, Viergever M (2014) Breast cancer histopathology image analysis: a review. *IEEE Trans Biomed Eng* 61:1400–1411
96. Irshad H, Veillard A, Roux L, Racocianu D (2014) Methods for nuclei detection, segmentation, and classification in digital histopathology: a review – current status and future potential. *IEEE Rev Biomed Eng* 7:97–114

97. Ciresan DC, Giusti A, Gambardella LM, Schmidhuber J (2013) Mitosis detection in breast cancer histology images with deep neural networks. In: International conference medical image computing and computer-assisted intervention (MICCAI), vol 8150, pp 411–418
98. Xing F, Xie Y, Yang L (2015) An automatic learning-based framework for robust nucleus segmentation. *IEEE Trans Med Imaging* PP(99):1
99. Mao Y, Yin Z, Schober JM (2015) Iteratively training classifiers for circulating tumor cell detection. In: IEEE international symposium on biomedical imaging, pp 190–194
100. Wang J, MacKenzie JD, Ramachandran R, Chen DZ (2015) Neutrophils identification by deep learning and voronoi diagram of clusters. In: medical image computing and computer-assisted intervention – MICCAI 2015: 18th international conference, Munich, Germany, 5–9 October 2015, Proceedings, Part III, pp 226–233
101. Dong B, Shao L, Costa MD, Bandmann O, Frangi AF (2015) Deep learning for automatic cell detection in wide-field microscopy zebrafish images. In: IEEE international symposium on biomedical imaging, pp 772–776
102. Shkolyar A, Gefen A, Benayahu D, Greenspan H (2015) Automatic detection of cell divisions (mitosis) in live-imaging microscopy images using convolutional neural networks. In: 2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC), pp 743–746
103. Liu F, Yang L (2015) A novel cell detection method using deep convolutional neural network and maximum-weight independent set. In: International conference on medical image computing and computer-assisted intervention (MICCAI), vol 9351, pp 349–357
104. Xie Y, Xing F, Kong X, Yang L (2015) Beyond classification: structured regression for robust cell detection using convolutional neural network. In: International conference medical image computing and computer-assisted intervention (MICCAI), vol 9351, pp 358–365
105. Sirinukunwattana K, Raza SEA, Tsang YW, Snead DRJ, Cree IA, Rajpoot NM (2016) Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE Trans Med Imaging* 35(5):1196–1206
106. Xie W, Noble JA, Zisserman A (2015) Microscopy cell counting with fully convolutional regression networks. In: MICCAI 1st workshop on deep learning in medical image analysis
107. Xie Y, Kong X, Xing F, Liu F, Su H, Yang L (2015) Deep voting: a robust approach toward nucleus localization in microscopy images. In: International conference on medical image computing and computer-assisted intervention (MICCAI), vol 9351, pp 374–382
108. Albarqouni S, Baur C, Achilles F, Belagiannis V, Demirci S, Navab N (2016) Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE Trans Med Imaging* 35(5):1313–1321
109. van Grinsven MJJP, van Ginneken B, Hoyng CB, Theelen T, Snchez CI (2016) Fast convolutional neural network training using selective data sampling: application to hemorrhage detection in color fundus images. *IEEE Trans Med Imaging* 35(5):1273–1284
110. Xu J, Xiang L, Liu Q, Gilmore H, Wu J, Tang J, Madabhushi A (2015) Stacked sparse autoencoder (ssae) for nuclei detection on breast cancer histopathology images
111. Razavian AS, Azizpour H, Sullivan J, Carlsson S (2014) CNN features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the 2014 IEEE conference on computer vision and pattern recognition workshops, CVPRW'14, pp 512–519
112. Tajbakhsh N, Shin JY, Gurudu SR, Hurst RT, Kendall CB, Gotway MB, Liang J (2016) Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE Trans Med Imaging* 35(5):1299–1312
113. Ciresan D, Giusti A, Gambardella LM, Schmidhuber J (2012) Deep neural networks segment neuronal membranes in electron microscopy images. In: Advances in neural information processing systems, pp 2843–2851
114. Fakhry A, Peng H, Ji S (2016) Deep models for brain EM image segmentation novel insights and improved performance. *Bioinformatics* 32:2352–2358
115. Ning F, Delhomme D, LeCun Y, Piano F, Bottou L, Barbano PE (2005) Toward automatic phenotyping of developing embryos from videos. *IEEE Trans Image Process* 14(9):1360–1371

116. Song Y, Zhang L, Chen S, Ni D, Lei B, Wang T (2015) Accurate segmentation of cervical cytoplasm and nuclei based on multi-scale convolutional network and graph partitioning. *IEEE Trans Biomed Eng* 62:2421–2433
117. Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: *Medical image computing and computer-assisted intervention – MICCAI 2015: 18th international conference, Munich, Germany, 5–9 October 2015, Proceedings, Part III*, pp 234–241
118. Chen H, Qi X, Cheng J, Heng PA (2016) Deep contextual networks for neuronal structure segmentation. In: *AAAI*, pp 1167–1173
119. Stollenga MF, Byeon W, Liwicki M, Schmidhuber J (2015) Parallel multi-dimensional lstm, with application to fast biomedical volumetric image segmentation. In: *Advances in neural information processing systems*, vol 28, pp 2980–2988
120. Su H, Xing F, Kong X, Xie Y, Zhang S, Yang L (2015) Robust cell detection and segmentation in histopathological images using sparse reconstruction and stacked denoising autoencoders. In: *International conference on medical image computing and computer assisted intervention (MICCAI)*, vol 9351, pp 383–390
121. Gao Z, Wang L, Zhou L, Zhang J (2016) Hep-2 cell image classification with deep convolutional neural networks. *IEEE J Biomed Health Inf PP(99):1*
122. Chen CL, Mahjoubfar A, Tai L, Blaby IK, Huang A, Niazi KR, Jalali B (2016) Deep learning in label-free cell classification. *Sci Rep* 6(21471)
123. Xu Y, Jia Z, Ai Y, Zhang F, Lai M, Chang EIC (2015) Deep convolutional activation features for large scale brain tumor histopathology image classification and segmentation. In: *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp 947–951
124. Xu Y, Mo T, Feng Q, Zhong P, Lai M, Chang EIC (2014) Deep learning of feature representation with multiple instance learning for medical image analysis. In: *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp 1626–1630
125. Cruz-Roa AA, Ovalle JEA, Madabhushi A, Osorio FAG (2013) A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In: *Medical image computing and computer-assisted intervention-MICCAI 2013*, pp 403–410
126. Chang H, Zhou Y, Spellman P, Parvin B (2013) Stacked predictive sparse coding for classification of distinct regions in tumor histopathology. In: *Proceedings of the IEEE international conference on computer vision*, pp 169–176
127. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: convolutional architecture for fast feature embedding. [arXiv:1408.5093](https://arxiv.org/abs/1408.5093)
128. Theano Development Team (2016) Theano: a Python framework for fast computation of mathematical expressions. [arXiv:abs/1605.02688](https://arxiv.org/abs/1605.02688)
129. Abadi M et al (2015) TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org
130. Collobert R, Kavukcuoglu K, Farabet C (2011) Torch7: a matlab-like environment for machine learning. In: *BigLearn, NIPS workshop*
131. Vedaldi A, Lenc K (2015) Matconvnet – convolutional neural networks for matlab
132. Mamoshina P, Vieira A, Putin E, Zhavoronkov A (2016) Applications of deep learning in biomedicine. *Mol Pharmaceutics* 13(5):1445–1454
133. Wang W, Zhang M, Chen G, Jagadish HV, Ooi BC, Tan KL (2016) Database meets deep learning: challenges and opportunities
134. Kothari S, Phan JH, Stokes TH, Wang MD (2013) Pathology imaging informatics for quantitative analysis of whole-slide images. *J Am Med Inform Assoc* 20(6):1099–1108

Part II

Detection and Localization

Chapter 3

Efficient False Positive Reduction in Computer-Aided Detection Using Convolutional Neural Networks and Random View Aggregation

**Holger R. Roth, Le Lu, Jiamin Liu, Jianhua Yao, Ari Seff,
Kevin Cherry, Lauren Kim and Ronald M. Summers**

Abstract In clinical practice and medical imaging research, automated computer-aided detection (CADe) is an important tool. While many methods can achieve high sensitivities, they typically suffer from high false positives (FP) per patient. In this study, we describe a two-stage coarse-to-fine approach using CADe candidate generation systems that operate at high sensitivity rates (close to 100% recall). In a second stage, we reduce false positive numbers using state-of-the-art machine learning methods, namely deep convolutional neural networks (ConvNet). The ConvNets are trained to differentiate hard false positives from true-positives utilizing a set of 2D (two-dimensional) or 2.5D re-sampled views comprising random translations, rotations, and multi-scale observations around a candidate's center coordinate. During the test phase, we apply the ConvNets on unseen patient data and aggregate all probability scores for lesions (or pathology). We found that this second stage is a highly selective classifier that is able to reject difficult false positives while retaining good sensitivity rates. The method was evaluated on three data sets (sclerotic metastases, lymph nodes, colonic polyps) with varying numbers patients (59, 176, and 1,186, respectively). Experiments show that the method is able to generalize to different applications and increasing data set sizes. Marked improvements are observed in all cases: sensitivities increased from 57 to 70%, from 43 to 77% and from 58 to 75% for sclerotic metastases, lymph nodes and colonic polyps, respectively, at low FP rates per patient (3 FPs/patient).

H.R. Roth · L. Lu (✉) · J. Liu · J. Yao · A. Seff · K. Cherry · L. Kim · R.M. Summers
Imaging Biomarkers and Computer-Aided Diagnosis Laboratory,
Radiology and Imaging Sciences Department, National Institutes
of Health Clinical Center, Bethesda, MD 20892-1182, USA
e-mail: le.lu@nih.gov

3.1 Introduction

Cancer is a leading cause of death in the world [1]. Timely detection of abnormalities and precursors of cancer can help fighting the disease. Accurate computer-aided detection (CADe) can help support the radiological diagnostic process. CADe can be used in determining the stage of a patient’s disease, potentially enhancing the treatment regimens [2]. For this purpose, computed tomography (CT), is often applied as a screening and staging modality that can visualize abnormal anatomy, including tumors and associated metastases. Still, diagnosis of CT scans is primarily done by hand, involving a radiologist scrolling through often thousands of image slices per patient. Today’s restrictions on radiologists’ time are prone to cause human errors when performing this complex and sometimes tedious task. Here, CADe has great potential to reduce radiologists’ clinical workload, serving as first or second readers [3–5], ultimately improving the disease assessment.

While CADe has been an area of active research for the preceding decades, most methods rely on pre-determined metrics, so called hand-crafted features. For example, intensity statistics, histogram of oriented gradients (HoG) [6], scale-invariant feature transform (SIFT) [7], Hessian based shape descriptors (such as blobness) [8], and many others are applied with the hope that these features can help differentiate normal from abnormal anatomy. They are typically computed on local regions of interest (ROIs) and then used to train shallow classifiers like support vector machines (SVM) or random forests. The process of finding suitable features requires considerable engineering skills and efforts to generate CADe systems that perform sufficiently. At present, only few examples of CADe have made it into the clinic, e.g., [9–13]. In many cases, CADe methods suffer from low sensitivity and/or specificity levels, and have not made the jump from academic research papers to clinical practice.

In this paper, we present a method to improve existing CADe systems by in hierarchical two-tiered approach that aims at high recalls together with reasonable low FP rates per patient. This is achieved by efficiently integrating state-of-the-art deep convolutional neural networks [14, 15] into CADe pipelines.

3.2 Related Work

The recent success of ConvNets in computer vision can be mainly attributed to more accessible and affordable parallel computation, i.e., Graphical Processing Units (or GPUs) and the increase in large amounts of annotated training sets. This made it feasible to train very deep ConvNets for recognition and classification [16, 17]. However, even modestly sized networks have been shown to increase the state-of-the-art in many applications that relied on hand-crafted features [18–20]. This has made ConvNets one of the work horses of in the field of “deep learning” [21, 22].

Shortly after the success of ConvNets in classifying natural images [14], they have also shown substantial advancements in biomedical applications across different

modalities, including electron microscopy (EM) images [23], digital pathology [24, 25], MRI [26], and computed tomography (CT) [27–29].

In this study, we show that ConvNets can be an efficient second stage classifier in CADe by employing random sets of 2D or 2.5D sampled views or observations. Random 2D/2.5D image decomposition can be an universal representation for utilizing ConvNets in CADe problems. The approach has also been shown to be applicable to problems where each view is sampled under some problem-specific constraints, e.g., using the local orientation vessels [30]). ConvNet scores can be simply aggregated at test time, making a more robust classifier. Here, we show that the proposed approach is generalizable by validating three different datasets with different numbers of patients and CADe applications. In all cases, we can report marked improvement compared to the initial performance of the CADe systems: sensitivities improve from 57 to 70%, from 43 to 77% and 58 to 75% at 3 FPs per patient for sclerotic metastases [4], lymph nodes [31, 32] and colonic polyps [10, 33], respectively. Our results indicate that ConvNets can be applied for effective false positive pruning while maintaining high sensitivity recalls in modern CADe systems.

3.2.1 Cascaded Classifiers in CADe

Cascaded classifiers for FP reduction have been proposed before and shown to be a valid strategy for CADe system design. One strategy is to design post-processing filters that can clean up candidates based on certain hand-crafted rules, such as for the removal of 3D flexible tubes [34], ileo-cecal valve [35], or extra-colonic findings [36] in CT colonography.

Alternatively, the classifiers can be retrained in a cascaded fashion using only pre-filtered candidates with reasonable high detection scores. This however is often not very effective and is not often employed in practice. A better strategy is to derive new image features at the candidate locations in order to train new classifiers [6, 27, 37–39]. Since the search space has been already narrowed down by the candidate generation step, more computationally expensive features can be extracted for the FP reduction stage. The hope is that these new features can reveal information that was omitted during candidate generation and hence arrive at a better final decision. We follow this last approach in this paper and derive new “data-driven” image features using deep ConvNets for classification.

3.3 Methods

3.3.1 Convolutional Neural Networks

Let us now introduce deep convolutional networks (ConvNets). As their name suggest, ConvNets apply convolutional filters to compute image features that are useful

for classification. The convolutional filter kernel elements are learned from the raw image training data in a supervised fashion. This is important as it avoids the “hand-crafting” of features for classification, while learning features that are useful for the classification problem to be solved [22]. Examples of trained convolutional filter kernels and responses of the first layer are shown in Fig. 3.2. The ConvNet was able to learn different kernel features that respond to certain texture pattern in the image. These complex features are useful to capture the essential information that is necessary to classify a ROI/VOI as TP or FP for the classification task at hand. A data-driven approach of feature learning like this is a major advance over trying to find suitable features by hand and has shown good performance on state-of-the-art data sets [17].

Furthermore, convolutional layers can be stacked in order to allow for several hierarchies of feature extraction. As shown in this study, similar configured ConvNet architectures can be used to detect very different lesions or pathologies without the need of manual feature design. An example of learned filter kernels from the first convolutional layer is shown in Fig. 3.1.

In addition to convolutional layers, typical ConvNets possess *max-pooling* layers that summarize feature activations across neighboring pixels (Fig. 3.2). This enables the ConvNet to learn features that are more spatially invariant to the location of objects within the input image. The last convolution layer is often fed into *locally connected* layers that act similar to a convolutional layer but without weight sharing [14]. Finally, the classification is achieved by *fully connected* neural network layers that are often topped by a *softmax* layer that provides a probabilistic score for each class. Intuitively, the ConvNet encodes the input image with increasing abstraction as the features are propagated through the layers of the network, concluding with a final abstract classification choice [21].

Popular methods to avoid overfitting during training are to design the fully connected layers as “*DropOut*” [40, 41] or “*DropConnect*” [42] layers. They can act as regularizers by preventing the co-adaptation of units in the neural network. In this study, we employ a relatively modest ConvNet with two convolutional layers, two locally connected layers, and a fully connected layer with a final two-way softmax layer for classification. Our experiments are based on the open-source

Fig. 3.1 A convolutional neural network (ConvNet) uses learned filter kernels to computer features from the input region of interest. In order to make the response image the same size as the input, the input image can be padded

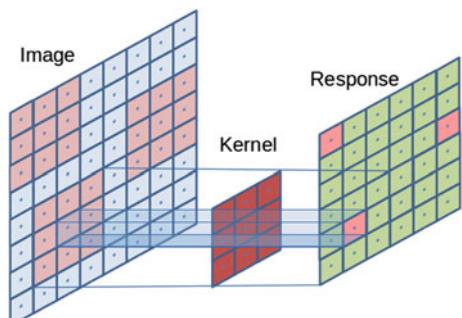
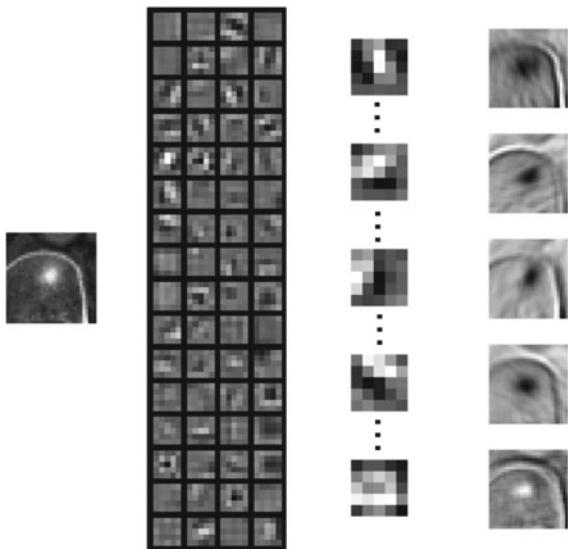


Fig. 3.2 Example of 64-trained filter kernels (*left*) of the first convolutional layer which are used for feature extraction (here for detection of sclerotic bone lesions in CT). Some filter responses are shown on the after convolution with the trained kernels. One can see learned kernels for complex higher order gradients, similar to blobness or difference of Gaussian filters. See [20] for more examples



implementations (*cuda-convnet*¹) by Krizhevsky et al. [14, 43] including efficient GPU acceleration and the DropConnect addition by [42]. In all experiments, our ConvNets are trained using stochastic gradient descent with momentum for 700-300-100-100 epochs on mini-batches of 64-64-32-16 images, following the suggestions of [42] for the CIFAR-10 data set (initial learning rate of 0.001 with the default weight decay). In each case, the data set mean image is subtracted from each training and testing image fed to the ConvNet (Table 3.1).

3.3.2 A 2D or 2.5D Approach for Applying ConvNets to CADe

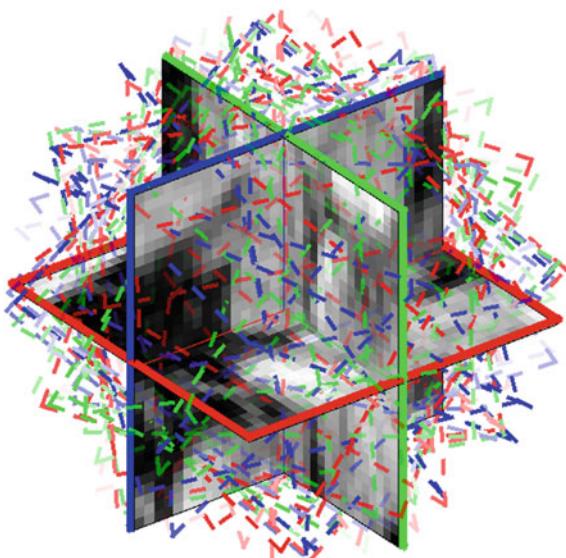
It depends on the imaging data whether a two-dimensional (2D) or two-and-a-half-dimensional (2.5D) decompositional approach is more suited to sample the image around CADe candidate locations for subsequent ConvNet classification (Fig. 3.3). In high-resolution 3D data, we take volumes-of-interest (VOIs) to extract 2.5D orthogonal planes within a VOI. These planes can be randomly orientated as explained later. However, low inter-slice resolution can limit the amount of 3D information visible in z-direction. Hence, 2D regions of interest (ROIs) can be extracted in random orientations. The location of both VOI and ROI regions can be obtained by some form of candidate generation process. Candidate generation should require close to 100% sensitivity as it will limit the sensitivity of the whole CADe system. FP rates should lie within a reasonable range of \sim 40 to \sim 150 per patient or volume. In our

¹<https://code.google.com/p/cuda-convnet>.

Table 3.1 The applied ConvNet architecture to 2.5D inputs detailing the number of filters, kernel sizes and output sizes across different layers. Note, we use overlapping kernels with stride 2 during max-pooling

Layer	Type	Parameters	Filters/Neurons	Output
0	Input			$32 \times 32 \times 3$
1	Cropping	Translations		$24 \times 24 \times 3$
2	Convolution	5×5 kernels	64	$24 \times 24 \times 3$
3	Max-pooling	3×3 kernels		$12 \times 12 \times 3$
4	Convolution	5×5 kernels	64	$12 \times 12 \times 3$
5	Max-pooling	3×3 kernels		$6 \times 6 \times 3$
6	Locally connected	3×3 kernels	64	$6 \times 6 \times 3$
7	Locally connected	3×3 kernels	32	$6 \times 6 \times 3$
8	Fully connected (DropConnect)		512	2
9	Fully connected		2	2
10	Softmax			2

Fig. 3.3 2.5D random view aggregation: CADe locations can be either observed as 2D image patches or using a 2.5D approach, that samples the image using three orthogonal views (shown by red, green, and blue borders). Here, a lymph node in CT is shown as the input to our method



experiments, we will show results including both extremes with lymph node candidates having around $40 \sim 60$ FPs/vol. and colonic polyps ~ 150 FPs/patient. Many existing CADe systems can deliver these specifics [4, 31–33]. Hence, our proposed FP reduction system using ConvNets could be useful in many applications.

3.3.3 Random View Aggregation

ConvNets have been described as being very “data hungry”. Even though a high-resolution CT scan have millions of voxels. FP locations of modern CADe systems should be less than a couple hundreds per case. Hence, we propose a simple and efficient way of increase both variation and number of training samples. Larger training data sets also reduce the changes of overfitting. This can be achieved by data augmentation such as translation and mirroring of images in the 2D case [14, 24, 25]. However, in volumetric data, multiple 2D or 2.5D observations per ROI or VOI can be generated by random translation, rotation and scaling within the 3D image space [44]. We perform N_t translations along a random vector v , N_r random rotation by $\alpha = [0^\circ, \dots, 360^\circ]$ around a ROI’s (translated) center coordinate, and N_s different physical scales s by changing the edge length of a local ROI (in our case, ROI/VOI have squared/cubed shapes). Note, that we keep the same number of pixels/voxels while resampling the different scales by adjusting the physical pixel sizes accordingly. In the case of limited 3D resolution (slice thicknesses of 5 mm or more), translations and rotations are just applied within the axial plane (2D). In total, this procedure will produce $N = N_s \times N_t \times N_r$ random observations of each ROI.

This $N \times$ increase of data samples will improve the ConvNets training and its ability to generalize to unseen cases as we will show in the results. Furthermore, we can apply the same strategy in testing unseen cases and aggregate ConvNet predictions $\{P_1(x), \dots, P_N\}$ at N random observations. We will show that simple averaging the ConvNet scores as in Eq. 3.1 will increase the overall performance of the system. Here, $P_i(x)$ is the classification score of one ConvNet prediction:

$$p(x|\{P_1(x), \dots, P_N(x)\}) = \frac{1}{N} \sum_{i=1}^N P_i(x). \quad (3.1)$$

3.3.4 Candidate Generation

In general, our FP reduction method can work well with any candidate generation system that runs at high sensitivity and reasonable low FP rates. We can use our ground truth data set to label each of N observations on candidate as ‘positive’ or ‘negative’ depending on whether it is on a true lesion (object of interest) or not. All labeled observations can then be used to train the ConvNets in a fully supervised fashion, typically using stochastic gradient descent on minibatches in order to minimize some loss function (see Sect. 3.3.1). In this paper, we leverage on three existing CADe systems with suitable performances for our FP reduction approach: [4] for *sclerotic bone lesion* detection, [31, 32] for *lymph nodes* detection, and [33] for *colonic polyps*.

3.4 Results

3.4.1 Computer-Aided Detection Data Sets

We chose three radiological data sets for different clinical applications of CADe, compromising sclerotic metastases detection in imaging of the spine, and for cancer monitoring and screening, detection of lymph nodes and detection of colonic polyps. The data sets also exhibit very different numbers of patients with 59, 176 (86 abdominal, 90 mediastinal) and 1,186 patients per data set respectively. This illustrates the ability of ConvNets to scale to different data set sizes, even if the data set is relatively small as in the sclerotic metastases case (59)—and large data sets of over 1,000 patients as in the case of colonic polyps (1,186). See Table 3.2 for further information on the patient populations. In the case of sclerotic metastases and lymph nodes, we sampled $N = 100$ ($N_t = 5$, $N_r = 5$ with $\alpha = [0^\circ, \dots, 360^\circ]$, and $N_s = 4$ with $s = [30, 35, 40, 45]$ mm) ROIs/VOIs (see Fig. 3.3) around each candidate coordinated given by the prior CADe systems [4, 31–33]. Due to the much larger data set size in the case of colonic polyps, we chose $N = 40$ ($N_s = 4$, $N_t = 2$ and $N_r = 5$), keeping s the same. The choice of s is important to cover the average dimensions of the lesions/objects of interest (i.e. bone metastases, lymph nodes, or colonic polyps), and to include some context area which might be useful for classification. The random translations were limited up to a maximum displacement of 3 mm in all cases. Each ROI, or VOI was sampled at 32×32 pixels for each channel (Table 3.3).

Typical training times for 1200 optimization epochs on a NVIDIA GeForce GTX TITAN (6GB memory) were 9–12 h for the lymph node data set, 12–15 h for the bone lesions data set, and 37 h for the larger colonic polyps data set. We used unit Gaussian random parameter initializations as in [42] in all cases. In testing, computing ConvNet scores on $N = 100$ 2D or 2.5D image patches at each ROI/VOI takes circa 1–5 min on a CT volume. For more detailed information about the performed experiments and results, we refer the reader to [20] (Fig. 3.4).

Table 3.2 CADe data sets used for evaluation: sclerotic metastases, lymph nodes, colonic polyps. Total/mean (target) lesion numbers, total true positive (TP) and false positive (FP) candidate numbers are stated. Note that one target can have several TP detections (see [20] for more detail)

Dataset	# Patients	# Targets	# TP	# FP	# Mean targets	# Mean candidates
Sclerotic lesions	59	532	935	3,372	9.0	73.0
Lymph nodes	176	983	1,966	6,692	5.6	49.2
Colonic polyps	1,186	252	468	174,301	0.2	147.4

Table 3.3 CADe performance with ConvNet Integration: previous¹ CADe performance compared to ConvNet² performance at the 3 FPs/patient rate (see [20] for more detailed experiments)

Dataset	Sensitivity ¹ (%)	Sensitivity ² (%)	AUC ¹ (%)	AUC ² (%)
Sclerotic lesions	57	70	n/a	0.83
Lymph nodes	43	77	0.76	0.94
Colonic polyps($>=6\text{ mm}$)	58	75	0.79	0.82
Colonic polyps($>=10\text{ mm}$)	92	98	0.94	0.99

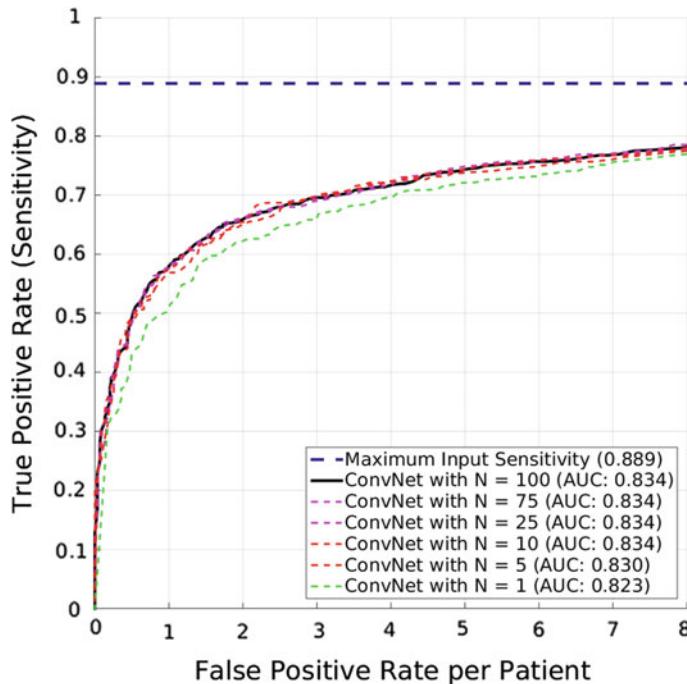


Fig. 3.4 Free Response Operating Characteristic (FROC) curve of *sclerotic bone lesion* detection [20]

3.5 Discussion and Conclusions

This work and many others (e.g., [18, 19, 45]) show the value of deep ConvNets for medical image analysis and the efficient implementation within existing computer-aided detection (CADe) frameworks. We showed marked improvements within three CADe applications, i.e., bone lesions, enlarged lymph nodes, and colonic polyps in CT (Fig. 3.5).

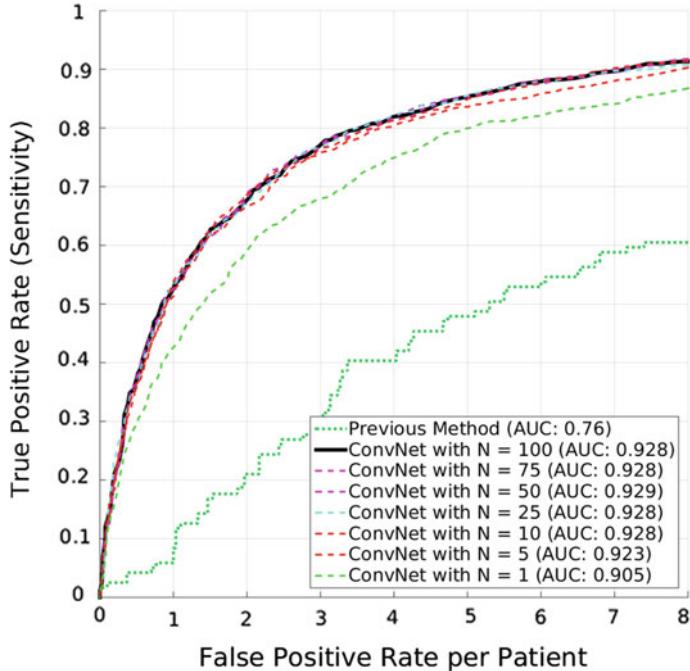


Fig. 3.5 Free Response Operating Characteristic (FROC) curve of *lymph node* detection [20]

ConvNets can be applied to 3D medical imaging applications using the standard architectures from computer vision [14, 42] with efficient 2.5D resampling of the 3D space, but applying 2D convolutions within the ConvNet architecture. Recent work that explores the direct application of 3D convolutional filters within the ConvNet architecture also shows promise [23, 45, 46]. It has to be established whether 2D or 3D ConvNet implementations are more suited for certain tasks. There is some evidence that ConvNet representations with direct 3D input suffer from the *curse-of-dimensionality* and are more prone to overfitting [20]. Volumetric object detection might require more training data and might suffer from scalability issues when full 3D data augmentation is not feasible. However, proper hyper-parameter tuning of the ConvNet architecture and enough training data (including data augmentation) might help eliminate these problems. In the mean time, random 2.5D resampling (as proposed here) might be an very efficient (computationally less expensive) way of diminishing the curse-of-dimensionality and to artificially increase the variation of training data. Furthermore, we showed that averaging scores of 2.5D observations can markedly improve robustness and stability of the overall CADe system (Sect. 3.4).

On another note, 2.5D (three-channel input ConvNets) have the advantage that pre-trained ConvNets which are trained on much larger available data bases of natural images (e.g. *ImageNet*) can be used. It has been shown that transfer learning is a viable approach when the medical imaging data set size is limited [18, 19]. There is

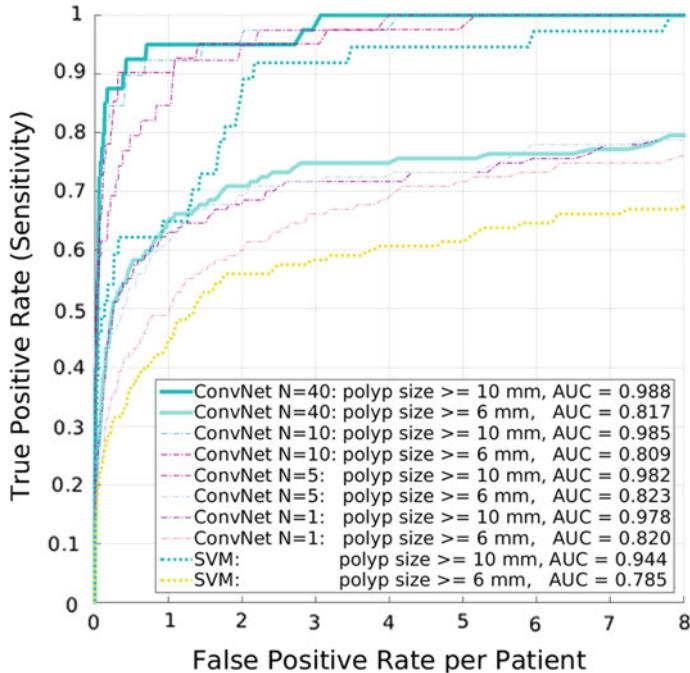


Fig. 3.6 Free Response Operating Characteristic (FROC) curve of *colonic polyp* detection [20]

evidence that even larger and deeper ConvNets perform better on classification tasks [16, 17, 47], however, even more training data is needed. In this case, the application of these modern networks to the medical imaging domain might especially benefit from pre-training [18] (Fig. 3.6).

In conclusion, the proposed 2D and 2.5D random aggregation of ConvNet scores is a promising approach for many CADe applications in medical imaging.

Acknowledgements This work was supported by the Intramural Research Program of the NIH Clinical Center.

References

- Organization, WH (2014) Cancer Fact sheet N297. WHO
- Msaouel P, Pissimisis N, Halapas A, Koutsilieris M (2008) Mechanisms of bone metastasis in prostate cancer: clinical implications. Best Pract Res Clin Endocrinol Metab 22(2):341–355
- Wiese T, Yao J, Burns JE, Summers RM (2012) Detection of sclerotic bone metastases in the spine using watershed algorithm and graph cut. In: SPIE medical imaging, p 831512
- Burns JE, Yao J, Wiese TS, Muñoz HE, Jones EC, Summers RM (2013) Automated detection of sclerotic metastases in the thoracolumbar spine at CT. Radiology 268(1):69–78

5. Hammon M, Dankerl P, Tsymbal A, Wels M, Kelm M, May M, Suehling M, Uder M, Cavallaro A (2013) Automatic detection of lytic and blastic thoracolumbar spine metastases on computed tomography. *Eur Radiol* 23(7):1862–1870
6. Seff A, Lu L, Cherry KM, Roth HR, Liu J, Wang S, Hoffman J, Turkbey EB, Summers RM (2014) 2D view aggregation for lymph node detection using a shallow hierarchy of linear classifiers. In: MICCAI. Springer, Berlin, pp 544–552
7. Toews M, Arbel T (2007) A statistical parts-based model of anatomical variability. *IEEE Trans Med Imaging* 26(4):497–508
8. Wu D, Lu L, Bi J, Shinagawa Y, Boyer K, Krishnan A, Salganicoff M (2010) Stratified learning of local anatomical context for lung nodules in CT images. In: 2010 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 2791–2798
9. Summers RM, Jerebko AK, Franaszek M, Malley JD, Johnson CD (2002) Colonic polyps: complementary role of computer-aided detection in CT colonography. *Radiology* 225(2):391–399
10. Ravesteijn V, Wijk C, Vos F, Truyen R, Peters J, Stoker J, Vliet L (2010) Computer aided detection of polyps in CT colonography using logistic regression. *IEEE Trans Med Imaging* 29(1):120–131
11. van Ginneken B, Setio A, Jacobs C, Ciompi F (2015) Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans. In: 2011 IEEE international symposium on biomedical imaging: from nano to macro. IEEE, pp 286–289
12. Firmino M, Moraes AH, Mendoza RM, Dantas MR, Hekis HR, Valentim R (2014) Computer-aided detection system for lung cancer in computed tomography scans: review and future prospects. *Biomed Eng Online* 13(1):41
13. Cheng HD, Cai X, Chen X, Hu L, Lou X (2003) Computer-aided detection and classification of microcalcifications in mammograms: a survey. *Pattern Recognit* 36(12):2967–2991
14. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. In: NIPS
15. LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD (1989) Backpropagation applied to handwritten zip code recognition. *Neural comput* 1(4)
16. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
17. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. [arXiv:1512.03385](https://arxiv.org/abs/1512.03385)
18. Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans Med Imaging* 35(5):1285–1298
19. Tajbakhsh N, Shin JY, Gurudu SR, Hurst RT, Kendall CB, Gotway MB, Liang J (2016) Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE Trans Med Imaging* 35(5):1299–1312
20. Roth HR, Lu L, Liu J, Yao J, Seff A, Cherry K, Kim L, Summers RM (2016) Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE Trans Med Imaging* 35(5):1170–1181
21. Jones N (2014) Computer science: the learning machines. *Nature* 505(7482):146–148
22. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):436–444
23. Turaga SC, Murray JF, Jain V, Roth F, Helmstaedter M, Briggman K, Denk W, Seung HS (2010) Convolutional networks can learn to generate affinity graphs for image segmentation. *Neural Comput* 22(2)
24. Cireşan DC, Giusti A, Gambardella LM, Schmidhuber J (2013) Mitosis detection in breast cancer histology images with deep neural networks. In: MICCAI
25. Ciresan D, Giusti A, Gambardella LM, Schmidhuber J (2012) Deep neural networks segment neuronal membranes in electron microscopy images. In: Advances in neural information processing systems, pp 2843–2851
26. Prasoon A, Petersen K, Igel C, Lauze F, Dam E, Nielsen M (2013) Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. In: MICCAI

27. Roth HR, Lu L, Seff A, Cherry K, Hoffman J, Wang S, Liu J, Turkbey E, Summers RM (2014) A new 2.5D representation for lymph node detection using random sets of deep convolutional neural network observations. In: Golland P, Hata N, Barillot C, Hornegger J, Howe R (eds) Medical image computing and computer-assisted intervention MICCAI 2014, vol 8673. Lecture Notes in Computer Science. Springer International Publishing, Berlin, pp 520–527
28. Roth H, Yao J, Lu L, Steiger J, Burns J, Summers R (2015) Detection of sclerotic spine metastases via random aggregation of deep convolutional neural network classifications. In: Yao J, Glocker B, Klinder T, Li S (eds) Recent advances in computational methods and clinical applications for spine imaging, vol 20. Lecture Notes in Computational Vision and Biomechanics. Springer International Publishing, Berlin, pp 3–12
29. Li Q, Cai W, Wang X, Zhou Y, Feng D.D, Chen, M (2014) Medical image classification with convolutional neural network. In: ICARCV
30. Tajbakhsh N, Gotway MB, Liang J (2015) Computer-aided pulmonary embolism detection using a novel vessel-aligned multi-planar image representation and convolutional neural networks. In: Medical image computing and computer-assisted intervention–MICCAI 2015. Springer International Publishing, Berlin, pp 62–69
31. Cherry KM, Wang S, Turkbey EB, Summers RM (2014) Abdominal lymphadenopathy detection using random forest. SPIE Med Imaging
32. Liu J, Zhao J, Hoffman J, Yao J, Zhang W, Turkbey EB, Wang S, Kim C, Summers RM (2014) Mediastinal lymph node detection on thoracic CT scans using spatial prior from multi-atlas label fusion. SPIE Med Imaging 43(7):4362
33. Summers RM, Yao J, Pickhardt PJ, Franaszek M, Bitter I, Brickman D, Krishna V, Choi JR (2005) Computed tomographic virtual colonoscopy computer-aided polyp detection in a screening population. Gastroenterology 129(6):1832–1844
34. Barbu A, Bogoni L, Comaniciu D (2006) Hierarchical part-based detection of 3D flexible tubes: application to CT colonoscopy. In: Larsen R, Nielsen M, Sporring J (eds) Medical image computing and computer-assisted intervention MICCAI, (2), pp 462–470
35. Lu L, Barbu A, Wolf M, Liang J, Bogoni L, Salganicoff M, Comaniciu D (2008) Simultaneous detection and registration for ileo-cecal valve detection in 3d CT colonography. In: Proceedings of european conference on computer vision, (4), pp 465–478
36. Lu L, Wolf M, Liang J, Dundar M, Bi J, Salganicoff M (2009) A two-level approach towards semantic colon segmentation: Removing extra-colonic findings. In: Medical image computing and computer-assisted intervention MICCAI, (1), pp 1009–1016
37. Yao J, Li J, Summers RM (2009) Employing topographical height map in colonic polyp measurement and false positive reduction. Pattern Recognit 42(6):1029–1040
38. Slabaugh G, Yang X, Ye X, Boyes R, Beddoe G (2010) A robust and fast system for CTC computer-aided detection of colorectal lesions. Algorithms 3(1):21–43
39. Lu L, Devarakota P, Vikal S, Wu D, Zheng Y, Wolf M (2014) Computer aided diagnosis using multilevel image features on large-scale evaluation. In: Medical computer vision. Large data in medical imaging. Springer, Berlin, pp 161–174
40. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR (2012) Improving neural networks by preventing co-adaptation of feature detectors. [arXiv:1207.0580](https://arxiv.org/abs/1207.0580)
41. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958
42. Wan L, Zeiler M, Zhang S, Cun YL, Fergus R (2013) Regularization of neural networks using dropconnect. In: Proceedings of the international conference on machine learning (ICML-13)
43. Krizhevsky A (2014) One weird trick for parallelizing convolutional neural networks. [arXiv:1404.5997](https://arxiv.org/abs/1404.5997)
44. Göktürk SB, Tomasi C, Acar B, Beaulieu CF, Paik DS, Jeffrey RB, Yee J, Napel Y (2001) A statistical 3-d pattern processing method for computer-aided detection of polyps in CT colonography. IEEE Trans Med Imaging 20:1251–1260
45. Dou Q, Chen H, Yu L, Zhao L, Qin J, Wang D, Mok VC, Shi L, Heng PA (2016) Automatic detection of cerebral microbleeds from MR images via 3d convolutional neural networks. IEEE Trans Med Imaging 35(5):1182–1195

46. Kamnitsas K, Ledig C, Newcombe VF, Simpson JP, Kane AD, Menon DK, Rueckert D, Glocker B (2016) Efficient multi-scale 3d CNN with fully connected CRF for accurate brain lesion segmentation. [arXiv:1603.05959](https://arxiv.org/abs/1603.05959)
47. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2014) Going deeper with convolutions. CoRR [arXiv:1409.4842](https://arxiv.org/abs/1409.4842)

Chapter 4

Robust Landmark Detection in Volumetric Data with Efficient 3D Deep Learning

**Yefeng Zheng, David Liu, Bogdan Georgescu, Hien Nguyen
and Dorin Comaniciu**

Abstract Recently, deep learning has demonstrated great success in computer vision with the capability to learn powerful image features from a large training set. However, most of the published work has been confined to solving 2D problems, with a few limited exceptions that treated the 3D space as a composition of 2D orthogonal planes. The challenge of 3D deep learning is due to a much larger input vector, compared to 2D, which dramatically increases the computation time and the chance of over-fitting, especially when combined with limited training samples (hundreds to thousands), typical for medical imaging applications. To address this challenge, we propose an efficient and robust deep learning algorithm capable of full 3D detection in volumetric data. A two-step approach is exploited for efficient detection. A shallow network (with one hidden layer) is used for the initial testing of all voxels to obtain a small number of promising candidates, followed by more accurate classification with a deep network. In addition, we propose two approaches, i.e., separable filter decomposition and network sparsification, to speed up the evaluation of a network. To mitigate the over-fitting issue, thereby increasing detection robustness, we extract small 3D patches from a multi-resolution image pyramid. The deeply learned image features are further combined with Haar wavelet-like features to increase the detection accuracy. The proposed method has been quantitatively evaluated for carotid artery bifurcation detection on a head-neck CT dataset from 455 patients. Compared to the state of the art, the mean error is reduced by more than half, from 5.97 mm to 2.64 mm, with a detection speed of less than 1 s/volume.

4.1 Introduction

An anatomical landmark is a biologically meaningful point on an organism, which can be easily distinguished from surrounding tissues. Normally, it is consistently present across different instances of the same organism so that it can be used to

Y. Zheng (✉) · D. Liu · B. Georgescu · H. Nguyen · D. Comaniciu
Medical Imaging Technologies, Siemens Healthcare, Princeton, NJ, USA
e-mail: yefeng.zheng@siemens.com

establish anatomical correspondence within the population. There are many applications of automatic anatomical landmark detection in medical image analysis. For example, landmarks can be used to align an input volume to a canonical plane on which physicians routinely perform diagnosis and quantification [1, 2]. A detected vascular landmark provides a seed point for automatic vessel centerline extraction and lumen segmentation [3, 4]. For a nonrigid object with large variation, a holistic detection may not be robust. Aggregation of the detection results of multiple landmarks on the object may provide a more robust solution [5]. In some applications, the landmarks themselves provide important measurements for disease quantification and surgical planning (e.g., the distance from coronary ostia to the aortic hinge plane is a critical indicator whether the patient is a good candidate for transcatheter aortic valve replacement [6]).

Various landmark detection methods have been proposed in the literature. Most of the state-of-the-art algorithms [1–6] apply machine learning (e.g., support vector machines, random forests, or boosting algorithms) on a set of handcrafted image features (e.g., SIFT features or Haar wavelet-like features). However, in practice, we found some landmark detection problems (e.g., carotid artery bifurcation landmarks in this work) are still too challenging to be solved with the current technology.

Deep learning [7] has demonstrated great success in computer vision with the capability to learn powerful image features (either supervised or unsupervised) from a large training set. Recently, deep learning has been applied in many medical image analysis problems, including body region recognition [8], cell detection [9], lymph node detection [10], organ detection/segmentation [11, 12], cross-modality registration [13], and 2D/3D registration [14]. On all these applications, deep learning outperforms the state of the art.

However, several challenges are still present in applying deep learning to 3D landmark detection. Normally, the input to a neural network classifier is an image patch, which increases dramatically in size from 2D to 3D. For example, a patch of 32×32 pixels generates an input of 1024 dimensions to the classifier. However, a $32 \times 32 \times 32$ 3D patch contains 32,768 voxels. Such a big input feature vector creates several challenges. First, the computation time of a deep neural network is often too slow for a real clinical application. The most widely used and robust approach for object detection is the *sliding window* based approach, in which the trained classifier is tested on each voxel in the volume. Evaluating a deep network on a large volume may take several minutes. Second, as a rule of thumb, a network with a bigger input vector requires more training data. With enough training samples (e.g., over 10 million in ImageNet), deep learning has demonstrated impressive performance gain over other methods. However, the medical imaging community is often struggling with limited training samples (often in hundreds or thousands) due to the difficulty to generate and share images. Several approaches can tackle or at least mitigate the issue of limited training samples. One approach is to reduce the patch size. For example, if we reduce the patch size from $32 \times 32 \times 32$ voxels to $16 \times 16 \times 16$, we can reduce the input dimension by a factor of eight. However, a small patch may not contain enough information for classification. Alternatively, instead of sampling a 3D patch, we can sample on three orthogonal planes [15] or even a 2D patch with a random

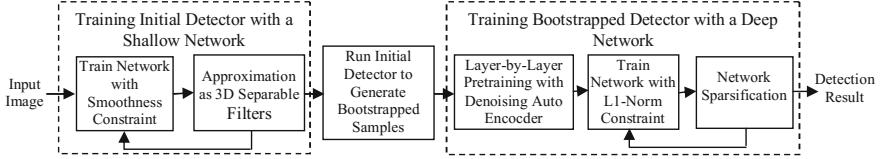


Fig. 4.1 Training procedure of the proposed deep network based 3D landmark detection method

orientation [10]. Although they can effectively reduce the input dimension, there is a concern on how much 3D information is contained in 2D planes.

In this work we tackle the above challenges in the application of deep learning for 3D anatomical structure detection (focusing on landmarks). Our approach significantly accelerates the detection speed, resulting in an efficient method that can detect a landmark in less than one second. We apply a two-stage classification strategy (as shown in Fig. 4.1). In the first stage, we train a shallow network with only one small hidden layer (e.g., with 64 hidden nodes). This network is applied to test all voxels in the volume in a sliding window process to generate 2000 candidates for the second-stage classification. The second network is much bigger with three hidden layers (each has 2000 nodes) to obtain more discriminative power. Such a cascaded classification approach has been widely used in object detection to improve detection efficiency and robustness.

In this work we propose two techniques to further accelerate the detection speed: separable filter approximation for the first-stage classifier and network sparsification for the second-stage classifier. The weights of a node in the first hidden layer are often treated as a filter (3D in this case). The response of the first hidden layer over the volume can be calculated as a convolution with the filter. Here, a neighboring patch is shifted by only one voxel; however, the response needs to be recalculated from scratch. In this work we approximate the weights as separable filters using tensor decomposition. Therefore, a direct 3D convolution is decomposed as three one-dimensional convolutions along the x , y , and z axis, respectively. Previously, such approximation has been exploited for 2D classification problems [16, 17]. However, in 3D, the trained filters are more difficult to be approximated as separable filters. We propose a new training cost function to enforce smoothness of the filters so that they can be approximated with high accuracy. The second big network only applies on a small number of candidates that have little correlation. Separable filter approximation does not help to accelerate classification. However, many weights in a big network are close to zero. We propose to add L1-norm regularization to the cost function to drive majority of the weights (e.g., 90%) to zero, resulting in a sparse network with increased classification efficiency without deteriorating accuracy.

The power of deep learning is on the automatic learning of a hierarchical image representation (i.e., image features). Instead of using the trained network as a classifier, we can use the responses at each layer (including the input layer, all hidden layers, and the output layer) as features and feed them into other state-of-the-art classifiers (e.g., boosting). After years of feature engineering, some handcrafted features

have considerable discriminative power for some applications and they may be complimentary to deeply learned features. In this work we demonstrate that combining deeply learned features and Haar wavelet-like features, we can reduce the detection failures.

The remainder of this chapter is organized as follows. In Sect. 4.2 we present a new method to train a shallow network with separable filters, which are efficient in a sliding window based detection scheme to prune the landmark candidates. Section 4.3 describes a sparse network that can effectively accelerate the evaluation of a deep network, which is used to further test the preserved landmark candidates. We present a feature fusion approach in Sect. 4.4 to combine Haar wavelet-like features and deeply learned features to improve the landmark detection accuracy. Experiments on a large dataset in Sect. 4.5 demonstrate the robustness and efficiency of the proposed method. This chapter concludes with Sect. 4.6. Please note, an early version of this work was published in [18].

4.2 Training Shallow Network with Separable Filters

A fully connected multilayer perceptron (MLP) neural network is a layered architecture. Suppose the input is a n_0 -dimensional vector $[X_1^0, X_2^0, \dots, X_{n_0}^0]$. The response of a node X_j^1 of the first hidden layer is

$$X_j^1 = g\left(\sum_{i=1}^{n_0} W_{i,j}^0 X_i^0 + b_j^0\right), \quad (4.1)$$

for $j = 1, 2, \dots, n_1$ (n_1 is the number of nodes in the first hidden layer). Here, $W_{i,j}^0$ is a weight; b_j^0 is a bias term; And, $g(\cdot)$ is a nonlinear function, which can be sigmoid, hypo-tangent, restricted linear unit (ReLU), or other forms. In this work we use the sigmoid function

$$g(x) = \frac{1}{1 + e^{-x}}, \quad (4.2)$$

which is the most popular nonlinear function. If we denote $\mathbf{X}^0 = [X_1^0, \dots, X_{n_0}^0]^T$ and $\mathbf{W}_j^0 = [W_{1,j}^0, \dots, W_{n_0,j}^0]^T$, Eq. (4.1) can be rewritten as $X_j^1 = g((\mathbf{W}_j^0)^T \mathbf{X}^0 + b_j^0)$. Multiple layers can be stacked together using Eq. (4.1) as a building block. For a binary classification problem as this work, the output of the network can be a single node \hat{X} . Suppose there are L hidden layers, the output of the neural network is $\hat{X} = g((\mathbf{W}^L)^T \mathbf{X}^L + b^L)$. During network training, we require the output to match the class label Y (with 1 for the positive class and 0 for negative) by minimizing the squared error $E = ||Y - \hat{X}||^2$.

In object detection using a sliding window based approach, for each position hypothesis, we crop an image patch (with a predefined size) centered at the position

hypothesis. We then serialize the patch intensities into a vector as the input to calculate response \hat{X} . After testing a patch, we shift the patch by one voxel (e.g., to the right) and repeat the above process again. Such a naive implementation is time consuming. Coming back to Eq.(4.1), we can treat the weights of a node in the first hidden layer as a filter. The first term of the response is a dot-product of the filter and the image patch intensities. Shifting the patch over the whole volume is equivalent to convolution using the filter. Therefore, alternatively, we can perform convolution using each filter \mathbf{W}_j^0 for $j = 1, 2, \dots, n_1$ and cache the response maps. During object detection, we can use the cached maps to retrieve the response of the first hidden layer.

Although such an alternative approach does not save computation time, it gives us a hint for speedup. With a bit abuse of symbols, suppose $\mathbf{W}_{x,y,z}$ is a 3D filter with size $n_x \times n_y \times n_z$. Let us further assume that $\mathbf{W}_{x,y,z}$ is separable, which means we can find three one-dimensional vectors, $\mathbf{W}_x, \mathbf{W}_y, \mathbf{W}_z$, such that

$$\mathbf{W}_{x,y,z}(i, j, k) = \mathbf{W}_x(i) \cdot \mathbf{W}_y(j) \cdot \mathbf{W}_z(k) \quad (4.3)$$

for any $i \in [1, n_x]$, $j \in [1, n_y]$, and $k \in [1, n_z]$. The convolution of the volume with $\mathbf{W}_{x,y,z}$ is equivalent to three sequential convolutions with $\mathbf{W}_x, \mathbf{W}_y$, and \mathbf{W}_z along its corresponding axis. Sequential convolution with one-dimensional filters is much more efficient than direct convolution with a 3D filter, especially for a large filter. However, in reality, Eq.(4.3) is just an approximation of filters learned by a neural network and such a rank-1 approximation is poor in general. In this work we search for S sets of separable filters to approximate the original filter as

$$\mathbf{W}_{x,y,z} \approx \sum_{s=1}^S \mathbf{W}_x^s \cdot \mathbf{W}_y^s \cdot \mathbf{W}_z^s. \quad (4.4)$$

Please note, with a sufficient number of separable filters (e.g., $S \geq \min\{n_x, n_y, n_z\}$), we can reconstruct the original filter perfectly.

To achieve detection efficiency, we need to cache $n_1 \times S$ filtered response maps. If the input volume is big (the size of a typical CT scan in our dataset is about 300 MB) and n_1 is relatively large (e.g., 64 or more), the cached response maps consume a lot of memory. Fortunately, the learned filters $\mathbf{W}_1^0, \dots, \mathbf{W}_{n_1}^0$ often have strong correlation (i.e., a filter can be reconstructed by a linear combination of other filters). We do not need to maintain a different filter bank for each \mathbf{W}_i^0 . The separable filters in reconstruction can be drawn from the same bank,

$$\mathbf{W}_i^0 \approx \sum_{s=1}^S c_{i,s} \cdot \mathbf{W}_x^s \cdot \mathbf{W}_y^s \cdot \mathbf{W}_z^s. \quad (4.5)$$

Here, $c_{i,s}$ is the combination coefficient, which is specific for each filter \mathbf{W}_i^0 . However, \mathbf{W}_x^s , \mathbf{W}_y^s , and \mathbf{W}_z^s are shared by all filters. Equation (4.5) is a rank- S decomposition of a 4D tensor $[\mathbf{W}_1^0, \mathbf{W}_2^0, \dots, \mathbf{W}_{n_1}^0]$, which can be solved using [19].

Using 4D tensor decomposition, we only need to convolve the volume S times (instead of $n_1 \cdot S$ times using 3D tensor decomposition) and cache S response maps. Suppose the input volume has $N_x \times N_y \times N_z$ voxels. For each voxel, we need to do $n_x n_y n_z$ multiplications using the original sliding window based approach. To calculate the response of a hidden layer with n_1 nodes, the total number of multiplications is $n_1 n_x n_y n_z N_x N_y N_z$. Using the proposed approach, to perform convolution with S set of separable filters, we need do $S(n_x + n_y + n_z) N_x N_y N_z$ multiplications. To calculate the response of n_1 hidden layer nodes, we need to combine the S responses using Eq. (4.5), resulting in $n_1 S N_x N_y N_z$ multiplications. The total number of multiplications is $S(n_x + n_y + n_z + n_1) N_x N_y N_z$. Suppose $S = 32$, $n_1 = 64$, the speedup is 62 times for a $15 \times 15 \times 15$ patch.

To achieve significant speedup and save memory footprint, we need to reduce S as much as possible. However, we found, with a small S (e.g., 32), it was more difficult to approximate 3D filters than 2D filters [16, 17]. Nonlinear functions $g(\cdot)$ are exploited in neural networks to bound the response to a certain range (e.g., $[0, 1]$ using the sigmoid function). Many nodes are saturated (with an output close to 0 or 1) and once a node is saturated, its response is not sensitive to the change of the weights. Therefore, a weight can take an extremely large value, resulting in a non-smooth filter. Here, we propose to modify the objective function to encourage the network to generate smooth filters

$$E = ||Y - \hat{X}||^2 + \alpha \sum_{i=1}^{n_1} ||\mathbf{W}_i^0 - \overline{\mathbf{W}}_i^0||^2. \quad (4.6)$$

Here, $\overline{\mathbf{W}}_i^0$ is the mean value of the weights of filter \mathbf{W}_i^0 . So, the second term measures the variance of the filter weights. Parameter α (often takes a small value, e.g., 0.001) keeps a balance between two terms in the objective function. The proposed smooth regularization term is different to the widely used L2-norm regularization, which is as follows

$$E = ||Y - \hat{X}||^2 + \alpha \sum_{j=1}^L \sum_{i=1}^{n_j} ||\mathbf{W}_i^0||^2. \quad (4.7)$$

The L2-norm regularization applies to all weights, while our regularization applies only to the first hidden layer. Furthermore, L2-norm regularization encourages small weights, therefore shrinks the capacity of the network; while our regularization encourages small variance of the weights.

The training of the initial shallow network detector is as follows (as shown in the left dashed box of Fig. 4.1). (1) Train a network using Eq. (4.6). (2) Approximate the learned filters using a filter bank with S ($S = 32$ in our experiments) sets of separable

filters to minimize the error of Eq.(4.5). The above process may be iterated a few times (e.g., three times). In the first iteration, the network weights and filter bank are initialized with random values. However, in the following iterations, they are both initialized with the optimal values from the previous iteration.

Previously, separable filter approximation has been exploited for 2D classification problems [16, 17]. We found 3D filters were more difficult to be approximated well with a small filter bank; therefore, we propose a new objective function to encourage the network to generate smooth filters for higher separability. Furthermore, unlike [17], we also iteratively retrain the network to compensate the loss of accuracy due to approximation.

4.3 Training Sparse Deep Network

Using a shallow network, we can efficiently test all voxels in the volume and assign a detection score to each voxel. After that, we preserve 2000 candidates with the largest detection scores. The number of preserved candidates is tuned to have a high probability to include the correct detection (e.g., hypotheses within one-voxel distance to the ground truth). However, most of the preserved candidates are still false positives. In the next step, we train a deep network to further reduce the false positives. The classification problem is now much tougher and a shallow network does not work well. In this work we use a big network with three hidden layers, each with 2000 nodes.

Even though we only need to classify a small number of candidates, the computation may still take some time since the network is now much bigger. Since the preserved candidates are often scattered over the whole volume, separable filter decomposition as used in the initial detection stage does not help to accelerate the classification. After checking the values of the learned weights of this deep network, we found most of weights were very small, close to zero. That means many connections in the network can be removed without sacrificing classification accuracy. Here, we apply L1-norm regularization to enforce sparse connection

$$E = ||Y - \hat{X}||^2 + \beta \sum_{j=1}^L \sum_{i=1}^{n_j} ||\mathbf{W}_i^j||. \quad (4.8)$$

Parameter β can be used to tune the number of zero weights. The higher β is, the more weights converge to zero. With a sufficient number of training epochs, part of weights converges exactly to zero. In practice, to speed up the training, we periodically check the magnitude of weights. The weights with a magnitude smaller than a threshold are set to zero and the network is refined again. In our experiments, we find that 90% of the weights can be set to zero after training, without deteriorating the classification accuracy. Thus, we can speed up the classification by roughly ten times.

The proposed acceleration technologies can be applied to different neural network architectures, e.g., a multilayer perceptron (MLP) and a convolutional neural network (CNN). In this work we use the MLP. While the shallow network is trained with back-propagation to directly minimize the objective function in Eq. (4.6), the deep network is pretrained using the denoising auto-encoder criterion [7] and then fine-tuned to minimize Eq. (4.8). The right dashed box of Fig. 4.1 shows the training procedure of the sparse deep network.

4.4 Robust Detection by Combining Multiple Features

To train a robust neural network based landmark detector on limited training samples, we have to control the patch size. The optimal patch size was searched and we found a size of $15 \times 15 \times 15$ achieved a good trade-off between detection speed and accuracy. However, a small patch has a limited field-of-view, thereby may not capture enough information for classification. In this work we extract patches on an image pyramid with multiple resolutions. A small patch in a low-resolution volume has a much larger field-of-view at the original resolution. To be specific, we build an image pyramid with three resolutions (1 mm, 2 mm, and 4-mm resolution, respectively). The intensities of patches from multiple resolutions are concatenated into a long vector to feed the network. As demonstrated in Sect. 4.5, a multi-resolution patch can improve the landmark detection accuracy.

Deep learning automatically learns a hierarchical representation of the input data. Representation at different hierarchical levels may provide complementary information for classification. Furthermore, through years' of feature engineering, some handcrafted image features can achieve quite reasonable performance on a certain task. Combining effective handcrafted image features with deeply learned hierarchical features may achieve even better performance than using them separately.

In this work we propose to use probabilistic boosting-tree (PBT) [20] to combine all features. A PBT is a combination of a decision tree and AdaBoost, by replacing a weak classification node in the decision tree with a strong AdaBoost classifier [21]. Our feature pool is composed of two types of features: Haar wavelet-like features (h_1, h_2, \dots, h_m) and neural network features r_i^j (where r_i^j is the response of node i at layer j). If $j = 0$, r_i^0 is an input node, representing the image intensity of a voxel in the patch. The last neural network feature is actually the response of the output node, which is the classification score by the network. This feature is the strongest feature and it is always the first selected feature by the AdaBoost algorithm.

Given 2000 landmark candidates generated by the first detection stage (Sect. 4.2), we evaluate them using the bootstrapped classifier presented in this section. We preserve 250 candidates with the highest classification score and then aggregate them into a single detection as follows. For each candidate we define a neighborhood, which is a $8 \times 8 \times 8$ mm³ box centered on the candidate. We calculate the total vote of each candidate as the summation of the classification score of all neighboring candidates. (The score of the current candidate is also counted since it is neighboring

to itself.) The candidate with the largest vote is picked and the final landmark position is the weighted average (according to the classification score) of all candidates in its neighborhood.

4.5 Experiments

In this section we validate the proposed method on carotid artery bifurcation detection. The carotid artery is the main vessel supplying oxygenated blood to the head and neck. The common carotid artery originates from the aortic arch and runs up toward the head before bifurcating to the external carotid artery (supplying blood to face) and internal carotid artery (supplying blood to brain). Examination of the carotid artery helps to assess the stroke risk of a patient. Automatic detection of this bifurcation landmark provides a seed point for centerline tracing and lumen segmentation, thereby making automatic examination possible. However, as shown in Fig. 4.2a, the internal/external carotid arteries further bifurcate to many branches and there are other vessels (e.g., vertebral arteries and jugular veins) present nearby, which may cause confusion to an automatic detection algorithm.

We collected a head-neck CT dataset from 455 patients. Each image slice has 512×512 pixels and a volume contains a variable number of slices (from 46 to 1181 slices). The volume resolution varies too, with a typical voxel size of $0.46 \times 0.46 \times 0.50 \text{ mm}^3$. To achieve a consistent resolution, we resample all input volumes to 1.0 mm . A fourfold cross validation is performed to evaluate the detection accuracy and determine the hyper parameters, e.g., the network size, smoothness constraint α in Eq. (4.6), sparsity constraint β in Eq. (4.8). There are two carotid arteries (left versus right) as shown in Fig. 4.2. Here, we report the bifurcation detection accuracy of the right carotid artery (as shown in Table 4.1) with different approaches. The detection accuracy of the left carotid artery bifurcation is similar.

The rough location of the carotid artery bifurcation can be predicted by other landmarks using a landmark network [22]. However, due to the challenge of the task, the prediction is not always accurate. We have to crop a box as large as $50 \times 50 \times 100 \text{ mm}^3$ around the predicted position to make sure the correct position of the carotid artery bifurcation is covered. To have a fair comparison with [4], in the following experiments, the landmark detection is constrained to this box for all compared methods.

For each approach reported in Table 4.1, we follow a two-step process by applying the first detector to reduce the number of candidates to 2000, followed by a bootstrapped detection to further reduce the number of candidates to 250. The final detection is picked from the candidate with the largest vote from other candidates.

The value of a CT voxel represents the attenuation coefficient of the underlying tissue to X-ray, which is often represented as a Hounsfield unit. The Hounsfield unit has a wide range from -1000 for air to 3000 for bones/metals and it is normally represented with a 12-bit precision. A carotid artery filled with contrasted agent

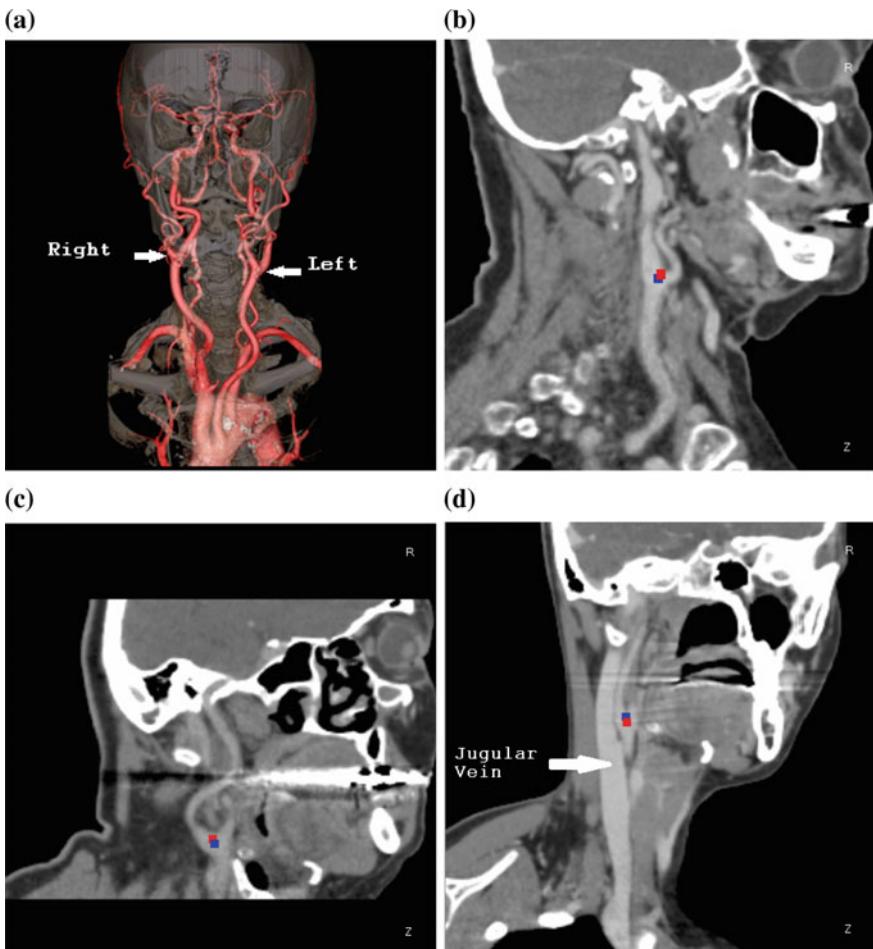


Fig. 4.2 Carotid artery bifurcation landmark detection in head-neck CT scans. **a** 3D visualization of carotid arteries with *white arrows* pointing to the *left* and *right* bifurcations (image courtesy of <http://blog.remakehealth.com/>). **b–d** A few examples of the *right* carotid artery bifurcation detection results with the ground truth labeled as *blue dots* and detected landmarks in *red*

Table 4.1 Quantitative evaluation of carotid artery bifurcation detection accuracy on 455 CT scans based on a fourfold cross validation. The errors are reported in millimeters

	Mean	Std	Median	80th Percentile
Haar + PBT	5.97	6.99	3.64	7.84
Neural network (Single resolution)	4.13	9.39	1.24	2.35
Neural network (Multi-resolution)	3.69	6.71	1.62	3.25
Network features + PBT	3.54	8.40	1.25	2.31
Haar + network + PBT	2.64	4.98	1.21	2.39

occupies only a small portion of the full Hounsfield unit range. Standard normalization methods of neural network training (e.g., linear normalization to [0, 1] using the minimum and maximum value of the input, or normalizing to zero-mean and unit-variance) do not work well for this application. In this work we use a window based normalization. Intensities inside the window of $[-24, 576]$ Hounsfield unit is linearly transformed to $[0, 1]$; Intensities less than -24 are truncated to 0; And, intensities higher than 576 are truncated to 1.

Previously, Liu et al. [4] used Haar wavelet-like features + boosting to detect vascular landmarks and achieved promising results. Applying this approach on our dataset, we achieve a mean error of 5.97 mm and the large mean error is caused by too many detection outliers. The neural network based approach can significantly improve the detection accuracy with a mean error of 4.13 mm using a $15 \times 15 \times 15$ patch extracted from a single resolution (1 mm). Using patches extracted from an image pyramid with three resolutions, we can further reduce the mean detection error to 3.69 mm. If we combine features from all layers of the network using the PBT, we achieve slightly better mean accuracy of 3.54 mm. Combining the deeply learned features and Haar wavelet-like features, we achieve the best detection accuracy with a mean error of 2.64 mm. We suspect that the improvement comes from the complementary information of the Haar wavelet-like features and neural network features. Figure 4.2 shows the detection results on a few typical datasets.

The proposed method is computationally efficient. Using the speedup technologies presented in Sects. 4.2 and 4.3, it takes 0.92 s to detect a landmark on a computer with a six-core 2.6 GHz CPU (without using GPU). For comparison, the computation time increases to 18.0 s if we turn off the proposed acceleration technologies (namely, separable filter approximation and network sparsification). The whole training procedure takes about 6 h and the sparse deep network consumes majority of the training time.

4.6 Conclusions

In this work we proposed 3D deep learning for efficient and robust landmark detection in volumetric data. We proposed two technologies to speed up the detection using neural networks, namely, separable filter decomposition and network sparsification. To improve the detection robustness, we exploit deeply learned image features trained on a multi-resolution image pyramid. Furthermore, we use the boosting technology to incorporate deeply learned hierarchical features and Haar wavelet-like features to further improve the detection accuracy. The proposed method is generic and can be retrained to detect other 3D landmarks or the center of organs.

References

1. Zhan Y, Dewan M, Harder M, Krishnan A, Zhou XS (2011) Robust automatic knee MR slice positioning through redundant and hierarchical anatomy detection. *IEEE Trans Med Imag* 30(12):2087–2100
2. Schwing AG, Zheng Y (2014) Reliable extraction of the mid-sagittal plane in 3D brain MRI via hierarchical landmark detection. In: Proceedings of the international symposium on biomedical imaging, pp 213–216
3. Zheng Y, Tek H, Funka-Lea G, Zhou SK, Vega-Higuera F, Comaniciu D (2011) Efficient detection of native and bypass coronary ostia in cardiac CT volumes: anatomical versus pathological structures. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 403–410
4. Liu D, Zhou S, Bernhardt D, Comaniciu D (2011) Vascular landmark detection in 3D CT data. In: Proceedings of the SPIE medical imaging, pp 1–7
5. Zheng Y, Lu X, Georgescu B, Littmann A, Mueller E, Comaniciu D (2009) Robust object detection using marginal space learning and ranking-based multi-detector aggregation: application to automatic left ventricle detection in 2D MRI images. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1343–1350
6. Zheng Y, John M, Liao R, Nottling A, Boese J, Kempfert J, Walther T, Brockmann G, Comaniciu D (2012) Automatic aorta segmentation and valve landmark detection in C-arm CT for transcatheter aortic valve implantation. *IEEE Trans Med Imaging* 31(12):2307–2321
7. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J Mach Learn Res* 11:3371–3408
8. Yan Z, Zhan Y, Peng Z, Liao S, Shinagawa Y, Metaxas DN, Zhou, XS (2015) Bodypart recognition using multi-stage deep learning. In: Proceedings of the information processing in medical imaging, pp 449–461
9. Liu F, Yang L (2015) A novel cell detection method using deep convolutional neural network and maximum-weight independent set. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 349–357
10. Roth HR, Lu L., Seff A, Cherry KM, Hoffman J, Wang S, Liu J, Turkbey E, Summers RM (2014) A new 2.5D representation for lymph node detection using random sets of deep convolutional neural network observations. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 520–527
11. Carneiro G, Nascimento JC, Freitas A (2012) The segmentation of the left ventricle of the heart from ultrasound data using deep learning architectures and derivative-based search methods. *IEEE Trans Image Process* 21(3):968–982
12. Ghesu FC, Krubasik E, Georgescu B, Singh V, Zheng Y, Hornegger J, Comaniciu D (2016) Marginal space deep learning: efficient architecture for volumetric image parsing. *IEEE Trans Med Imag* 35(5):1217–1228
13. Cheng X, Zhang L, Zheng Y (2016) Deep similarity learning for multimodal medical images. *Comput Methods Biomed Eng Imaging Vis* 4:1–5
14. Miao S, Wang ZJ, Zheng Y, Liao R (2016) Real-time 2D/3D registration via CNN regression. In: Proceedings of the IEEE international symposium on biomedical imaging, pp 1–4
15. Prasoon A, Petersen K, Igel C, Lauze F, Dam E, Nielsen M (2013) Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. In: Proceedings of the international conference on medical image computing and computer assisted intervention, vol 8150, pp 246–253
16. Rigamonti R, Sironi A, Lepetit V, Fua P (2013) Learning separable filters. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2754–2761
17. Denton E, Zaremba W, Bruna J, LeCun Y, Fergus R (2014) Exploiting linear structure within convolutional networks for efficient evaluation. In: Advances in neural information processing systems, pp 1–11

18. Zheng Y, Liu D, Georgescu B, Nguyen H, Comaniciu D (2015) 3D deep learning for efficient and robust landmark detection in volumetric data. In: Proceedings of the international conference medical image computing and computer assisted intervention, pp 565–572
19. Acar E, Dunlavy DM, Kolda TG (2011) A scalable optimization approach for fitting canonical tensor decompositions. *J Chemom* 25(2):67–86
20. Tu Z (2005) Probabilistic boosting-tree: learning discriminative methods for classification, recognition, and clustering. In: Proceedings of the international conference on computer vision, pp 1589–1596
21. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. *J Comput Syst Sci* 55(1):119–139
22. Liu D, Zhou S, Bernhardt D, Comaniciu D (2010) Search strategies for multiple landmark detection by submodular maximization. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2831–2838

Chapter 5

A Novel Cell Detection Method Using Deep Convolutional Neural Network and Maximum-Weight Independent Set

Fujun Liu and Lin Yang

Abstract Cell detection is an important topic in biomedical image analysis and it is often the prerequisite for the following segmentation or classification procedures. In this chapter, we propose a novel algorithm for general cell detection problem: First, a set of cell detection candidates is generated using different algorithms with varying parameters. Second, each candidate is assigned a score by a trained deep convolutional neural network (DCNN). Finally, a subset of best detection results are selected from all candidates to compose the final cell detection results. The subset selection task is formalized as a maximum-weight independent set problem, which is designed to find the heaviest subset of mutually nonadjacent nodes in a graph. Experiments show that the proposed general cell detection algorithm provides detection results that are dramatically better than any individual cell detection algorithm.

5.1 Introduction

Cell detection is an important topic in biomedical image analysis because it is often the first step for the following tasks, including cell counting, segmentation, and morphological analysis. Many automatic cell detection algorithms are proposed in recent literatures [1–3]. Parvin et al. proposed an iterative voting algorithm based on oriented kernels to localize cell centers, in which the voting direction and areas were dynamically updated within each iteration. In [2], a simple and reliable cell detector was designed based on a Laplacian of Gaussian filter. A learning-based cell detection algorithm was proposed in [3]. It used an efficient maximally stable extremal regions (MSER) detector [4] to find a set of nested candidate regions that will form a tree

F. Liu · L. Yang (✉)

Department of Electrical & Computer Engineering, Medford, MA, USA
e-mail: Lin.Yang@bme.ufl.edu

L. Yang

J. Crayton Pruitt Family Department of Biomedical Engineering, University of Florida,
Gainesville, FL, USA

graph. Then a nonoverlapping subset of those regions was selected for cell detection via dynamic programming.

All the methods reviewed above give good detection results under certain circumstances. However, in general, they all have some limitations. For example, both [1] and [2] are sensitive to the selection of proper cell diameter parameters. However, finding an appropriate parameter that works under all conditions is extremely difficult when the cells exhibit large size variations. In [3], the algorithm heavily depends on the quality of MSER detector that does not take advantage the prior cell shape information and the performance will deteriorate when the cells overlap with one another.

In this chapter, we propose a novel algorithm for general cell detection that does not require the fine tuning of parameters. First, a set of cell detection candidates is produced from different algorithms with varying parameters. Second, each candidate will be assigned a score using a trained deep convolutional neural network (DCNN) [5, 6]. Third, we will construct a weighted graph that has the detection candidates as nodes and the detection scores (DCNN outputs) as weights (an edge exists between two nodes if their corresponding detection results lie in the same cell). Finally, a subset of mutually nonadjacent graph nodes is chosen to maximize the sum of the weights of the selected nodes. An overview of the algorithm is shown in Fig. 5.1. The selection of the best subset is formulated as a maximum-weight independent set problem (MWIS). MWIS is a combinatorial optimization problem that has been successfully applied in clustering [7], segmentation [8], and tracking [9], etc.

To the best of our knowledge, this is the first work that formulates the general cell detection problem as a MWIS problem, and this is also the first work to introduce DCNN to provide weights to a graph for future combinational optimization.

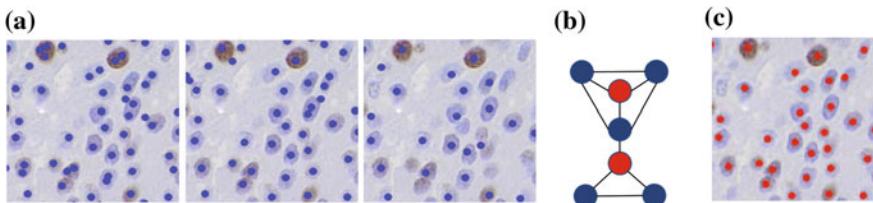


Fig. 5.1 An overview of the proposed general cell detection algorithm. **a** A set of detections candidates generated using multiple detection algorithms. Each candidate is marked with *blue dot*. **b** An undirected weighted graph was constructed from all detection candidates. The *red color* indicates the selected nodes. **c** The final cell detection results using the proposed method

5.2 Methodology

5.2.1 Cell Detection Using MWIS

A set of cell detection candidates (points), $P = \{p_1, \dots, p_n\}$, are first generated based on different cell detection algorithms with various parameters. An undirected and weighted graph, $G = (V, E, w)$, is constructed, where the node v_i corresponds to the i -th cell detection candidate p_i , E denotes undirected edges between nodes, and w_i denotes weight for the i -th node v_i . Two nodes v_i and v_j are adjacent, $(v_i, v_j) \in E$, if the Euclidean distance between their respective detection results p_i and p_j is smaller than a threshold λ . A node v_i will be assigned a larger weight value w_i if its corresponding detection result p_i is close to the real cell center, otherwise smaller weight will be assigned. After graph G is constructed, an optimal subset of V will be selected with the constraint that two nodes adjacent to each other will not be selected simultaneously. A subset is represented by an indicator vector $\mathbf{x} = \{x_1, \dots, x_i, \dots, x_n\}$, where $x_i \in \{0, 1\}$. $x_i = 1$ indicates that node v_i is in the subset, and $x_i = 0$ represents that v_i is not in the subset. This best subset selection is then formulated as finding the maximum-weight independent set (MWIS) \mathbf{x}^* .

$$\mathbf{x}^* = \arg \max_{\mathbf{x}} w^T \mathbf{x}, \text{ s. t. } \mathbf{x}^T \mathbf{A} \mathbf{x} = 0, \quad x_i \in \{0, 1\}, \quad (5.1)$$

where $\mathbf{A} = (a_{ij})_{n \times n}$ is the adjacent matrix, $a_{ij} = 1$ if $(v_i, v_j) \in E$ and $a_{ij} = 0$ otherwise. The diagonal elements of \mathbf{A} are zeros. The quadric constraints can be integrated into the object function to reformulate the optimization as

$$\mathbf{x}^* = \arg \max_{\mathbf{x}} \left(w^T \mathbf{x} - \frac{1}{2} \alpha \mathbf{x}^T \mathbf{A} \mathbf{x} \right), \quad \text{s. t. } x_i \in \{0, 1\}, \quad (5.2)$$

where α is a positive regularization parameter to encode the nonadjacent constraints in (5.1).

The MWIS optimization can be solved by some numerical approximation algorithms [8, 10]. In [10], the integer constraints in (5.2) are relaxed, and a graduated assignment algorithm iteratively maximizes a Taylor series expansion of the object function in (5.2) around the previous solution in the continuous domain. The relaxed continuous solution will then be binarized to obtain the discrete solution. This binarization procedure might lead to errors. In order to avoid this type of error, [8] directly seeks a discrete solution in each iteration in maximizing the Taylor series approximation. However, in this case the solution of (5.2) might not satisfy the nonadjacent constraints in (5.1). In our algorithm, unlike all the previous procedures, we propose to find the optimal results iteratively only in the solution space of (5.1).

Denote $f(\mathbf{x})$ as the objective function in (5.2), let $x^{(t)} \in \{0, 1\}^n$ denotes the current solution in the t -th iteration, each iteration consists of the following two steps in our algorithm.

Step 1: For any point $x \in \{0, 1\}^n$ in the neighborhood of $x^{(t)}$, we first find the first-order Taylor series approximation of $f(x)$ as

$$f(\mathbf{x}) \approx T(\mathbf{x}) = f(x^{(t)}) + (x - x^{(t)})^T(w - \alpha Ax^{(t)}) = x^T(w - \alpha Ax^{(t)}) + const, \quad (5.3)$$

where $const$ represents an item that does not depend on x . Define $y^{(t)}$ as the intermediate solution to (5.3), it can be computed by maximizing the approximation $T(x)$ as $y^{(t)} = \mathbb{1}(w - \alpha Ax^{(t)} \geq 0)$, where $\mathbb{1}(\cdot)$ is an indicator function.

Step 2: The solution of (5.3) might not satisfy the nonadjacent constraints listed in (5.1). If this is the case, we need to find a valid solution of (5.1) based on $y^{(t)}$. This is achieved by the following steps: (1) We first sort all the nodes based on their weights with a decreasing order. The nodes with $y^{(t)} = 1$ will be placed in front of the nodes that have $y^{(t)} = 0$. (2) The nodes are then selected from the front of the queue sequentially with a constraint that the picked node will not be adjacent to those that are already chosen.

After we find the valid solution, the $x^{(t+1)}$ in the solution space of (5.1) based on $y^{(t)}$ is computed using a local search method by first randomly removing k selected nodes and the probability to remove each node is inversely proportional to its weight, then choosing the maximum weighted node in the queue that are not adjacent to those selected until all nodes are considered. This procedure continues until convergence or maximum iterations reached and the best solution is selected as $x^{(t+1)}$. The reason that we randomly remove k selected nodes is to help the optimization escape from potential local maxima.

5.2.2 Deep Convolutional Neural Network

In this section, we need to calculate the weight w_i for each detection candidate $v_i \in V$ from Sect. 5.2.1. A deep convolutional neural network (DCNN) is trained for this purpose to assign each node a proper score as its weight. In our algorithm, a detection candidate is described by a small rectangle region centering around the detected position. Some training samples are shown in the first row in Fig. 5.2. The patches whose centers are close to the true cell centers are annotated as positive (+1) samples, marked with red rectangles in Fig. 5.2. Patches that have centers far away from true cell centers will be annotated as negative (-1) samples, marked with blue rectangles in Fig. 5.2.

DCNN Architecture: In our algorithm, the input features are the raw intensities of 31×31 image patches around the detected position. Considering the staining variations and the generality of the detection framework, color information is disregarded since they may change dramatically with respect to different staining protocols. The DCNN consists of seven layers: three convolutional (C) layers, two pooling layers, and two fully connected (FC) layers. In our implementation, max pooling (MP) is applied. The MP layers select the maximal activations over nonoverlapping patches

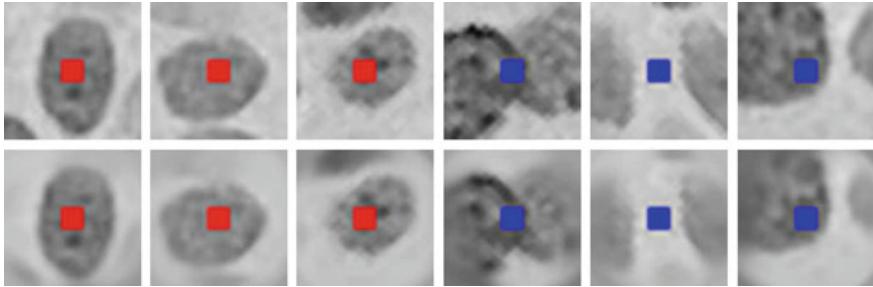


Fig. 5.2 Some training samples for DCNN and their foveation versions. The first row denote original training samples. Positive samples are marked with *red rectangles* and negatives are marked with *blue*. The second row denote the samples after foveation

Table 5.1 The configuration of the proposed DCNN architecture in our algorithm

Layer	Type	Maps (M) and neurons (N)	Filter size	Nonlinearity	Weights
0	I	$1M \times 31N \times 31N$	—	—	—
1	C	$6M \times 28N \times 28N$	4×4	Rectifier	102
2	MP	$6M \times 14N \times 14N$	2×2	—	—
3	C	$12M \times 12N \times 12N$	3×3	Rectifier	660
4	MP	$12M \times 6N \times 6N$	2×2	—	—
5	C	$12M \times 4N \times 4N$	3×3	Rectifier	1308
6	FC	100N	1×1	Rectifier	19300
7	FC	2N	1×1	Softmax	202

of the input layers. Except the output layer, where the two-way *softmax* function is used as activation function, the rectifier nonlinear activation functions are used in the convolutional layers and the fully connected layer prior to the output layer. A detailed configuration of the DCNN used in our algorithm is shown in Table 5.1.

Foveation: The task of DCNN is to classify the center pixel of each rectangle patch, so it will be ideal if we can keep the focus on the central region (fovea) and also retain the general structure of the image. Foveation, inspired by the structure of human photoreceptor topography, has been shown to be effective in imposing a spatially variant blur on images [11]. In our algorithm, a Gaussian pyramid is first built for each input image, then all the pyramid layers are resized to the input image scale. In the foveated image, pixels closer to the image center are assigned intensity values in higher resolution layers at the same coordinate, pixels far away

from the centers will be assigned values from lower resolution layers. Some foveation examples are shown in the second row of Fig. 5.2.

DCNN Training: Several cell detection algorithms [1, 2] with varying parameters are chosen to generate training samples for the DCNN. All the true cell centers are manually annotated in training images. The detected results within a certain distance τ_1 to the annotated cell centers are marked as positive training samples, others that locate far away from the centers (measured by τ_2) are marked as negative training samples, where $\tau_2 \geq \tau_1$. Each training sample is further rotated by seven angles. In our implementation, a mini-batch of size 10, which is a compromise between the standard and stochastic gradient descent forms, is used to train the DCNN. The learning rate is initiated as 0.01, and decreases as the number of epoches increases.

5.3 Experiments

The proposed algorithm is tested with two datasets: (1) 24 neuroendocrine (NET) tissue microarray (TMA) images, and (2) 16 lung cancer images. Each image contains roughly 150 cells. For each dataset, twofold cross-validation is used to evaluate the accuracy. All the true cell centers are manually labeled by doctors. An automatic detection is considered as true positive (TP) if the detected result is within a circle centered at the ground-truth annotation with a radius r . The detected results that do not fall into the circle will be labeled as false positive (FP). All missed true cell centers are counted as false negative (FN). The results are reported in terms of precision ($P = \frac{TP}{TP+FP}$) and recall ($R = \frac{TP}{TP+FN}$). Both the maximum-weight independent set (MWIS) and the deep convolutional neural network (DCNN) are evaluated in the following sections.

First, in order to justify the proposed two-step iterative algorithm to solve the MWIS problem stated in Eq.(5.1), we have compared it with a commonly used greedy non-maximum suppression (NMS) method [6], which keeps selecting an available node with the highest score and then removing the node and its neighbors until all the nodes are checked. As defined before, two nodes are considered as neighbors if their Euclidean distance is smaller than a threshold parameter λ . Taking detection results obtained from [1–3] as inputs, we generate a set of detection results for both the proposed algorithm and NMS by changing the parameter λ . The comparison of the converged object function values of Eq.(5.1) achieved by the proposed algorithm (Ours) and NMS are shown in Fig. 5.3a, d. The comparative results of detection accuracy (F_1 score) are shown in Fig. 5.3b, e for Net and Lung dataset, respectively. We can observe that: (1) Both the proposed algorithm and NMS method are insensitive to parameter λ , and (2) the proposed algorithm consistently produces solutions of better qualities in terms of maximizing the object function in Eq. (5.1) and outperforms the NMS method in most cases in terms of detection accuracy, F_1 score. For both methods, the detect candidates with scores below than 0 (1 denotes positive and -1 denotes negative while training) will not be considered.

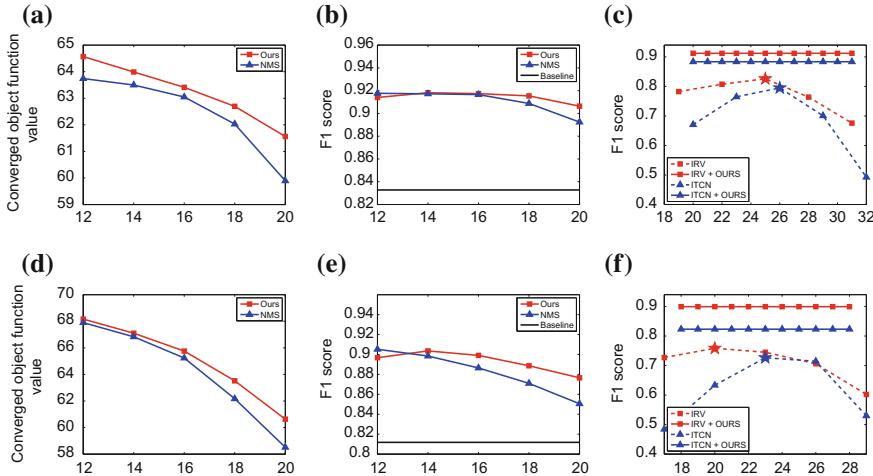


Fig. 5.3 The evaluation of the proposed general cell detection algorithm using two different datasets. The first and second row denotes results of the NET and Lung datasets, respectively. **a** and **d** The comparison of the object values of Eq. (5.1) achieved by the proposed algorithm (Ours) and NMS with different parameter λ . **b** and **e** The comparison of the proposed algorithm (Ours) and NMS by changing the parameter λ . The baseline method is the algorithm presented by Arteta et al. [3]. **c** and **f** The comparison of detection accuracies among IRV, ITCN (different parameters), and our algorithm. The best results of IRV and ITCN are marked with stars

Second, the proposed cell detection algorithm is compared with three detection algorithms: (1) Iterative radial voting (IRV) [1] with different cell diameter parameter $\{19, 22, 25, 28, 31\}$ for NET and $\{17, 20, 23, 26, 29\}$ for Lung dataset; (2) Image-based tool for counting nuclei (ITCN) [2] with diameter parameter set as $\{20, 23, 26, 29, 32\}$ for NET and $\{17, 20, 23, 26, 29\}$ for Lung dataset; (3) A learning-based cell detection algorithm (Arteta et al. [3]) that does not require the parameter selection once a structured supported vector machine is learned on the training images. Both algorithms (1) and (2) will generate a pool of detection candidates, and we will evaluate whether the proposed algorithm is capable of finding the best subset that outperforms each individual algorithm. Please note that we use IRV+OURS and ITCN+OURS to denote the proposed algorithm using the detection results of IRV and ITCN as candidates for best subset selection, respectively. The experimental results are shown in Fig. 5.3. The first row denotes the testing results using the NET dataset, and the second row presents the testing results using the lung cancer dataset. The detailed comparative results are explained below.

The comparative results of IRV, IRV+OURS, ITCN, ITCN+OURS with respect to different parameters are shown in Fig. 5.3c, f. As one can tell, whether or not IRV and ITCN can provide satisfactory results heavily depend on proper parameter selections, which is not always feasible or convenient during runtime. When the parameter is not selected correctly, the performance will deteriorate significantly as illustrated in (c) and (f) (red and blue dotted lines). However, our proposed algorithm

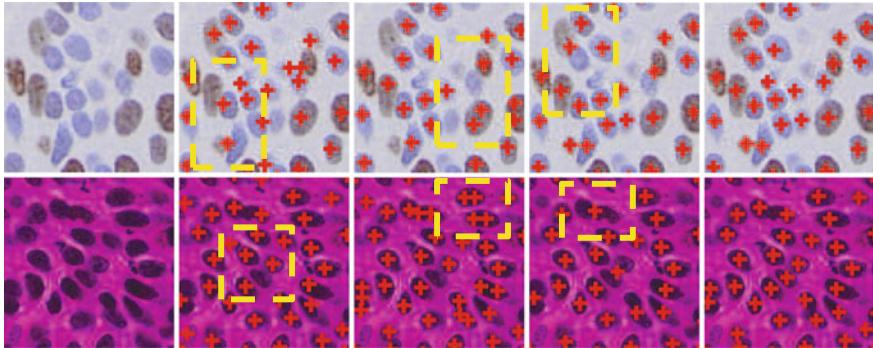


Fig. 5.4 Qualitative cell detection results using different algorithms. The first row denotes the cell detection results using NET and the second row denotes the cell detection results using the Lung cancer dataset. From *left* to *right*, the columns denote: cropped image patch, cell detection results of [1–3], and the proposed algorithm. The detection errors are labeled with *dotted yellow rectangles*

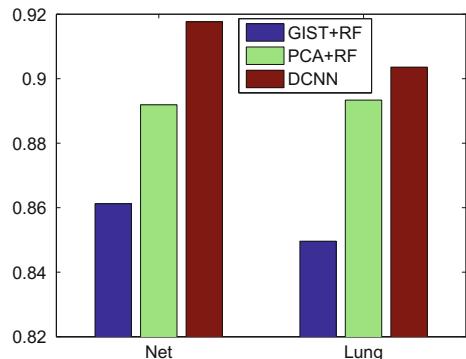
Table 5.2 Comparison of cell detection accuracy

Method	NET			Lung		
	F ₁ -score	Prec.	Rec.	F ₁ -score	Prec.	Rec.
IRV [1]	0.8260	0.7999	0.8539	0.7584	0.6657	0.8812
ITCN [2]	0.7950	0.8277	0.7647	0.7264	0.6183	0.8804
Arteta et al. [3]	0.8328	0.8806	0.7899	0.8118	0.8820	0.7520
[1]+[2]+[3]+OURS	0.9182	0.9003	0.9369	0.9036	0.8843	0.9237

does not require careful selection of parameters as shown in Fig. 5.3b, e. In addition, it consistently outperforms any best individual detection result using IRV and ITCN (red and blue lines) (Fig. 5.4).

In order to justify the accuracy of the assigned weights w using DCNN in Eq. (5.1), we have compared DCNN with a random forest (RF) classifier using different features: (1) Global scene descriptor (GIST) and (2) raw pixel values following by a principle component analysis (PCA) for the dimension reduction. The comparison results can be seen in Fig. 5.5. It is obvious that DCNN consistently provides better results than other methods on both Net and Lung datasets. The quantitative detection results are summarized in Table 5.2. We can see that the proposed algorithm consistently performs better than both the parameter sensitive methods (IRV and ITCN) and the parameter nonsensitive method [3]. Please note that in Table 5.2, we report the best detection results of [1] and [2] using the optimal parameters. Some qualitative automatic cell detection results are shown in Fig. 5.4 using both NET and lung cancer data.

Fig. 5.5 Comparisons of methods to compute w in Eq.(5.1)



5.4 Conclusion

In this chapter, we have proposed a novel cell detection algorithm based on maximum-weight independent set selection that will choose the heaviest subset from a pool of cell detection candidates generated from different algorithms using various parameters. The weights of the graph are computed using a deep convolutional neural network. Our experiments show that this novel algorithm provides ensemble detection results that can boost the accuracy of any individual cell detection algorithm.

References

- Parvin B, Yang Q, Han J, Chang H, Rydberg B, Barcellos-Hoff MH (2007) Iterative voting for inference of structural saliency and characterization of subcellular events. *TIP* 16(3):615–623
- Byun J, Verardo MR, Sumengen B, Lewis GP, Manjunath B, Fisher SK (2006) Automated tool for the detection of cell nuclei in digital microscopic images: application to retinal images. *Mol. Vis.* 12:949–960
- Arteta C, Lempitsky V, Noble JA, Zisserman A (2012) Learning to detect cells using non-overlapping extremal regions. In: MICCAI. Springer, Berlin, pp 348–356
- Matas J, Chum O, Urban M, Pajdla T (2004) Robust wide-baseline stereo from maximally stable extremal regions. *Image Vis Comput* 22(10):761–767
- LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324
- Cireşan DC, Giusti A, Gambardella LM, Schmidhuber J (2013) Mitosis detection in breast cancer histology images with deep neural networks. In: MICCAI. Springer, Berlin, pp 411–418
- Li N, Latecki LJ (2012) Clustering aggregation as maximum-weight independent set. In: NIPS, pp 791–799
- Brendel W, Todorovic S (2010) Segmentation as maximum-weight independent set. In: NIPS, pp 307–315

9. Brendel W, Amer M, Todorovic S (2011) Multiobject tracking as maximum weight independent set. In: CVPR. IEEE, New York, pp 1273–1280
10. Gold S, Rangarajan A (1996) A graduated assignment algorithm for graph matching. PAMI 18(4):377–388
11. Ciresan D, Giusti A, Schmidhuber J et al. (2012) Deep neural networks segment neuronal membranes in electron microscopy images. In: NIPS, pp 2852–2860

Chapter 6

Deep Learning for Histopathological Image Analysis: Towards Computerized Diagnosis on Cancers

Jun Xu, Chao Zhou, Bing Lang and Qingshan Liu

Abstract Automated detection and segmentation of histologic primitives are critical steps for developing computer-aided diagnosis and prognosis system on histopathological tissue specimens. For a number of cancers, the clinical cancer grading system is highly correlated with the pathomic features of histologic primitives that appreciated from histopathological images. However, automated detection and segmentation of histologic primitives is pretty challenged because of the complicity and high density of histologic data. Therefore, there is a high demand for developing intelligent and computational image analysis tools for digital pathology images. Recently there have been interests in the application of “Deep Learning” strategies for classification and analysis of big image data. Histopathology, given its size and complexity, represents an excellent use case for application of deep learning strategies. In this chapter, we present deep learning based approaches for two challenged tasks in histological image analysis: (1) Automated nuclear atypia scoring (NAS) on breast histopathology. We present a Multi-Resolution Convolutional Network (MR-CN) with Plurality Voting (MR-CN-PV) model for automated NAS. MR-CN-PV consists of three Single-Resolution Convolutional Network (SR-CN) with Majority Voting (SR-CN-MV) model for getting independent NAS. MR-CN-PV combines three scores via plurality voting for getting final score. (2) Epithelial (EP) and stromal (ST) tissues discrimination. The work utilized a pixel-wise Convolutional Network (CN-PI) based segmentation model for automated EP and ST tissues discrimination. We present experiments on two challenged datasets. For automated NAS, the MR-CN-PV model was evaluated on MITOS-ATYPIA-14 Challenge dataset. MR-CN-PV model got 67 score which was placed the second comparing with the scores of other five teams. The proposed CN-PI model outperformed patch-wise CN (CN-PA) models in discriminating EP and ST tissues on a breast histological images.

J. Xu (✉) · C. Zhou · B. Lang · Q. Liu

Jiangsu Key Laboratory of Big Data Analysis Technique, Nanjing University of Information Science and Technology, Nanjing 210044, China
e-mail: xujung@gmail.com

6.1 Introduction

Cancer is the leading cause of death in the United States [1] and China [2]. In 2015 alone there were 4.3 million new cancer cases and more than 2.8 million cancer deaths in China [2]. Fortunately, most of the cancers have a very high chance of cure if detected early and treated adequately. Therefore, earlier diagnosis on cancers and better prognostic prediction of disease aggressiveness and patient outcome are pretty important. The pathological diagnosis remains the “gold standard” in cancer diagnosis [3]. Currently, the routine assessment of histological grade and other prognostic factors for cancers are done by pathologists looking over the Hematoxylin & Eosin (H&E) stained histopathology images under microscope. Histological grade based on H&E image is a morphological assessment of tumor biological characteristics and has been shown to be highly correlated to the patient outcome in long-term survival or disease-free survival [4, 5]. For instance, the Nottingham Grading System (NGS) is one of the popular grading system used world wide for evaluating the aggressiveness of breast cancer. In this system, the pathologists take into considerations three factors, which are nuclear atypia, tubule formation, and mitotic rate. Nuclear atypia refers to abnormal appearance of cell nuclei. The introduction of nuclear atypia scoring covers a second important criteria necessary for breast cancer grading. It gives an indication about the stage of evolution of the cancer. Nuclear Atypia Score (NAS) is a value, 1, 2, or 3, corresponding to a low, moderate or strong nuclear atypia (see Fig. 6.1), respectively. Epithelial (EP) and Stromal (ST) tissues are two basic tissues in histological samples. In breast tissue samples, ST tissue includes the fatty and fibrous connective tissues surrounding the ducts and lobules, blood vessels, and lymphatic vessels, which are supportive framework of an organ. EP tissue is the cellular tissue lining and found in the ductal and lobular system of the breast milk ducts. About 80% breast tumors originate in the breast EP cells. Although ST tissue is typically considered as not being part of malignant tissue, the changes in the stroma tend to drive tumor invasion and metastasis [6]. Therefore, tumor-stroma ratio in histological tissues is being recognized as an important prognostic value [7], since cancer growth and progression is dependent on the microenvironment of EP and ST tissues. Yuan et al. in [8] found that the spatial arrangement of stromal cell in tumors is a prognostic factor in breast cancer. Consequently a critical initial step in developing automated computerized algorithms for risk assessment and prognosis determination is to be able to distinguish stromal from epithelial tissue compartments on digital pathology images.

Histologic image assessment has remained experience-based qualitative [9], and it always causes intra- or inter-observers variation [10] even for experienced pathologists [11]. This ultimately results in inaccurate diagnosis. Moreover, human interpretation on histological images has low agreements among different pathologists [9]. Inaccurate diagnosis may results in severe overtreatment or undertreatment, thus causing serious harms to patients. There is an acute demand for developing computational image analysis tools to help pathologists make faster and more accurate diagnosis [12]. With the advent of whole-slide digital scanners, the traditional glass

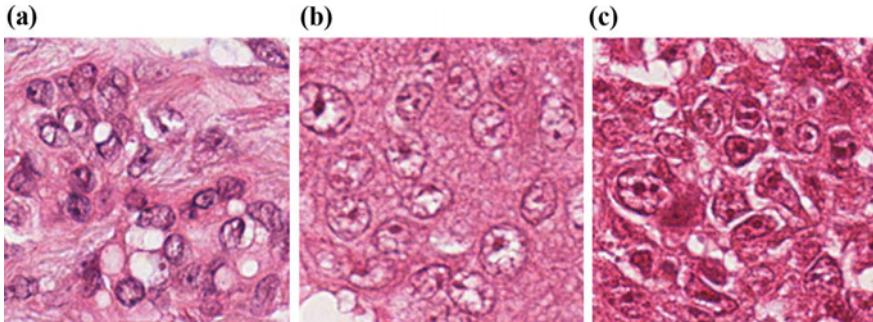


Fig. 6.1 The sample images under $\times 40$ magnification with different Nuclear Atypia Scores (NAS): **a** NAS = 1, **b** NAS = 2, and **c** NAS = 3

slides can now be digitalized and stored in digital image form [13]. Digital pathology makes computerized quantitative analysis of histopathology imagery possible [5]. The interpretation of pathological images via computerized techniques is becoming a powerful tool for probing a wide variety of pathology problems [14, 15]. Studies have showed that such tools have the potential to tackle the inherent subjectivity in manual qualitative interpretation, and they largely reduce the workload of pathologists via high-throughput analysis [9]. Toward this end, in this chapter, we focus on two challenged problems: (1) Automated nuclear atypia scoring; (2) Epithelial (EP) and Stromal (ST) tissues discrimination on breast histopathology.

The rest of the paper is organized as follows: A review of previously related works on deep learning (DL) for histological image analysis, EP and ST discrimination, and automated nuclear atypia scoring (NAS) is presented in Sect. 6.2. A detailed description of methodology on leveraging DL for automated NAS as well as EP and ST tissues discrimination are presented in Sects. 6.3 and 6.4, respectively. The experimental setup and comparative strategies are discussed in Sect. 6.5. The experimental results and discussions are reported in Sect. 6.6. Concluding remarks are presented in Sect. 6.7.

6.2 Previous Works

As Figs. 6.1 and 6.2 show, histologic images are highly challenged data. It is extremely challenging for automated image analysis tools due to the high data density, the complexity of the tissue structures, and the inconsistencies in tissue preparation. Therefore, it is crucial to develop intelligent algorithms for automated detection and segmentation of histologic primitives as well as the classification of tissue samples in an accurate, fast, practical, and robust manner [9].

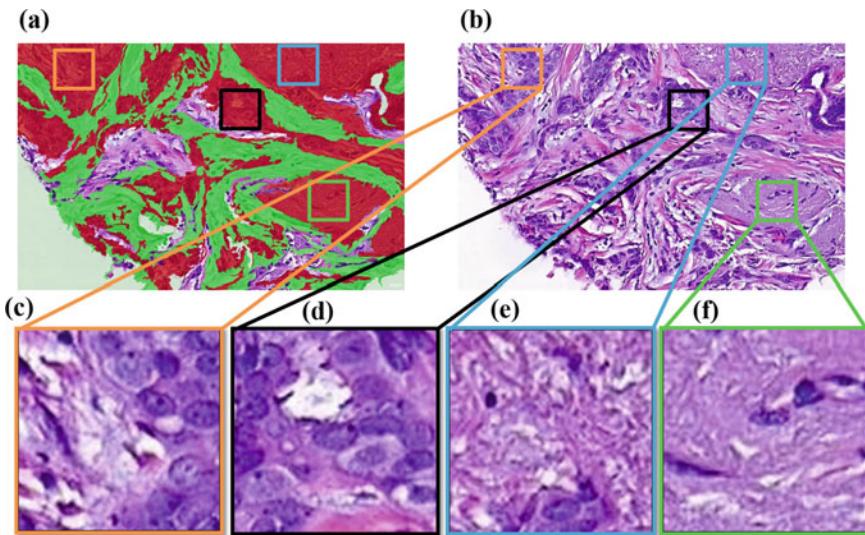


Fig. 6.2 The illustration of epithelial and stromal regions in a breast histologic tissue sample. The annotated epithelial (red) and stromal (green) regions are shown in **a**. **c–f** are four different epithelial patches from the original tissue sample (**b**) which have been magnified to show the details of epithelial regions

6.2.1 Previous Works on Deep Learning for Histological Image Analysis

Recently, deep convolutional network (CN), an instance of Deep Learning (DL) architectures, have shown its advantageous in image analysis over other non-deep learning based approaches. DL is a data-driven and end-to-end learning approach which learns high-level structure features from just pixel intensities alone that are useful for differentiating objects by a classifier. Recently, it has been successfully employed for medical image analysis with various applications [16–24]. The DL based approaches have evoked great interests from the histological image analysis community since the pioneer work in [25]. Histopathology, given its size and complexity, represents an excellent use case for application of deep learning strategies. In [25], a popular six-layer CN, which is also called “ConvNet” is employed for mitotic detection. This work won the ICPR 2012 contest and MICCAI 2013 Grand Challenge on mitotic detection. The model was a patch-wise training process for pixel-wise labeling. It was first trained with a great amount of context patches. Basically, there were two types of context patches: foreground patches whose central pixels are located within target objects and background patches whose central pixels are located around the neighborhood pixel of the target objects. After training, the model was employed to predict the central pixel of chosen patches being targeted objects or not. Recently, much effort has been focused on nuclear detection or segmentation

[26–28]. In terms of nuclear detection, a Stacked Sparse Autoencoder (SSAE) based model was employed in [29] for discriminating nuclear and non-nuclear patches. Then integrating with the sliding window operation the SSAE model was further utilized to automated nuclear detection from high-resolution histological images in [26]. In [27], a Spatially Constrained CN was presented to nucleus detection. This segmentation free strategy can detect and classify different nuclear types simultaneously on colorectal adenocarcinoma images. The CN involves convolutional and subsampling operations to learn a set of locally connected neurons through local receptive fields for feature extraction. Therefore, CN is good at capturing contextual information. Based on these contexture information, a pixel-wise based CN was developed for pixel-wise segmentation of nuclear regions in [28]. Pixel-wise segmentation is different from patch-wise classification since pixel-wise segmentation aims to predict class label of each pixel in an image based on a local patch around that pixel [25], while patch-wise classification aims to assign a single label to the entire image patch [30]. Therefore, pixel-wise classification is more challenged. In [31], the authors employed a convolutional autoencoder neural network architecture with autoencoder for histopathological image representation learning. Then a softmax classification is employed for classifying regions of cancer and noncancer.

6.2.2 Previous Works on Nuclear Atypia Scoring

Automated Nuclear Atypia Scoring (NAS) is a challenged task. Currently, a few works were reported in this field. Most of current works focus on nuclei detection and segmentation since the NAS criteria is highly related to the shape, texture, and morphological features of nuclei in the tissue samples. In [32, 33], nuclear regions were first segmented and then a classifier was trained to grade the tissues based on the nuclear features extracted from segmented nuclear regions. In [34], the authors developed a method to select and segment critical nuclei within a histopathological image for NAS. However, it is a pretty challenged task to accurately detect and segment nuclei in the cancerous regions, especially in the regions with strong NAS (i.e., $NAS = 3$). Therefore, image-level classification might be a better solution to this problem. Different from nuclear segmentation based approaches, a image-level analysis based approach was proposed in [35] for NAS. Image-level feature extraction had been extensively utilized for distinguishing normal and cancerous tissue samples in [31, 36]. In order to differentiate entire ER+BCa histopathology slides based on their mBR grades, a multi-fields-of-view (multi-FOV) classifier was utilize in [37, 38] to automatically integrate image features from multiple fields of views (FOV) at various sizes. Inspired by these works, we present a Multi-Resolution Convolutional Network (MR-CN) which consists of a combination of three Single-Resolution Convolutional Network (SR-CN) paths connected with plurality voting strategy for automated NAS, while each SR-CN path integrated with majority voting for independent NAS.

6.2.3 Previous Works on Epithelial and Stromal Segmentation

Handcrafted features based approaches had been extensively employed to the recognition of different tissues in histological images. In [39], local binary pattern (LBP) and contrast measure based texture features were used for discriminating EP and ST regions from immunohistochemistry (IHC) stained images of colorectal cancer. More recently, five perception-based features related to human perception were studied in [40] to differentiate EP and ST patches. In [41], color-based texture features extracted from square image blocks for automated segmentation of stromal tissue from IHC images of breast cancer. A binary graph cuts approach where the graph weights were determined based on the color histogram of two regions, was used for segmenting EP and ST regions from odontogenic cysts images in [42]. A wavelet-based multiscale texture is presented in [43] for the segmentation of the various types of stromal compartments on ovarian carcinoma virtual slides. In [44], a cell graph feature describing the topological distribution of the tissue cell nuclei was used for discriminating tumor and stromal areas on immunofluorescence histological images. In [45], IHC stained TMA cores were automatically stratified as tumor or non-tumor cores based on a visual word dictionary learning approach. In [46], a multi-class texture based model is presented for identifying eight different types of tissues from colorectal cancer histology. A publicly available interactive tool called Ilastik was employed in [47] for pixel-wise image segmentation of glands and epithelium from other types of tissue in the colorectal digitized sample. The tool is trained based on labels provided by the user and each pixels neighborhood in the image is characterized by nonlinear features such as color, texture, and edge features. A random forest classifier is then used to produce pixel-level segmentations. We showed in [30] that patch-wise CN (CN-PA) based models outperform handcrafted features based approaches in [39, 40] for discriminating EP and ST tissues. In this work, we will leverage pixel-wise CN (CN-PI) for pixel-pixel segmentation of EP and ST regions from histological tissue images.

For simplicity, the symbols used in the paper was enumerated in Table 6.1.

6.3 Deep Learning for Nuclear Atypia Scoring

A slide (or case) with different Resolution of Views (ROVs) is defined as an image set

$$\mathbf{y} = \{\mathbf{y}(s_h), \mathbf{y}(s_2), \dots, \mathbf{y}(s_H)\}, \quad (6.1)$$

where s_h ($h = 1, 2, \dots, H$) represents a ROV of a slide. For instance, a slide \mathbf{y} in Fig. 6.8 has three different ROVs which are $\mathbf{y}(s_1)$ at $\times 10$, $\mathbf{y}(s_2)$ at $\times 20$, and $\mathbf{y}(s_3)$ at $\times 40$, respectively. Their sizes are 769×688 , 1539×1376 , and 3078×2752 , respectively. Each slide or case \mathbf{y} has a NAS or label l . Here $l \in \{1, 2, 3\}$ represents

Table 6.1 Enumeration of the symbols used in the paper

Symbol	Description	Symbol	Description
NGS	Nottingham Grading System	NAS	Nuclear Atypia Scoring
EP	Epithelial/Epithelium	ST	Stromal/Stroma
H & E	Hematoxylin and Eosin	IHC	Immunohistochemistry
PI	Pixel-wise	PA	Patch-wise
DL	Deep Learning	CN	Convolutional Networks
CN-PI	Pixel-wise CN	CN-PA	Patch-wise CN
SR	Single Resolution	MR	Multi-Resolution
MV	Majority Voting	PV	Plurality Voting
SR-CN-MV	Single-Resolution Convolutional Networks with Majority Voting	MR-CN-PV	Multi-Resolution Convolutional Networks Plurality Voting
ROV	Resolution of View	FOV	Field of View
NKI	Netherlands Cancer Institute	VGH	Vancouver General Hospital
D ₁	Dataset 1: Nuclear Atypia Scoring	D ₂	Dataset 2: EP and ST tissues discrimination
SLIC	Simple Linear Iterative Clustering algorithm	Ncut	Normalized Cuts algorithm
SP	Superpixel	SW	Sliding Window
F1	F1 score	ACC	Accuracy
TPR	True Positive Rate	FPR	False Positive Rate
FNR	False Negative Rate	TNR	True Negative Rate
PPV	Positive Predictive Rate	Negative Predictive Rate	NPR
FDR	False Discovery Rate	FNR	False Negative Rate
MCC	Matthews Correlation Coefficient	ROC	Receiver Operating Characteristic

NAS with three different scores, i.e., NAS = 1, NAS = 2, and NAS = 3, respectively. The aim of the work is to find a optimal map $f(\cdot)$ for each input slide \mathbf{y} which can be written as

$$f : \mathbf{y} \mapsto l \quad (6.2)$$

where \mathbf{y} comprises of three ROVs of a slide.

Figure 6.3 shows the flowchart of proposed Multi-Resolution Convolutional Network with Plurality Voting (MR-CN-PV) model for NAS. The model consists of a combination of three Single-Resolution Convolutional Network with Majority Voting (SR-CN-MV) paths. Each path independently learns a representation via CN and

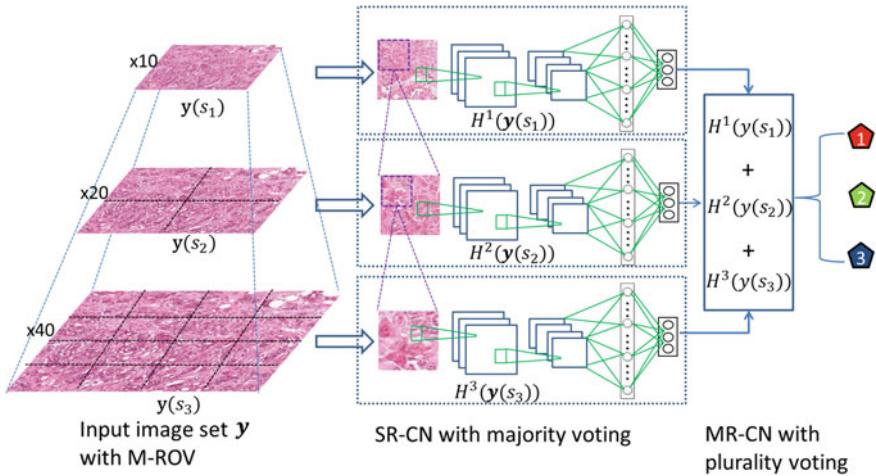


Fig. 6.3 The flowchart of proposed MR-CN-PV model for automated nuclear atypia scoring on a multi-resolution-of-view (M-ROV) histologic image. MR-CN-PV consists of a combination of three SR-CN-MV paths connected with plurality voting strategy

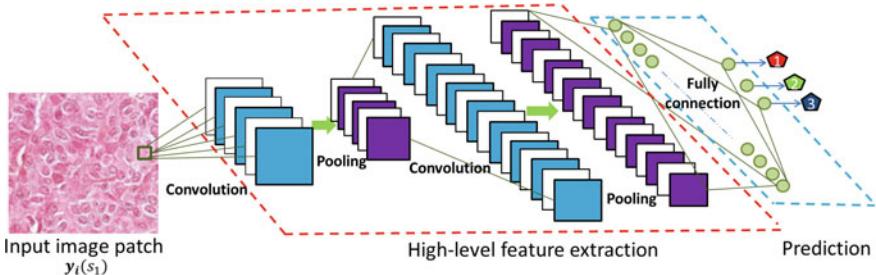


Fig. 6.4 The illustration of CN configuration for automated nuclear atypia scoring

performs NAS through majority voting at its own. Predictions across all paths are then integrated with plurality voting strategy for final NAS.

6.3.1 CN Model for Nuclear Atypia Scoring

CN model employed in this work is based on AlexNet network [48]. The review of AlexNet network architecture is described in [49] and we direct interested readers to the paper for a detailed description of the architecture. The configuration of the network is shown Fig. 6.5. The detailed architecture used in this work is shown in Table 6.2. From each image $\mathbf{y}(s_l)$ in the training set, we randomly chose many

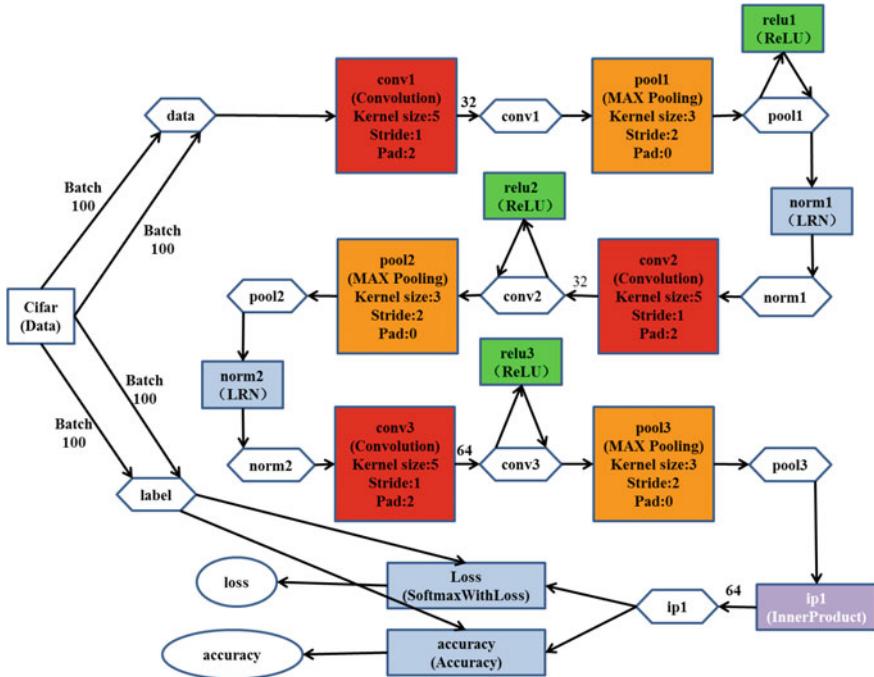


Fig. 6.5 The visualization of CN architecture used for pixel-wise segmentation

256×256 histologic patches $\mathbf{y}_i(s_l)$ to train a CN. The number of training patches chosen from training slides is shown in Table 6.3.

6.3.2 Integration MR-CN with Combination Voting Strategies for NAS

As Fig. 6.4 shows, each SR-CN independently learns a representation via CN from training images with a single resolution. Each SR-CNN is integrated with majority voting to independently performs NAS.

6.3.2.1 Majority Voting

For each ROV of a slide, a majority voting strategy [50] is integrated with trained SR-CN for NAS. The voting strategy can be described as

Table 6.2 The architecture of CN used in this work

Layer	Operation	# of Kernels	Kernel size	Stride	Padding	Activation function	Normalization
Nuclear atypia scoring							
1	Input	3	–	–	–	–	–
2	Convolution	96	11×11	4	–	ReLU	LRN
3	Pooling	96	3×3	2	–	–	–
4	Convolution	256	5×5	1	2	ReLU	LRN
5	Pooling	256	3×3	2	–	–	–
6	Convolution	384	3×3	1	1	ReLU	–
7	Convolution	384	3×3	1	1	ReLU	–
8	Convolution	256	3×3	1	1	ReLU	–
9	Pooling	256	3×3	2	–	–	–
10	Fully connected	256	–	–	–	ReLU	–
11	Fully connected	128	–	–	–	ReLU	–
12	Output	3	–	–	–	–	–
EP and ST discrimination							
1	Input	3	32×32	–	–	–	–
2	Convolution	32	5×5	1	2	–	–
3	Max pooling	32	3×3	2	0	ReLU	LRN
4	Convolution	32	5×5	1	2	ReLU	–
5	Max pooling	32	3×3	2	0	–	LRN
6	Convolution	64	5×5	1	2	ReLU	–
7	Max pooling	64	3×3	2	0	–	–
8	Fully connected	64	–	–	–	–	–
9	Fully connected	64	–	–	–	–	–
10	Output	2	–	–	–	–	–

$$H(\mathbf{y}(s_l)) = \begin{cases} C_j, & \text{if } \sum_{i=1}^T h^j(y_i(s_l)) > 0.5 \sum_{k=1}^N \sum_{i=1}^T h^k(y_i(s_l)); \\ \text{random}, & \text{otherwise.} \end{cases} \quad (6.3)$$

where $h^j(y_i(s_l))$ is the predicted class C_j ($j \in \{1, 2, 3\}$) for the input of i th image patch $y_i(s_l)$ by the CN-based classifier $h(\cdot)$ (see Fig. 6.3). Here s_l ($l \in \{1, 2, 3\}$) represents a slide \mathbf{y} in the testing set with three different resolutions: $\times 10$, $\times 20$, and $\times 40$, respectively. Similar to training images, the patches $y_i(s_l)$ are randomly chosen from a testing image via sliding window scheme whose size is 256×256 . The voting strategy in Eq. (6.3) can be described as follows. For an image $\mathbf{y}(s_l)$ under a particular resolution s_l , if there are more than 50% image patches in $\mathbf{y}(s_l)$ being predicted as C_j , the image is labeled as C_j . Otherwise, the image will be randomly labeled.

Table 6.3 The number of training and testing images as well as the corresponding training and testing patches for D₁ and D₂

Nuclear atypia scoring							
Resolution	# of total images	Training				Testing	
		Training		Validation			
		# of images	# of patches	# of images	# of patches		
×10	297	274	7023	23	703	124	
			7049		706		
			7080		705		
×20	1188	1116	14004	72	1328	496	
			14160		1344		
			14140		1356		
×40	4752	4544	28108	208	2655	1984	
			28235		2834		
			28265		2647		
EP and ST discrimination							
Dataset	# of images	Tissue	Training			Testing	
			# of images	Training set	Validation set		
NKI	106	Epithelium Stroma	85	77804	41721	21	
				70215	37625		
VGH	51	Epithelium Stroma	41	40593	16914	10	
				36634	15264		

6.3.2.2 Plurality Voting

After each ROV in a slide is graded with the corresponding SR-CN-MV model, a plurality voting strategy is leveraged to combine three scores by SR-CN-MV models for getting final score. The plurality voting is defined as [50]

$$FS(\mathbf{y}) = C_{\arg\max_j} \sum_{l=1}^3 H^j(\mathbf{y}(s_l)) \quad (6.4)$$

where $FS(\mathbf{y})$ is the final score of each slide in the testing set and $H^j(\mathbf{y}(s_l))$ is the score by a SR-CN-MV model which is computed with Eq. (6.3).

We train three SR-CN-MV models with three ROVs of the slides in the training set, respectively. During the testing, each trained SR-CN-MV independently performs NAS on a ROV of the input slide. Finally, a plurality voting approach is utilized to combine the scores by three SR-CN-MV and get final NAS for each slide in the testing set.

6.4 Deep Learning for Epithelial and Stromal Tissues Segmentation

In this section, we present CN-based approach for pixel-wise segmentation of EP and ST tissues on breast histological tissue samples.

6.4.1 The Deep Convolutional Neural Networks

The CN employed in this work is based on AlexNet network [48] which was trained on image benchmark challenge CIFAR-10. The visualization of the network's architecture is shown in Fig. 6.5.

6.4.2 Generating Training and Testing Samples

In this work, the generation of training samples is critical. Define $R(\cdot)$ a $d \times d$ patch extraction operator and $R(c_{uv}) \in R^{d^2}$ a context patch around pixel c_{uv} of the image C.

$$R(c_{uv}) = \left\{ c_{lm}, c_{lm} \in \mathbf{C} \mid i - \frac{d}{2} \leq l \leq i + \frac{d}{2}, j - \frac{d}{2} \leq m \leq j + \frac{d}{2} \right\}. \quad (6.5)$$

where $d = 32$ in this work. The context patch $R(c_{uv})$ accommodate the local spatial dependencies among central pixel and its neighborhoods in the context patch. As Fig. 6.6 shows, two types of training patches are extracted from training images

1. The EP patches whose central pixels (c_{uv} in Eq. (6.5)) are located within annotated EP regions;
2. The ST patches whose central pixels (c_{uv} in Eq. (6.5)) are located within annotated ST regions.

Table 6.3 shows the number of training and testing images for two data cohorts of D₂ evaluated in this work. For D₂, we randomly selected 106 images from Netherlands Cancer Institute (NKI) Dataset cohort and 51 from Vancouver General Hospital (VGH) dataset cohort as training images. The remaining 21 images from NKI and 10 images from VGH were used for testing, respectively. The images corresponding to the training sets with NKI and VGH dataset cohorts were used for generating training patches. Figure 6.6 shows the procedure of generating challenged training patches from the images corresponding to the training sets with NKI and VGH dataset cohorts. First, dilation operation is applied to the boundaries of EP and ST regions of a training image. This operation will result in the thicker boundary maps. Then EP (region in red box in Fig. 6.6e) or stromal (region in green box 6.6f) patches are extracted from the tissues based on the location of central pixel (c_{uv} in Eq. (6.5))

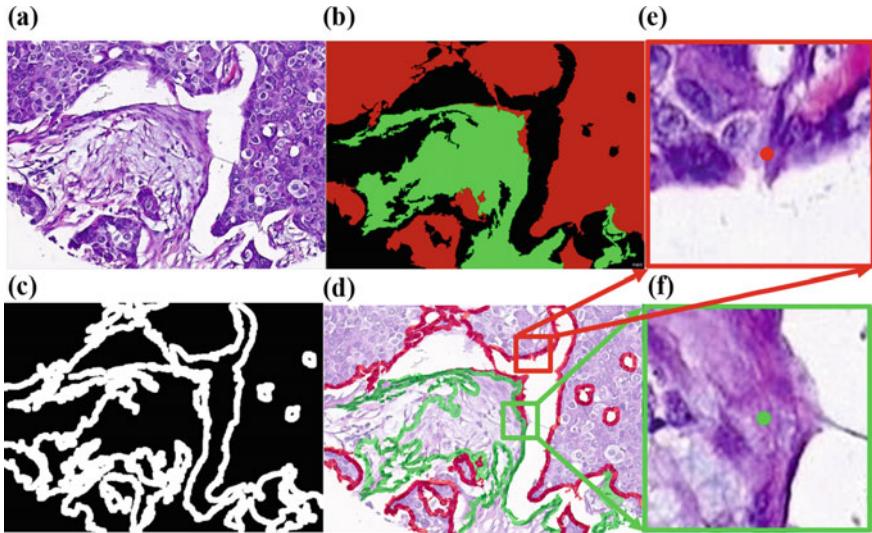


Fig. 6.6 The illustration of generating challenged training patches on a tissue sample (a). Dilation operation is applied to the boundaries of epithelial and stromal regions in annotation map (b) which results in the thicker boundary maps (c). Then epithelial (e) or stromal (f) patches are extracted from the tissues based on the location of central pixel (c_{uv} in Eq. (6.5)) in epithelial or stromal regions, respectively

in EP or ST regions, respectively. Besides the challenged training patches, we also extracted a great amount of general patches within EP and ST regions based on the manual annotation (see red and green regions Fig. 6.6b). The number of training patches in this work extracted from training images is given in Table 6.3.

6.4.3 The Trained CN for the Discrimination of EP and ST Regions

Figure 6.7 shows the flowchart of proposed CN based model for pixel-wise discriminating EP and ST regions in a breast histological tissue sample. For each testing image from NKI and VGH cohorts, a sliding window scheme is leveraged to choose the same context image patches based on Eq. (6.5). The window slides across the entire image row by row from upper left corner to the lower right with a step size of 1 pixel. Border padding is employed to address issues of boundary artifacts (see Table 6.2). The pixel-wise segmentation is achieved by predicting the class probabilities of the central pixel c_{uv} of each context patch $R(c_{uv})$ chosen by the sliding window scheme, which can be described by the following equation:

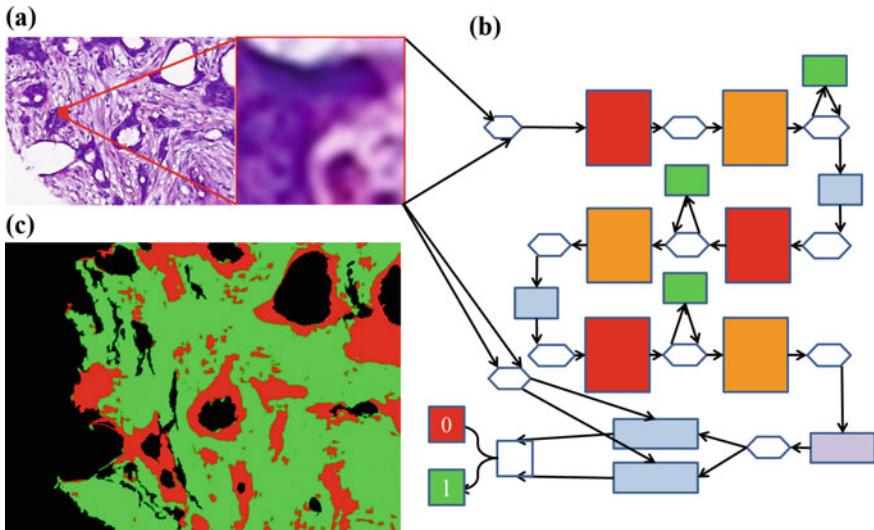


Fig. 6.7 The illustration of pixel-wise segmentation with CN for epithelial and stromal segmentation for an input tissue sample **(a)** where each context patch feeding to CN is 32×32 . **b** The trained CN. **c** The segmentation results on a input tissue sample **(a)** where the resultant false map represent EP (red) and ST (green), respectively

$$p_W(l = k|R(c_{uv})) = \frac{1}{1 + \exp(-W^T R(c_{uv}))}, \quad (6.6)$$

where $p_W(\cdot)$ ($k \in \{0, 1\}$) is a sigmoid function with parameters W . The final predicted class $l = 0$ or $l = 1$ of c_{uv} is determined by the higher probability of prediction results on the context patch $R(c_{uv})$.

In Fig. 6.7, predicted class $l = 0$ or $l = 1$ with Eq. (6.6) via CN model represents the EP or ST pixel, respectively.

6.5 Experimental Setup

In order to show the effectiveness of the proposed models on two challenged tasks, the proposed and comparative models are evaluated on two data sets D_1 and D_2 , respectively.

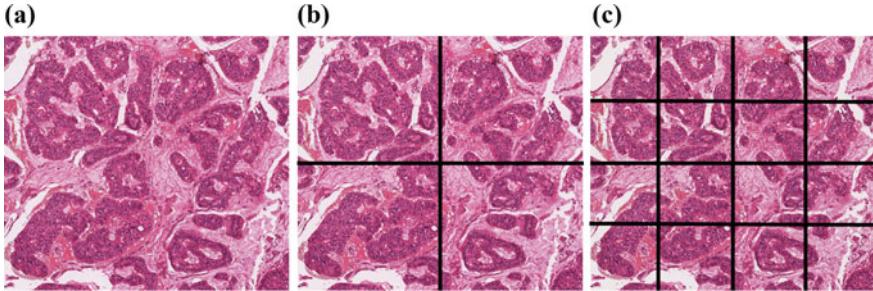


Fig. 6.8 The illustration of a case y in D_1 with three different magnifications: **a** $y(s_1)$ has a frame at $\times 10$, **b** $y(s_2)$ has 4 frames at $\times 20$, and **c** $y(s_3)$ has 16 frames at $\times 40$

6.5.1 Data Set

6.5.1.1 Dataset 1 (D_1): Nuclear Atypia Scoring

The dataset was provided by the 22nd ICPR NAS contest [51]. The slides were stained with standard H&E dyes and they have been scanned by slide scanner Aperio Scanscope XT. In each slide, the pathologists selected several frames at $\times 10$ magnification. Each $\times 10$ frame is subdivided into four frames at $\times 20$ magnification. Each $\times 20$ frame is also subdivided into four frames at $\times 40$ magnification (see Fig. 6.8). Therefore, each slide y includes three H&E stained histologic images with three different magnifications: $\times 10$ (i.e., $y(s_1)$), $\times 20$ (i.e., $y(s_2)$), and $\times 40$ (i.e., $y(s_3)$), whose sizes are 769×688 , 1539×1376 , and 3078×2752 , respectively. The number of training images with different resolutions are shown in Table 6.3. 124 slides were provided for testing. The dataset only provided the NAS for the slides in the training set. The NAS of the slides in the testing set were not provided.

6.5.1.2 Dataset 2 (D_2): Epithelial and Stromal Tissues Discrimination

This data set was downloaded via the links provided in [52]. The data was acquired from two independent cohorts: Netherlands Cancer Institute (NKI) and Vancouver General Hospital (VGH). It consists of 157 rectangular image regions (106 NKI, 51 VGH) in which Epithelial and Stromal regions were manually annotated by pathologists. The images are H&E stained histologic images from breast cancer TMAs. The size of each image is 1128×720 pixels at a $20X$ optical magnification.

6.5.2 Comparison Strategies

6.5.2.1 Nuclear Atypia Scoring

The proposed method is compared against other state-of-the-art methods for the performance of automated NAS on the slides from the testing set. As the dataset did not provide the groundtruth, the NAS on each testing slide were submitted to the organizer of the contest for evaluation. The performance and the rank were then returned by the organizer. We also compare the performance of each single path of SR-CN-MV on a particular ROV against MR-CN-PV across three ROVs (see Fig. 6.3).

6.5.2.2 Epithelial and Stromal Tissues Discrimination

We compare the proposed pixel-wise CN (CN-PI) with patch-wise CN (CN-PA) studied in [30] in discriminating EP and ST tissues. A detailed description of proposed and comparative models are illustrated in Table 6.4. The comparative models are described in the paper [30] and we direct interested readers to the paper for a detailed description of the models.

Table 6.4 The illustration of models considered in the paper for comparison and the detailed description of the different models

Nuclear atypia scoring		Input images	Size of patches	Network	Voting strategy
SR-CN-MV	Image with single resolution	256 × 256	AlexNet	Majority voting	
	Image with multiple resolutions				Plurality voting
EP and ST discrimination					
Models		Generating patches	Size of patches	Network	Classifier
CN-PA	CN-SW [30]	Sliding window + square image	50 × 50	AlexNet	SMC
	CN-Ncut [30]	Superpixel(Ncut) + square image			
	CN-SLIC [30]	Superpixel(SLIC) + square image			
CN-PI		Context patch of pixel level	32 × 32		

6.5.3 Computational and Implemental Consideration

All the experiments were carried out on a PC (Intel Core(TM) i7-3770@3.40GHz GHz processor with 16 GB of DDR3 1600MHz RAM), a Titan X NVIDIA Graphics Processor Unit and Hard Disk Seagate ST31000524AS (1TB/7200). The software implementation was performed using MATLAB 2014b with Ubuntu14.04 Linux system. CN model was implemented on Caffe framework [53].

6.6 Results and Discussion

6.6.1 Qualitative Results

The qualitative segmentation results (Fig. 6.9c–f) of the different CN models for a histological image in D_2 (Fig. 6.9a) are shown in Fig. 6.9. In Fig. 6.9b–f, green and red regions represent epithelial and stromal regions that were segmented with respect to the pathologist determined ground truth (Fig. 6.9b). The black areas in Fig. 6.9b–f were identified as background regions and hence not worth computationally interrogating. The results in Fig. 6.9c–f appear to suggest that CN-PI based model (Fig. 6.9c) outperform CN-PA models Fig. 6.9d–f.

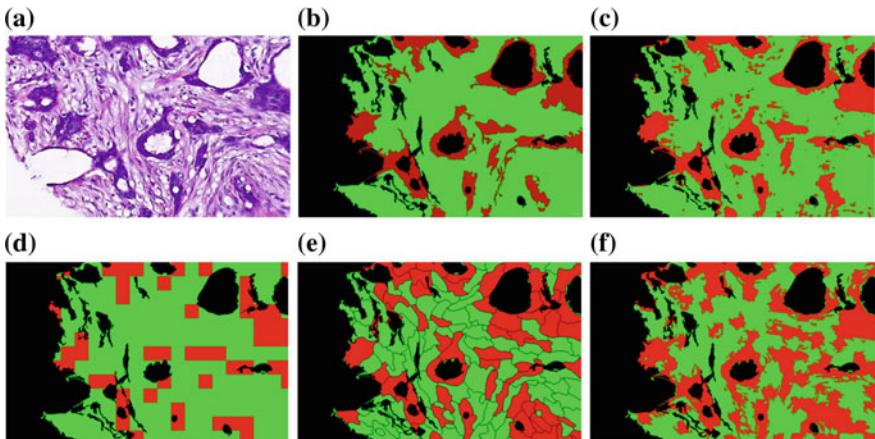


Fig. 6.9 Segmentation of epithelial (red) and stromal (green) regions on a tissue image (a) using the different segmentation approaches on D_2 . b The ground truth of annotations of stromal and epithelial regions by an expert pathologist in a. The classification results are shown for CN-PI (c), CN-SW (d), CN-Ncut (e), and CN-SLIC (f)

Fig. 6.10 The histogram plotting of scores by our model and other five top models by different groups. Our result ranked the second (in green bar) as comparing to other results

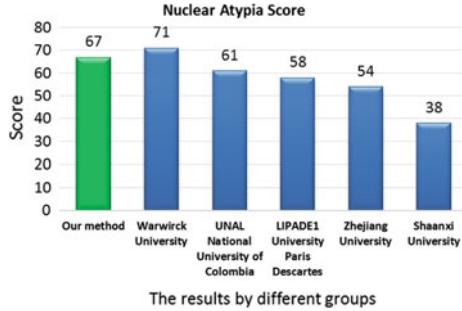
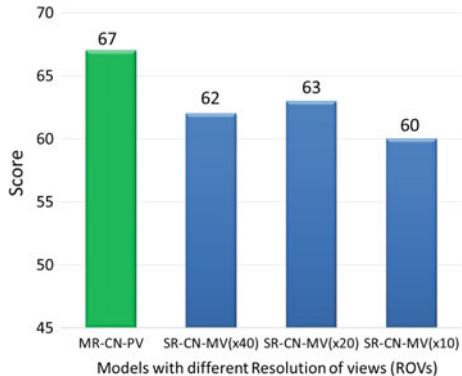


Fig. 6.11 The histogram plotting of nuclear atypia scoring by SR-CN-MV with three single resolution of views (ROVs) and MR-CN-PV across three ROVs



6.6.2 Quantitative Results

The quantitative performance of NAS for proposed MR-CN-PV model and other five top state-of-the-art methods are shown in Fig. 6.10. As the dataset did not provide the groundtruth for the slides in the testing set, the scores by proposed MR-CN-PV model and other state-of-the-art methods were provided by the organizer. Besides our score, the organizer also provided other five top scores by five groups that attended the contest. The proposed MR-CN-PV model got 67 points in the slides of testing set which ranks the second comparing with five top state-of-the-art methods. Moreover, the MR-CN-PV model is computationally efficient. The average computational time on each images with the resolution of $\times 10$, $\times 20$, and $\times 40$ are 1.2, 5.5, and 30s, respectively. The histogram plotting of scores in Fig. 6.10 suggests the effectiveness of proposed approach in automated NAS. Figure 6.11 shows the histogram plotting of three SR-CN-MV model on each resolution and MR-CN-PV model across three ROVs for NAS. The results also suggest the effectiveness of proposed MR-CN-PV model.

The quantitative performance for tissue segmentation for different models on D₂ are shown in Table 6.5. The results show that the CN-PI model outperforms the CN-PA models in discriminating two tissues.

Table 6.5 The quantitative evaluation of segmentation and classification results on the dataset with different models. The bolded numbers reflect the best performance for a specific performance measure within the data set

Models	Dataset	TPR	TNR	PPV	NPV	FPR	FDR	FNR	ACC	F1	MCC
CN-SW	NKI	77.95	80.68	81.63	76.86	19.32	18.37	22.05	79.25	79.25	58.56
	VGH	82.18	86.12	87.46	80.40	13.88	12.54	17.82	83.99	84.74	68.08
CN-Ncut	NKI	88.92	67.94	75.23	84.85	32.06	24.77	11.08	78.91	81.05	58.45
	VGH	89.37	86.63	88.67	87.42	13.37	11.31	10.63	88.11	89.03	76.05
CN-SLIC	NKI	86.31	82.15	84.11	84.60	17.85	15.89	13.66	84.34	85.21	68.60
	VGH	87.88	82.13	85.22	85.25	17.87	14.78	12.12	85.23	86.53	70.24
CN-PI	NKI	91.05	89.54	90.90	89.71	10.46	9.10	8.95	90.34	90.97	80.59
	VGH	95.44	93.41	91.95	96.29	6.59	8.06	4.56	94.30	93.66	88.54

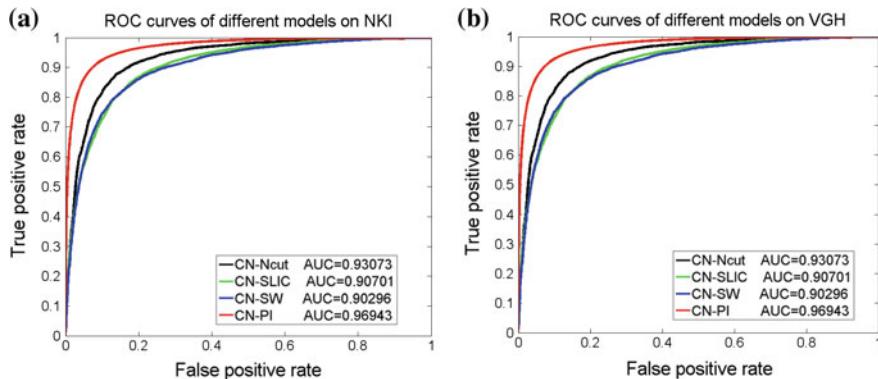


Fig. 6.12 The ROC curves for the different models (see Table 6.4) for detecting EP and ST regions on NKI (a) and VGH (b) data cohorts where AUC values of each models are shown in the figures

Figure 6.12a, b show the ROC curves corresponding to segmentation accuracy for different models on NKI (Fig. 6.12a) and VGH (Fig. 6.12b) of D₂. The AUC values suggest that the pixel-wise based CN outperform patch-wise based CN.

6.7 Concluding Remarks

In this chapter, we utilized deep convolutional neural network (CN) for two challenged problems: (1) Automated nuclear atypia scoring (NAS); (2) Epithelial (EP) and Stromal (ST) tissues discrimination on breast histopathology. For NAS, we integrated CN with two combination strategies. The proposed approach yielded good performance in automated NAS. This shows that the proposed approach can be applied in clinical routine procedure for automated NAS on histologic images. For epithelial and stromal tissues discrimination, we presented a pixel-wise CN-based model for segmentation of two tissues. Both qualitative and quantitative evaluation results show that the proposed model outperformed patch-wise CN based models.

Acknowledgements This work is supported by the National Natural Science Foundation of China (Nos. 61273259, 61272223); Six Major Talents Summit of Jiangsu Province (No. 2013-XXRJ-019), the Natural Science Foundation of Jiangsu Province of China (No. BK20141482), and Jiangsu Innovation & Entrepreneurship Group Talents Plan (No.JS201526).

References

1. Siegel RL, Miller KD, Jemal A (2015) Cancer statistics, 2015. CA Cancer J Clin 65(1):5–29
2. Chen W, Zheng R, Baade PD, Zhang S, Zeng H, Bray F, Jemal A, Yu XQ, He J (2016) Cancer statistics in china, 2015. CA: A Cancer J Clin 66(2):115–132

3. Rorke LB (1997) Pathologic diagnosis as the gold standard. *Cancer* 79(4):665–667
4. Rakha E, Reis-Filho J, Baehner F, Dabbs D, Decker T, Eusebi V, Fox S, Ichihara S, Jacquemier J, Lakhani S, Palacios J, Richardson A, Schnitt S, Schmitt F, Tan PH, Tse G, Badve S, Ellis I (2010) Breast cancer prognostic classification in the molecular era: the role of histological grade. *Breast Cancer Res* 12(4):207
5. Madabhushi A (2009) Digital pathology image analysis: opportunities and challenges. *Imaging Med* 1(1):7–10
6. De Wever O, Mareel M (2003) Role of tissue stroma in cancer cell invasion. *J Pathol* 200(4):429–447
7. Downey CL, Simpkins SA, White J, Holliday DL, Jones JL, Jordan LB, Kulkarni J, Pollock S, Rajan SS, Thygesen HH, Hamby AM, Speirs V (2014) The prognostic significance of tumour-stroma ratio in oestrogen receptor-positive breast cancer. *Br J Cancer* 110(7):1744–1747
8. Yuan Y et al (2012) Quantitative image analysis of cellular heterogeneity in breast tumors complements genomic profiling. *Sci Transl Med* 4(157):157ra143
9. Bourzac K (2013) Software: the computer will see you now. *Nature* 502(7473):S92–S94
10. Meyer JS, Alvarez C, Milikowski C, Olson N, Russo I, Russo J, Glass A, Zehnbauer BA, Lister K, Parwaresch R, Cooperative Breast Cancer Tissue Resource (2005) Breast carcinoma malignancy grading by bloom-richardson system vs proliferation index: reproducibility of grade and advantages of proliferation index. *Mod Pathol* 18(8):1067–1078
11. Robbins P, Pinder S, de Klerk N, Dawkins H, Harvey J, Sterrett G, Ellis I, Elston C (1995) Histological grading of breast carcinomas: a study of interobserver agreement. *Hum Pathol* 26(8):873–879
12. Brachtel E, Yagi Y (2012) Digital imaging in pathology-current applications and challenges. *J Biophotonics* 5(4):327–335
13. Gurcan MN, Boucheron LE, Can A, Madabhushi A, Rajpoot NM, Yener B (2009) Histopathological image analysis: a review. *IEEE Rev Biomed Eng* 2:147–171
14. Hamilton PW, Bankhead P, Wang YH, Hutchinson R, Kieran D, McArt DG, James J, Salto-Tellez M (2014) Digital pathology and image analysis in tissue biomarker research. *Methods* 70(1):59–73
15. Rimm DL (2011) C-path: a watson-like visit to the pathology lab. *Sci Transl Med* 3(108):108fs8
16. Anthimopoulos M, Christodoulidis S, Ebner L, Christe A, Mougiakakou S (2016) Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. *IEEE Trans Med Imaging* 35(5):1207–1216
17. Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans Med Imaging* 35(5):1285–1298
18. Albarqouni S, Baur C, Achilles F, Belagiannis V, Demirci S, Navab N (2016) AggNet: deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE Trans Med Imaging* 35(5):1313–1321
19. Pereira S, Pinto A, Alves V, Silva CA (2016) Brain tumor segmentation using convolutional neural networks in MRI images. *IEEE Trans Med Imaging* 35(5):1240–1251
20. Tajbakhsh N, Shin JY, Gurudu SR, Hurst RT, Kendall CB, Gotway MB, Liang J (2016) Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE Trans Med Imaging* 35(5):1299–1312
21. Moeskops P, Viergever MA, Mendrik AM, de Vries LS, Benders MJNL, Isgum I (2016) Automatic segmentation of MR brain images with a convolutional neural network. *IEEE Trans Med Imaging* 35(5):1252–1261
22. Setio AAA, Ciompi F, Litjens G, Gerke P, Jacobs C, van Riel SJ, Wille MMW, Naqibullah M, Snchez CI, van Ginneken B (2016) Pulmonary nodule detection in CT images: false positive reduction using multi-view convolutional networks. *IEEE Trans Med Imaging* 35(5):1160–1169
23. van Grinsven MJJP, van Ginneken B, Hoyng CB, Theelen T, Snchez CI (2016) Fast convolutional neural network training using selective data sampling: application to hemorrhage detection in color fundus images. *IEEE Trans Med Imaging* 35(5):1273–1284

24. Liskowski P, Krawiec K (2016) Segmenting retinal blood vessels with deep neural networks. *IEEE Trans Med Imaging* PP(99):1–1
25. Ciresan DC et al (2013) Mitosis detection in breast cancer histology images with deep neural networks. In: MICCAI 2013. LNCS, vol 8150. Springer, Berlin, pp 411–418
26. Xu J, Xiang L, Liu Q, Gilmore H, Wu J, Tang J, Madabhushi A (2016) Stacked sparse autoencoder (SSAE) for nuclei detection on breast cancer histopathology images. *IEEE Trans Med Imaging* 35(1):119–130
27. Sirinukunwattana K, Raza SEA, Tsang YW, Snead D, Cree IA, Rajpoot NM (2016) Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE Trans Med Imaging* 35:1196
28. Xing F, Xie Y, Yang L (2016) An automatic learning-based framework for robust nucleus segmentation. *IEEE Trans Med Imaging* 35(2):550–566
29. Xu J, Xiang L, Hang R, Wu J (2014) Stacked sparse autoencoder (SSAE) based framework for nuclei patch classification on breast cancer histopathology. In: ISBI
30. Xu J, Luo X, Wang G, Gilmore H, Madabhushi A (2016) A deep convolutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images. *Neurocomputing* 191:214–223
31. Cruz-Roa A et al (2013) A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In: MICCAI 2013, vol 8150. Springer, Berlin, pp 403–410
32. Cosatto E, Miller M, Graf HP, Meyer JS (2008) Grading nuclear pleomorphism on histological micrographs. In: 19th International conference on pattern recognition, ICPR 2008, pp 1–4
33. Lu C, Ji M, Ma Z, Mandal M (2015) Automated image analysis of nuclear atypia in high-power field histopathological image. *J Microsc* 258(3):233–240
34. Dalle JR, Li H, Huang CH, Leow WK, Racoceanu D, Putti TC (2009) Nuclear pleomorphism scoring by selective cell nuclei detection
35. Khan AM, Sirinukunwattana K, Rajpoot N (2015) A global covariance descriptor for nuclear atypia scoring in breast histopathology images. *IEEE J Biomed Health Inform* 19(5):1637–1647
36. Shi J, Wu J, Li Y, Zhang Q, Ying S (2016) Histopathological image classification with color pattern random binary hashing based PCANet and matrix-form classifier. *IEEE J Biomed Health Inform* PP(99):1–1
37. Basavanhally A, Feldman MD, Shih N, Mies C, Tomaszewski J, Ganesan S, Madabhushi A (2011) Multi-field-of-view strategy for image-based outcome prediction of multi-parametric estrogen receptor-positive breast cancer histopathology: comparison to oncotype DX. *J Pathol Inform* 2 (01/2012 2011)
38. Basavanhally A, Ganesan S, Feldman M, Shih N, Mies C, Tomaszewski J, Madabhushi A (2013) Multi-field-of-view framework for distinguishing tumor grade in ER+ breast cancer from entire histopathology slides. *IEEE Trans Biomed Eng* 60(8):2089–99
39. Linder N et al (2012) Identification of tumor epithelium and stroma in tissue microarrays using texture analysis. *Diagn Pathol* 7(1):22
40. Bianconi F, lvarez Larri A, Fernndez A (2015) Discrimination between tumour epithelium and stroma via perception-based features. *Neurocomputing* 154(0):119–126
41. Hiary H et al (2013) Automated segmentation of stromal tissue in histology images using a voting bayesian model. *Signal Image Video Process* 7(6):1229–1237
42. Eramian M et al (2011) Segmentation of epithelium in H&E stained odontogenic cysts. *J Microsc* 244(3):273–292
43. Signolle N, Revenu M, Plancoulaine B, Herlin P (2010) Wavelet-based multiscale texture segmentation: application to stromal compartment characterization on virtual slides. *Signal Process* 90(8):2412–2422 Special Section on Processing and Analysis of High-Dimensional Masses of Image and Signal Data
44. Lahrmann B, Halama N, Sinn HP, Schirmacher P, Jaeger D, Grabe N (2011) Automatic tumor-stroma separation in fluorescence tmas enables the quantitative high-throughput analysis of multiple cancer biomarkers. *PLoS ONE* 6(12):e28048

45. Amaral T, McKenna S, Robertson K, Thompson A (2013) Classification and immunohistochemical scoring of breast tissue microarray spots. *IEEE Trans Biomed Eng* 60(10):2806–2814
46. Kather JN, Weis CA, Bianconi F, Melchers SM, Schad LR, Gaiser T, Marx A, Zilner FG (2016) Multi-class texture analysis in colorectal cancer histology. *Sci Rep* 6:27988
47. Bychkov D, Turkki R, Haglund C, Linder N, Lundin J (2016) Deep learning for tissue microarray image-based outcome prediction in patients with colorectal cancer, vol 9791, pp 979115–979115–6
48. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
49. Janowczyk A, Madabhushi A (2016) Deep learning for digital pathology image analysis: a comprehensive tutorial with selected use cases. *J Pathol Inform* 7(1):29
50. Zhou ZH (2016) Machine learning. Tsinghua University Press, Beijing
51. ICPR2014 (2010) MITOS & ATYPIA 14 contest. <https://grand-challenge.org/site/mitos-atypia-14/dataset/>. Accessed 30 Sept 2010
52. Beck AH et al (2011) Systematic analysis of breast cancer morphology uncovers stromal features associated with survival. *Sci Transl Med* 3(108):108ra113
53. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the ACM international conference on multimedia. ACM, pp 675–678

Chapter 7

Interstitial Lung Diseases via Deep Convolutional Neural Networks: Segmentation Label Propagation, Unordered Pooling and Cross-Dataset Learning

Mingchen Gao, Ziyue Xu and Daniel J. Mollura

Abstract Holistically detecting interstitial lung disease (ILD) patterns from CT images is challenging yet clinically important. Unfortunately, most existing solutions rely on manually provided regions of interest, limiting their clinical usefulness. We focus on two challenges currently existing in two publicly available datasets. First of all, missed labeling of regions of interest is a common issue in existing medical image datasets due to the labor-intensive nature of the annotation task which requires high levels of clinical proficiency. Second, no work has yet focused on predicting more than one ILD from the same CT slice, despite the frequency of such occurrences. To address these limitations, we propose three algorithms based on deep convolutional neural networks (CNNs). The differences between the two main publicly available datasets are discussed as well.

7.1 Introduction

Interstitial lung disease (ILD) refers to a group of more than 150 chronic lung diseases that causes progressive scarring of lung tissues and eventually impairs breathing. The gold standard imaging modality for diagnosing ILD patterns is high-resolution computed tomography (HRCT) [1, 2]. Figures 7.1 and 7.2 depict examples of the most typical ILD patterns.

Automatically detecting ILD patterns from HRCT images would help the diagnosis and treatment of this morbidity. The majority of previous work on ILD detection is limited to patch-level classification, which classifies small patches from manu-

M. Gao (✉) · Z. Xu · D.J. Mollura
Department of Radiology and Imaging Sciences,
National Institutes of Health (NIH), Bethesda, MD 20892, USA
e-mail: mingchen.gao@nih.gov

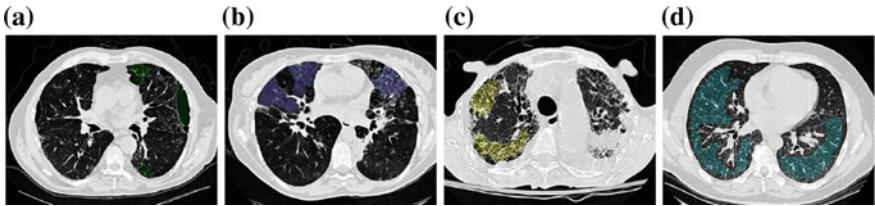


Fig. 7.1 Visual aspects of the most common lung tissue patterns in HRCT axial slices in UHG dataset. Infected regions are annotated with different colors in the publicly available dataset [1]. **a** Emphysema (EM). **b** Ground Glass (GG). **c** Fibrosis (FB). **d** Micronodules (MN)

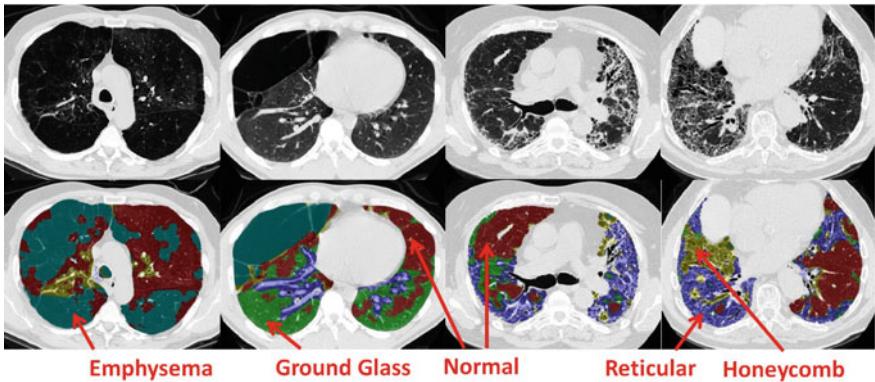


Fig. 7.2 Examples of ILD patterns. Every voxel in the lung region is labeled as healthy or one of the four ILDs: ground glass, reticular, honeycomb or emphysema. The *first row* is the lung CT images. The *second row* is their corresponding labelings

ally generated regions of interest (ROIs) into one of the ILDs. Approaches include restricted Boltzmann machines [3], convolutional neural networks (CNNs) [4], local binary patterns [5, 6] and multiple instance learning [7]. An exception to the patch-based approach is the recent work of Gao et al. [8], which investigated a clinically more realistic scenario for ILD classification, assigning a *single* ILD label to any holistic two-dimensional axial CT slice without any pre-processing or segmentation. Although holistic detection is more clinically desirable, the underlying problem is much harder without knowing the ILD locations and regions *a priori*. The difficulties lie on several aspects, which include the tremendous amount of variation in disease appearance, location, and configuration and also the expense required to obtain delicate pixel-level ILD annotations of large datasets for training.

We would like to tackle these challenges from several aspects. There are two main publicly available datasets for CT imaging based ILD classification [1, 2]. The first one is from University Hospital of Geneva (UHG), with limited annotations [1]. As

shown in Fig. 7.1, we find that only less than 15% of the lung region in the pixel coverage measure is labeled, which significantly restricts the number of available training image pixels, patches or data. Assigning semantic labels to each pixel of a CT image is tedious, time-consuming and error-prone, or simply is not affordable and feasible for a large amount of patients. Supervised learning, the most common technique for integrating domain knowledge, usually needs the manual annotation from expensive medical experts to assign a label to every pixel. This hinders the learning scalability both in the amount of training data and in the number of classes. On the other hand, we have witnessed the success of many applications in computer vision and medical imaging analysis when a large-scale well-annotated dataset is available [9].

Therefore automated image annotation or labeling methods are needed to assist doctors during the labeling process. In an ideal framework, computerized algorithms would complete most of the tedious tasks, and doctors would merely validate and fine-tune the results, if necessary. We propose a segmentation propagation algorithm that combines the cues from the initial or partial manual annotations, deep convolutional neural networks (CNN) based single pixel classification and formulate into a constrained dense fully connected conditional random field (CRF) framework [10]. Our main technical novelties are the **constrained unary** (*manually labeled pixels are hard-enforced with their original ILD image labels; pixels outside of lung are considered as hard-encoded background; unlabeled lung pixels are the key subjects to be assigned ILD labels using our method*) and **pairwise terms** (*message passing is only allowed for any pair of lung image pixels*) and their efficient implementation in [11]. The proposed method is applicable to other problems as a generic semi-supervised image segmentation solution. This work is partially inspired by interactive graph-cut image segmentation [12] and automatic population of pixelwise object-background segmentation from manual annotations on ImageNet database [13].

Another challenge we would like to solve is detecting multiple ILDs simultaneously without the locations which has not been addressed by previous studies [3, 4, 8, 14], including that of Gao et al. [8], which all treat ILD detection as a single-label classification problem. When analyzing the Lung Tissue Research Consortium (LTRC) dataset [2], the most comprehensive lung disease image database with detailed annotated segmentation masks, we found that there are significant amounts of CT slices associated with two or more ILD labels. For this reason, and partially inspired by the recent natural image classification work [15], we model the problem as multi-label regression and solve it using a CNN [16]. We note that multi-label regression has also been used outside of ILD contexts for heart chamber volume estimation [17, 18]. However, this prior work used hand-crafted features and random-forest-based regression, whereas we employ learned CNN-based features, which have enjoyed dramatic success in recent years over hand-crafted variants [9]. Thus, unlike prior ILD detection work [3–6, 8], our goal is to detect multiple ILDs on holistic CT slices simultaneously, providing a more clinically useful tool.

While CNNs are powerful tools, their feature learning strategy is not invariant to the spatial locations of objects or textures within a scene. This order-sensitive feature encoding, reflecting the spatial layout of the local image descriptors, is effective in

object and scene recognition. However, it may not be beneficial or even be counter-productive for texture classification. The spatial encoding of order-sensitive image descriptors can be discarded via unordered feature encoders such as bag of visual words (BoVW), Fisher vectors (FV) [19], or aggregated by order-sensitive spatial pyramid matching (SPM). Given the above considerations, we enhance our CNN-regression approach using spatial-invariant encodings of feature activations for multi-label multi-class ILD detection.

7.2 Methods

Our algorithms are of two respects. The first one would extend the limited labels in the UHG dataset to every pixel in the lung region. Specifically, we explore the possible ways to **propagate** the ILD labels from the limited manually drawn regions to the whole lung slice as a **per-pixel multi-class image segmentation and labeling**. The fully connected conditional random field builds the pairwise potentials densely on all pairs of pixels in the image. The CRF optimization is conducted as message passing that can naturally handle multi-class labeling. The CRF unary energies are learned from CNN-based image patch labeling. Ground truth labels by radiologists are also integrated into the CRF as hard constraints. The proposed algorithm is evaluated on a publicly available dataset [1] and the segmentation/labeling results are validation by an expert radiologist.

The second method focuses on predicting multiple labels simultaneously on the same slice and is tested on the LTRC dataset. We propose two variations of multi-label deep convolutional neural network regression (MLCNN-R) models to address the aforementioned challenges. First, an end-to-end CNN network is trained for multi-label image regression. The loss functions are minimized to estimate the actual pixel numbers occupied per ILD class or the binary [0,1] occurring status. Second, the convolutional activation feature maps at different network depths are spatially aggregated and encoded through the FV [19] method. This encoding removes the spatial configurations of the convolutional activations and turns them into location-invariant representations. This type of CNN is also referred as FV-CNN. The unordered features are then trained using a multivariate linear regressor (Mvregress function in MATLAB) to regress the numbers of ILD pixels or binary labels. Our proposed algorithm is demonstrated using the LTRC ILD dataset [2], composed of 533 patients. Our experiments use fivefold cross-validation (CV) to detect the most common ILD classes of ground glass, reticular, honeycomb and emphysema. Experimental results demonstrate the success of our approach in tackling the challenging problem of multi-label multi-class ILD classification.

7.2.1 Segmentation Label Propagation

We formulate the segmentation problem as a maximum a posteriori (MAP) inference in a CRF defined over pixels. To take into account of long-range image interactions, an efficient fully connected CRF method is adapted [11].

The CRF representation captures the conditional distribution of the class labeling X given an image I . Consider a random field X defined over a set of variables $\{X_1, \dots, X_N\}$, with $X_i \in X$ being associated with every pixel $i \in V$ and taking a value from the label set $L = \{l_1, \dots, l_K\}$ of label categories. The labeling of X from images is obtained with a maximum a posterior (MAP) estimation of the following conditional log-likelihood:

$$E(x) = \sum_i \psi_u(x_i) + \sum_{i < j} \psi_p(x_i, x_j), \quad (7.1)$$

where i and j range from 1 to N . $\psi_u(x_i)$, the unary potential, is computed independently by the convolutional neural network classifier for each pixel/patch. The pairwise potentials in our model have the form

$$\begin{aligned} \psi_p(x_i, x_j) &= u(x_i, x_j) \sum_{m=1}^K k(f_i, f_j) \\ &= u(x_i, x_j) \sum_{m=1}^K \omega^{(m)} k^{(m)}(f_i, f_j). \end{aligned} \quad (7.2)$$

Each $k^{(m)}$ is a Gaussian kernel

$$k^{(m)}(f_i, f_j) = \exp(-\frac{1}{2}(f_i - f_j)^T \Lambda^{(m)}(f_i - f_j)), \quad (7.3)$$

where the vectors f_i and f_j are feature vectors for pixels i and j in an arbitrary feature space; u is a label compatibility function; and $\omega^{(m)}$ are linear combination weights.

In our implementation, we use two-kernel potentials, defined in terms of the CT attenuation vectors I_i and I_j (introduced in [8]) and positions p_i and p_j :

$$\begin{aligned} k(f_i, f_j) &= \omega^{(1)} \exp\left(-\frac{|p_i - p_j|^2}{2\theta_\alpha^2} - \frac{|I_i - I_j|^2}{2\theta_\beta^2}\right) \\ &\quad + \omega^{(2)} \exp\left(-\frac{|p_i - p_j|^2}{2\theta_\gamma^2}\right). \end{aligned} \quad (7.4)$$

The first term presents the appearance kernel, which represents the affinities of nearby pixels with similar CT attenuation patterns. The second term presents the smoothness

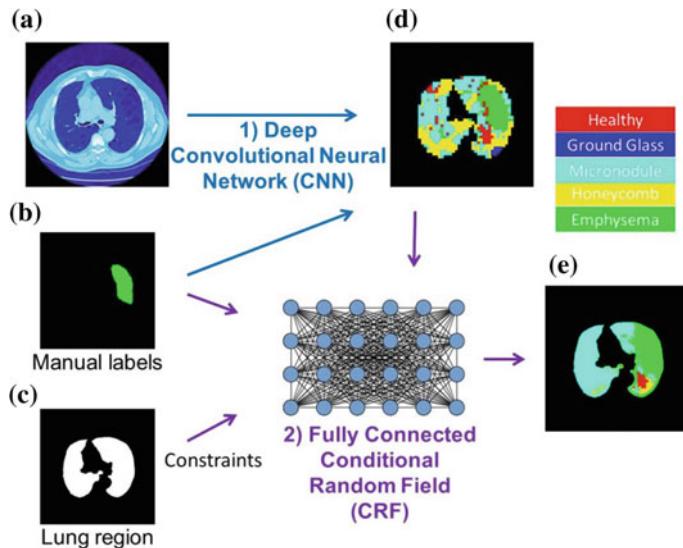


Fig. 7.3 Major intermediate results. **a** Three channels of different HU windows illustrated as RGB values. **b** Annotated ROI. **c** Annotated lung mask. **d** CNN classifier at a spatial interval of 10 pixels. **e** Final result integrating image features, unary prediction and hard constraints

kernel, which removes small isolated regions. The parameters θ_α , θ_β and θ_γ are used to control the degree of nearness and similarity. The inference of fully connected conditional random field is efficiently approximated by an iterative message passing algorithm. Each iteration performs a message passing, a compatibility transform and a local update. The message passing can be performed using Gaussian filtering in feature space. The complexity of the algorithm reduces from quadratic to linear in the number of variables N and sublinear in the number of edges in the model (Fig. 7.3).

Unary classifier using Convolutional Neural Network: At present, there is a vast amount of relevant work on computerized ILD pattern classification. The majority focuses on image patch based classification using hand-crafted [5, 6, 20] or CNN learned features [8]. We use the CNN-based CRF unary classifier because of its state-of-the-art performance: classification accuracy of 87.9% reported in [8]. To facilitate comparison, five common ILD patterns are studied in this work: healthy, emphysema, ground glass, fibrosis and micronodules (Fig. 7.4). Image patches of size 32×32 pixels within the ROI annotations of these five classes, are extracted to train a deep CNN classifier. The well known CNN AlexNet model [9] trained on ImageNet is used to fine-tune on our image patch dataset. 32×32 pixel images patches are rescaled to 224×224 and three channels of different HU windows [8] are generated to accommodate the CNN model.

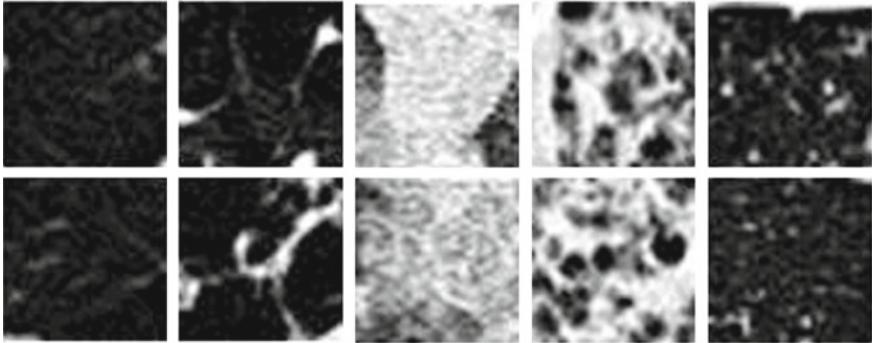


Fig. 7.4 Examples of 32×32 patches for each ILD category. From *left* to *rights* columns: healthy, emphysema, ground glass, fibrosis and micronodules

Hard Constraints: The image labels given by radiologists from the dataset [1] are considered as ground truth and ought to be strictly enforced. During each CRF message passing iteration, the hard constrained image regions are hard-reset to be consistent with their ground truth labels. In such cases, there is only message passing out of the hard constrained regions towards unlabeled lung image pixels. On the other hand, we assume that the ILD label map should only be inferred within the lung field. The lung field CRF ILD labeling is conditionally independent of image pixel patterns outside the lung mask. In implementation of Eq. 7.4, the parameters θ_α , θ_β and θ_γ are set to be a small constant (0.001) for any pixel pairs linking lung and non-lung spatial indexes (p_i, p_j) so the associated $k(f_i, f_j)$ has a numerically vanishing value, which is equivalent to no message passing.

Specifically, we explore the possible ways to **propagate** the ILD labels from the limited manually drawn regions to the whole lung slice as a **per-pixel multi-class image segmentation and labeling**. The fully connected conditional random field builds the pairwise potentials densely on all pairs of pixels in the image. The CRF optimization is conducted as message passing that can naturally handle multi-class labeling. The CRF unary energies are learned from CNN based image patch labeling. Ground truth labels by radiologists are also integrated into the CRF as hard constraints. The proposed algorithm is evaluated on a publicly available dataset [1] and the segmentation/labeling results are validation by an expert radiologist.

7.2.2 Multi-label ILD Regression

Our algorithm contains two major components: (1) we present a squared L_2 loss function based multi-label deep CNN regression method to estimate either the observable ILD areas (in the numbers of pixels), or the binary [0,1] status of “non-appearing” or “appearing”. This regression-based approach allows our algorithm to naturally

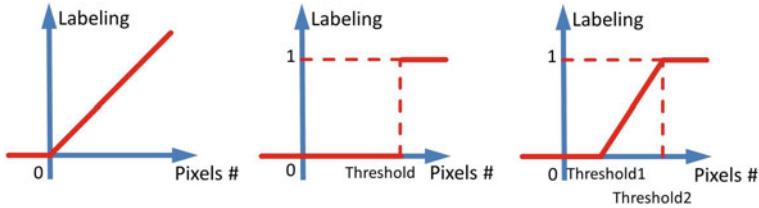


Fig. 7.5 Three functions for mapping the number of pixels to the regression label

preserve the co-occurrence property of ILDs in CT imaging. (2) CNN activation vectors are extracted from convolutional layers at different depths of the network and integrated using a Fisher vector feature encoding scheme in a spatially unordered manner, allowing us to achieve a location-invariant deep texture description. ILD classes are then discriminated using multivariate linear regression.

CNN Architecture: Deep CNN regression is used to calculate the presence or the area of spatial occupancy for IDL in the image, where multiple pathology patterns can co-exist. The squared L_2 loss function is adopted for regression [15] instead of the more widely used softmax or logistic-regression loss for CNN-based classification [4, 8, 9]. There are multiple ways to model the regression labels for each image. One straightforward scheme is to count the total number of pixels annotated per disease to represent its severity, e.g., Fig. 7.5 **left**. We can also use a step function to represent the presence or absence of the disease, as shown in Fig. 7.5 **middle**, where the stage threshold T may be defined using clinical knowledge. For any ILD in an image, if its pixel number is larger than T , the label is set to be 1; otherwise as 0. A more sophisticated model would have a piecewise linear transform function, mapping the pixel numbers towards the range of [0,1] (Fig. 7.5 **right**). We test all approaches in our experiments.

Suppose that there are N images and c types of ILD patterns to be detected or classified, the label vector of the i^{th} image is represented as a c -length multivariate vector $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{ic}]$. An all-zero labeling vector indicates that the slice is healthy or has no targeted ILD found based on the ground truth annotation. The L_2 cost function to be minimized is defined as

$$L(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \sum_{i=1}^N \sum_{k=1}^c (y_{ik} - \hat{y}_{ik})^2, \quad (7.5)$$

There are several successful CNN structures from previous work, such as AlexNet [9] and VGGNet [21]. We employ a variation of AlexNet, called **CNN-F** [22], for a trade-off between efficiency and performance based on the amount of available annotated image data. **CNN-F** contains five convolutional layers, followed by two fully connected (FC) layers. We set the last layer to the squared L_2 loss function. Four classes of ILDs are investigated in our experiments: ground glass, reticular, honeycomb and emphysema (other classes have too few examples in the LTRC

database [2]). The length of y_i is $c = 4$ to represent these four ILD classes. Based on our experience, random initialization of the CNN parameters worked better than ImageNet pre-trained models. Model parameters were optimized using stochastic gradient descent.

Unordered Pooling Regression via Fisher Vector Encoding: In addition to CNN-based regression, we also test a spatially invariant encoding of CNN feature activations. We treat the output of each k -th convolutional layer as a 3D descriptor field $X_k \in \mathbb{R}^{W_k \times H_k \times D_k}$, where W_k and H_k are the width and height of the field and D_k is the number of feature channels. Therefore, the whole deep feature activation map is represented by $W_k \times H_k$ feature vectors and each feature vector is of dimension D_k .

We then invoke FV encoding [19] to remove the spatial configurations of total $W_k \times H_k$ vectors per activation map. Following [19], each descriptor $x_i \in X_k$ is soft-quantized using a Gaussian mixture model. The first- and second-order differences $(u_{i,m}^T, v_{i,m}^T)$ between any descriptor x_i and each of the Gaussian cluster mean vectors $\{\mu_m\}$, $m = 1, 2, \dots, M$ are accumulated in a $2MD_k$ -dimensional image representation:

$$f_i^{FV} = [u_{i,1}^T, v_{i,1}^T, \dots, u_{i,M}^T, v_{i,M}^T]^T. \quad (7.6)$$

The resulting FV feature encoding results in very high $2MD_k$ (e.g., $M = 32$ and $D_k = 256$) dimensionality for deep features of X_k . For computational and memory efficiency, we adopt principal component analysis (PCA) to reduce the f_i^{FV} features to a lower-dimensional parameter space. Based on the ground truth label vectors y_i , multivariate linear regression is used to predict the presence or non-presence of ILDs using the low-dimensional image features $PCA(f_i^{FV})$.

7.3 Experiments and Discussion

The proposed algorithms are testing on two datasets, UHG and LTRC, respectively. The training folds and testing fold are split at the patient level to prevent overfitting (i.e., no CT slices from the same patient are used for both training and validation). CNN training was performed in MATLAB using MatConvNet [23] and was run on a PC with an Nvidia Tesla K40 GPU.

7.3.1 Segmentation Label Propagation

UHG dataset [1] is used for training and validation under twofold cross-validation for the segmentation propagation problem. ROIs of total 17 different lung patterns and lung masks are also provided along with the dataset. Figure 7.6 shows the annotation provided by the dataset, the labeling obtained from our algorithm and the ground

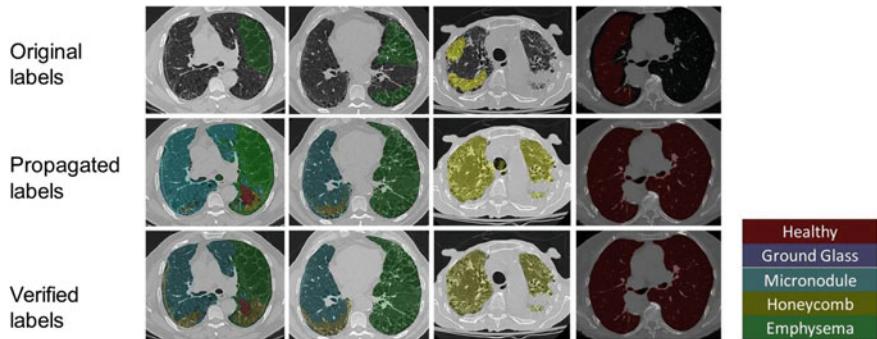


Fig. 7.6 Comparison between the annotation provided by the UHG dataset, our labeling results, and the final annotation verified by experienced radiologists

Table 7.1 Confusion matrix, precision, recall and F-score of ILD pattern labeling

Ground truth	Prediction				
	NM	EM	GG	FB	MN
NM	0.9792	0.0067	0.0029	0.0020	0.0092
EM	0.2147	0.7389	0	0.0170	0.0294
GG	0	0	1.0000	0	0
FB	0.0414	0.0271	0.0118	0.8046	0.1151
MN	0.0007	0.0013	0.0174	0.0058	0.9748
Precision	0.9500	0.9320	0.8175	0.9060	0.9666
Recall	0.9792	0.7389	1	0.8046	0.9748
F-score	0.9644	0.8243	0.8996	0.8523	0.9707

truth validated by radiologists. In our implementation, we use the lung mask provided from the dataset. Please note that trachea is included in the lung mask provided from the dataset. This misleads our algorithm to give a prediction in the trachea region. A recent rough lung segmentation method [24] can be used to automate this process.

Quantitative evaluation is given in Table 7.1 with the total accuracy reaching 92.8%. More importantly, the amount of auto-annotated pixels is 7.8 times greater than the amount of provided annotation [1]. Thus the labeled training dataset [1] is significantly enlarged via segmentation label propagation. This data expansion is a critical contribution of this paper. The CRF solver is implemented in C++. The most time consuming part is the unary classification of densely sampled image patches. To speed up testing, a relatively coarse prediction map of image patches is sufficient. This map can be bi-linearly interpolated and later refined by the CRF pairwise constraints. In our implementation, we predict the labels of image patches at a spatial interval of 10 pixels. Parameters θ_α , θ_β and θ_γ are set to be 80, 13, and 3 through a small calibration dataset within training. We set $\omega^{(1)} = \omega^{(2)} = 1$, which is found to work well in practice.

7.3.2 Multi-label ILD Regression

LTRC dataset [2] enjoys complete ILD labeling at the CT slice level [10]. We use the LTRC dataset to evaluate the algorithm detecting multiple labels simultaneously. Every pixel in the CT lung region is labeled as healthy or one of the four tissue types: ground glass, reticular, honeycomb or emphysema. Only 2D axial slices are investigated here, without taking successive slices into consideration. Many CT scans for ILD study have large inter-slice distances (for example 10 mm in [1]) between axial slices, making direct 3D volumetric analysis implausible. The original resolution of the 2D axial slices is 512×512 pixels. All images are resized to the uniform size of 214×214 pixels.

To conduct holistic slice based ILD classification [8], we first convert the pixelwise labeling into slice-level labels. There are 18883 slices in total for training and testing. Without loss of generality, if we set $T = 6000$ pixels as the threshold to differentiate the presence or absence of ILDs, there are 3368, 1606, 1247 and 2639 positive slices for each disease, respectively. In total there are 11677 healthy CT images, 5675 images with one disease, 1410 images with two diseases, 119 images with three diseases, and 2 images with four diseases. We treat the continuous values after regression (in two types of pixel numbers or binary status) as “classification confidence scores”. We evaluate our method by comparing against ground truth ILD labels obtained from our chosen threshold.

Each ILD pattern is evaluated separately by thresholding the “classification confidence scores” from our regression models to make the binary presence or absence decisions. Classification receiver operating characteristic (ROC) curves can be generated in this manner. We experimented with Fig. 7.5’s three labeling converting functions. Regression using the ILD occupied pixel numbers or the binary status labels produced similar quantitative ILD classification results. However, the piecewise linear transformation did not perform well.

When constructing the FV-encoded features, f_i^{FV} , the local convolutional image descriptors are pooled into 32 Gaussian components, producing dimensionalities as high as 16K dimensions [19]. We further reduce the FV features to 512 dimensions using PCA. Performance was empirically found to be insensitive to the number of Gaussian kernels and the dimensions after PCA.

All quantitative experiments are performed under fivefold cross-validation. The training folds and testing fold are split at the patient level to prevent overfitting (i.e., no CT slices from the same patient are used for both training and validation). CNN training was performed in MATLAB using MatConvNet [23] and was run on a PC with an Nvidia Tesla K40 GPU. The training for one fold takes hours. The testing could be accomplished in seconds per image.

We show the ROC results directly regressed to the numbers of ILD pixels in Fig. 7.7. The area under the curve (AUC) values are marked in the plots. In Fig. 7.7d, AUC scores are compared among configurations using FV encoding on deep image features pooled from different CNN convolutional layers. Using activations based on the first fully connected layer (fc6) are also evaluated. Corresponding quantitative

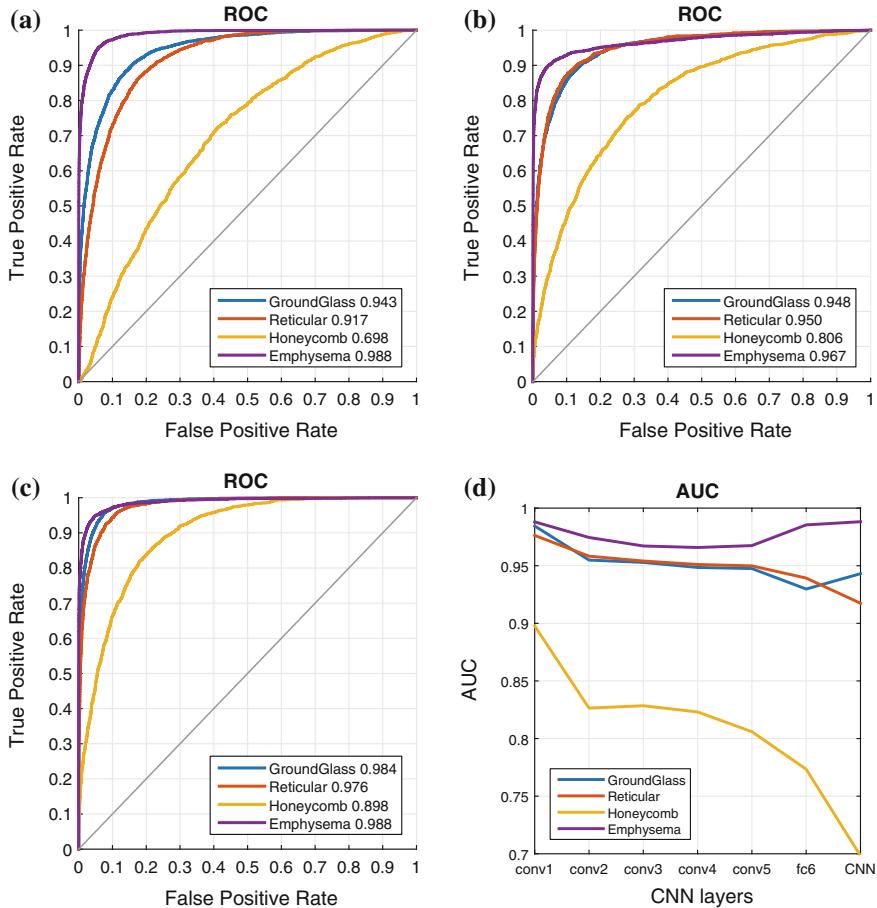


Fig. 7.7 ILD detection results shown in ROC curves. Both CNN and FV-CNN regression are used to regress to the numbers of pixels. **a** Detection results of CNN regression. **b, c** Detection results of FV-CNN via the unordered feature pooling using conv5 and conv1 layer, respectively. **d** AUC versus FV pooling at different convolutional layers

results are shown in Table 7.2. Both deep regression models achieve high AUC values for all four major ILD patterns. FV unordered pooling operating on the first CNN convolutional layer **conv1** produces the overall best quantitative results, especially for Honeycomb. Despite residing in the first layer, the filters and activations on **conv1** are still part of a deep network since they are learned through back-propagation. Based on these results, this finding indicates that using FV encoding with deeply learned **conv1** filter activations is an effective approach to ILD classification.

Figure 7.8 presents some examples of successful and misclassified results. First four cases of examples are well successfully detected all types of ILD patterns. In the second to last, although it is marked as misclassified (compared to the ground truth

Table 7.2 Quantitative results comparing the AUC between different layers. Both CNN and multi-variate linear regression regress to pixel numbers

Disease	Area under the curve (AUC)						
	conv1	conv2	conv3	conv4	conv5	fc6	CNN
Ground glass	0.984	0.955	0.953	0.948	0.948	0.930	0.943
Reticular	0.976	0.958	0.954	0.951	0.950	0.939	0.917
Honeycomb	0.898	0.826	0.828	0.823	0.806	0.773	0.698
Emphysema	0.988	0.975	0.967	0.966	0.967	0.985	0.988

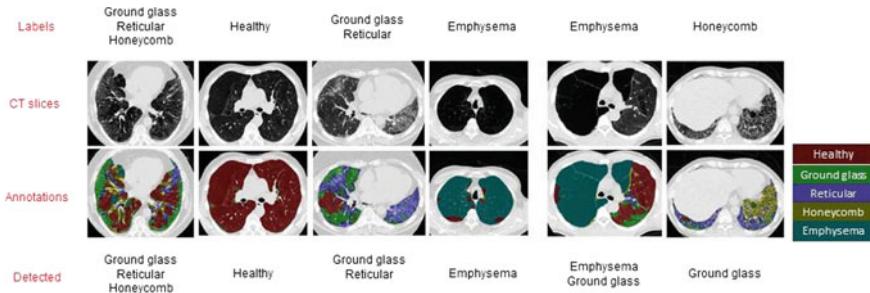


Fig. 7.8 Examples of correctly detected and misclassified ILD slices

binary labels with $T = 6000$ pixels), our method finds and classifies emphysema and ground glass correctly that do occupy some image regions. These qualitative results visually confirm the high performance demonstrated by our quantitative experiments.

7.4 Conclusion

In this work, we present several solutions related to ILD pattern detections. The first segmentation label propagation method efficiently populates the labels from the annotated regions to the whole CT image slices. High segmentation/labeling accuracy are achieved. The amount of labeled training data in [1] is significantly expanded and will be publicly shared upon publication¹.

We also present a new ILD pattern detection algorithm using multi-label CNN regression combined with unordered pooling of the resulting features. In contrast to previous methods, our method can perform multi-label multi-class ILD detection. Moreover, this is performed without the manual ROI inputs needed by much of the state-of-the-art [3–5]. We validate on a publicly available dataset of 533 patients using five-fold CV, achieving high AUC scores of 0.982, 0.972, 0.893 and 0.993 for Ground-Glass, Reticular, Honeycomb and Emphysema, respectively. Future work includes

¹<http://www.research.rutgers.edu/minggao>.

performing cross-dataset learning and incorporating weakly supervised approaches to obtain more labeled training data. Nonetheless, as the first demonstration of effective multi-class ILD classification, this work represents an important contribution toward clinically effective CAD solutions.

References

1. Depeursinge A, Vargas A, Platon A, Geissbuhler A, Poletti P-A, Müller H (2012) Building a reference multimedia database for interstitial lung diseases. CMIG 36(3):227–238
2. Holmes III D, Bartholmai B, Karwoski R, Zavaleta V, Robb R (2006) The lung tissue research consortium: an extensive open database containing histological, clinical, and radiological data to study chronic lung disease. *Insight J*
3. van Tulder G, de Brujne M (2016) Combining generative and discriminative representation learning for lung CT analysis with convolutional restricted Boltzmann machines. *TMI*
4. Anthimopoulos M, Christodoulidis S, Ebner L, Christe A, Mougiakakou S (2016) Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. *IEEE Trans Med Imaging* 35(5):1207–1216
5. Song Y, Cai W, Huang H, Zhou Y, Feng D, Wang Y, Fulham M, Chen M (2015) Large margin local estimate with applications to medical image classification. *TMI* 34(6):1362–1377
6. Song Y, Cai W, Zhou Y, Feng DD (2013) Feature-based image patch approximation for lung tissue classification. *TMI* 32(4):797–808
7. Hofmanninger J, Langs G (2015) Mapping visual features to semantic profiles for retrieval in medical imaging. In: CVPR, pp 457–465
8. Gao M, Bagci U, Lu L, Wu A, Buty M, Shin H-C, Roth H, Papadakis GZ, Depeursinge A, Summers RM, et al (2016) Holistic classification of CT attenuation patterns for interstitial lung diseases via deep convolutional neural networks. *Comput Methods Biomed Eng Imaging Vis* 1–6
9. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: NIPS, pp 1097–1105
10. Gao M, Xu Z, Lu L, Nogues I, Summers R, Mollura D (2016) Segmentation label propagation using deep convolutional neural networks and dense conditional random field. In: IEEE international symposium on biomedical imaging
11. Krähenbühl P, Koltun V (2011) Efficient inference in fully connected CRFs with Gaussian edge potentials. In: NIPS, pp 109–117
12. Boykov YY, Jolly M-P (2001) Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In: Eighth IEEE international conference on proceedings, vol 1. IEEE, pp 105–112
13. Guillaumin M, Küttel D, Ferrari V (2014) Imagenet auto-annotation with segmentation propagation. *IJCV* 110(3):328–348
14. Gong Y, Wang L, Guo R, Lazebnik S (2014) Multi-scale orderless pooling of deep convolutional activation features. In: ECCV 2014. Springer, pp 392–407
15. Wei Y, Xia W, Huang J, Ni B, Dong J, Zhao Y, Yan S (2014) CNN: single-label to multi-label. arXiv preprint [arXiv:1406.5726](https://arxiv.org/abs/1406.5726)
16. Gao M, Xu Z, Lu L, Harrison AP, Summers RM, Mollura DJ (2016) Multi-label deep regression and unordered pooling for holistic interstitial lung disease pattern detection. Machine learning in medical imaging
17. Zhen X, Islam A, Bhaduri M, Chan I, Li S (2015) Direct and simultaneous four-chamber volume estimation by multi-output regression. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 669–676
18. Zhen X, Wang Z, Islam A, Bhaduri M, Chan I, Li S (2014) Direct estimation of cardiac biventricular volumes with regression forests. In: MICCAI, pp 586–593

19. Perronnin F, Sánchez J, Mensink T (2010) Improving the fisher kernel for large-scale image classification. In: ECCV, pp 143–156
20. Depeursinge A, Van de Ville D, Platon A, Geissbuhler A, Poletti P-A, Muller H (2012) Near-affine-invariant texture learning for lung tissue analysis using isotropic wavelet frames. IEEE Trans Inf Technol Biomed 16(4):665–675
21. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
22. Chatfield K, Simonyan K, Vedaldi A, Zisserman A (2014) Return of the devil in the details: delving deep into convolutional nets. arXiv preprint [arXiv:1405.3531](https://arxiv.org/abs/1405.3531)
23. Vedaldi A, Lenc K (2015) MatConvNet: convolutional neural networks for MATLAB. In: Proceedings of the 23rd annual ACM conference on multimedia conference. ACM, pp 689–692
24. Mansoor A, Bagci U, Xu Z, Foster B, Olivier KN, Elinoff JM, Suffredini AF, Udupa JK, Mollura DJ (2014) A generic approach to pathological lung segmentation

Chapter 8

Three Aspects on Using Convolutional Neural Networks for Computer-Aided Detection in Medical Imaging

**Hoo-Chang Shin, Holger R. Roth, Mingchen Gao, Le Lu,
Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura
and Ronald M. Summers**

Abstract Deep convolutional neural networks (CNNs) enable learning trainable, highly representative and hierarchical image feature from sufficient training data which makes rapid progress in computer vision possible. There are currently three major techniques that successfully employ CNNs to medical image classification: training the CNN from scratch, using off-the-shelf pretrained CNN features, and transfer learning, i.e., fine-tuning CNN models pretrained from natural image dataset (such as large-scale annotated natural image database: ImageNet) to medical image tasks. In this chapter, we exploit three important factors of employing deep convolutional neural networks to computer-aided detection problems. First, we exploit and evaluate several different CNN architectures including from shallower to deeper CNNs: classical CifarNet, to recent AlexNet and state-of-the-art GoogLeNet and their variants. The studied models contain five thousand to 160 million parameters and vary in the numbers of layers. Second, we explore the influence of dataset scales and spatial image context configurations on medical image classification performance. Third, when and why transfer learning from the pretrained ImageNet CNN models (via fine-tuning) can be useful for medical imaging tasks are carefully examined. We study two specific computer-aided detection (CADe) problems, namely thoracoabdominal lymph node (LN) detection and interstitial lung disease (ILD) classification. We achieve the state-of-the-art performance on the mediastinal LN detection and report the first fivefold cross-validation classification results on predicting axial

H.-C. Shin (✉) · H.R. Roth · M. Gao · L. Lu · Z. Xu · I. Nogues ·

J. Yao · D. Mollura · R.M. Summers

Department of Radiology and Imaging Sciences, National Institutes of Health Clinical Center, Bethesda, MD 20837, USA

e-mail: hoochang.shin@nih.gov

L. Lu

e-mail: le.lu@nih.gov

R.M. Summers

e-mail: rms@nih.gov

CT slices with ILD categories. Our extensive quantitative evaluation, CNN model analysis, and empirical insights can be helpful to the design of high-performance CAD systems for other medical imaging tasks, without loss of generality.

8.1 Introduction

Large-scale annotated image datasets (i.e., ImageNet [1, 2]) coupled with the rekindled deep convolutional neural networks (CNN) [3, 4] have led to rapid progress in natural image recognition. From the perspective of data-driven learning, large-scale labeled datasets with representative data distribution characteristics are crucial to learning accurate or generalizable models [4, 5]. ImageNet [1] offers a very comprehensive database of more than 1.2 million categorized natural images of 1000+ classes. CNN models pretrained on ImageNet serve as the backbone for many object detection and image segmentation methods that are fine-tuned for other datasets [6, 7], such as PASCAL [8] and medical image categorization [9–12]. However, there exists no large-scale annotated medical image dataset comparable to ImageNet, because data acquisition is difficult and quality annotation may be very costly.

Currently, there are three major strategies to employ CNNs on medical image classification: (1) “training CNN from scratch” [13–17]; (2) using “off-the-shelf CNN” features (without retraining the CNN) complementary to existing handcrafted image features, on Chest X-ray [10] and CT lung nodule identification [9, 12]; and (3) performing unsupervised pretraining on natural or medical images and fine-tuning on target medical images via CNN or other types of deep learning models [18–21].

Previous studies have analyzed three-dimensional patch creation for LN detection [22, 23], atlas creation from chest CT [24], and the extraction of multi-level image features [25, 26]. Recently, decompositional 2.5D view resampling and aggregation of random view classification scores are used to acquire a sufficient number of training image samples for CNN. There are also several extensions from the decompositional view representation [27, 28], such as using a novel vessel-aligned multi-planar image representation for pulmonary embolism detection [29], fusing unregistered multiview for mammogram analysis [16], and classifying pulmonary perifissural nodules via an ensemble of 2D views [12].

Although natural and medical images differ significantly, image descriptors developed for object recognition in natural images, such as scale-invariant feature transform (SIFT) [30] and histogram of oriented gradients (HOG) [31], are widely used for object detection and segmentation in medical image analysis. ImageNet pretrained CNNs have been used for chest pathology identification and detection in X-ray and CT modalities [9, 10, 12]. Better performance results are reported when deep image features are integrated with low-level image features (e.g., GIST [32], bag-of-visual-words (BoVW) and bag-of-frequency [12]).

Here we exploit and discuss three important aspects of employing deep convolutional neural networks for computer-aided detection problems. Particularly, we explore and evaluate different CNN architectures varying in width (ranging from

5 thousand to 160 million parameters) and depth (various numbers of layers), describe the performance effects of varying dataset sizes and spatial image contexts, and discuss when and why transfer learning from pretrained ImageNet CNN models can be valuable. We further verify our hypothesis that inheriting and adapting rich hierarchical image features [5, 33] from ImageNet dataset for computer-aided diagnosis (CAD) is helpful. CNN architectures of the most studied seven-layered “AlexNet-CNN” [4], a shallower “Cifar-CNN” [27], and a much deeper version of “GoogLeNet-CNN” [33] (with our modifications on CNN structures) are studied. This work is partially motivated by recent studies [34, 35] in computer vision. The thorough quantitative analysis and evaluation on deep CNN [34] or sparsity image coding methods [35] elucidate the emerging techniques of the time and provide useful suggestions for their future stages of development, respectively.

Two specific computer-aided detection (CADe) problems, namely thoracoabdominal lymph node (LN) detection and interstitial lung disease (ILD) classification are explored. On the task of mediastinal LN detection, we surpass all currently reported results. We obtain 86% sensitivity on 3 false positives (FP) per patient, versus the prior state-of-art performance at sensitivities of 78% [36] (stacked shallow learning) and 70% [27] (CNN). ILD classification outcomes under the patient-level fivefold cross-validation protocol (CV5) are investigated and reported. The ILD dataset [37] contains 905 annotated image slices with 120 patients and 6 ILD labels. Such sparsely annotated datasets are generally difficult for CNN learning, due to the scarcity of labeled instances. Previous studies are all based on image patch classification [37–39].

Evaluation protocols and details are critical to deriving significant empirical findings [34]. Our experimental results suggest that different CNN architectures and dataset resampling protocols are critical for the LN detection tasks where the amount of labeled training data is sufficient and spatial contexts are local. Since LN images are more flexible than ILD images with respect to spatial resampling and reformatting, LN datasets can be extensively augmented by such image transformations. Thus LN datasets contain more training and testing data instances (due to data augmentation) than ILD datasets. Fine-tuning ImageNet-trained models for ILD classification is clearly advantageous and yields early promising results, when the amount of labeled training data is insufficient and multi-class categorization is used, as opposed to the LN dataset’s binary class categorization. Another important finding is that CNNs trained from scratch or fine-tuned from ImageNet models consistently outperform CNNs that merely use off-the-shelf CNN features, in both the LN and ILD classification tasks. We further analyze, via CNN activation visualizations, when and why transfer learning from non-medical to medical images in CADe problems can be valuable.

8.2 Datasets and Related Work

Convolutional neural networks are employed two CADe problems: thoracoabdominal lymph node (LN) detection and interstitial lung disease (ILD) detection. Until the detection aggregation approach [27, 40], thoracoabdominal lymph node (LN) detection via CADe mechanisms has yielded poor performance results. In [27], each 3D LN candidate produces up to 100 random 2.5D orthogonally sampled images or views which are then used to train an effective CNN model. The best performance on abdominal LN detection is achieved at 83% recall on 3FP per patient [27], using a “Cifar-10” CNN. Using the thoracoabdominal LN detection datasets [27], we aim to surpass this CADe performance level, by testing different CNN architectures, exploring various dataset re-sampling protocols, and applying transfer learning from ImageNet pretrained CNN models.

Interstitial lung disease (ILD) comprises about 150 diseases affecting the interstitium, which can severely impair the patient’s ability to breathe. Gao et al. [41] investigate the ILD classification problem in two scenarios: (1) slice-level classification: assigning a holistic two-dimensional axial CT slice image with its occurring ILD disease label(s); and (2) patch-level classification: (2.a) sampling patches within the 2D ROIs (Regions of Interest provided by [37]), then (2.b) classifying patches into seven category labels (six disease labels and one “healthy” label). Song et al. [38, 39] only address the second subtask of patch-level classification under the “leave-one-patient-out” (LOO) criterion. By training on the moderate-to-small scale ILD dataset [37], our main objective is to exploit and benchmark CNN based ILD classification performances under the CV5 metric (more realistic and unbiased than LOO [38, 39] and hard split [41]), with and without transfer learning.

Thoracoabdominal Lymph Node Datasets. We use the publicly available dataset from [27, 40]. There are 388 mediastinal LNs labeled by radiologists in 90 patient CT scans, and 595 abdominal LNs in 86 patient CT scans. To facilitate comparison, we adopt the data preparation protocol of [27], where positive and negative LN candidates are sampled with the fields-of-view (FOVs) of 30–45 mm, surrounding the annotated and detected LN centers (obtained by a candidate generation process). More precisely, [27, 36, 40] follow a coarse-to-fine CADe scheme, partially inspired by [42], which operates with $\sim 100\%$ detection recalls at the cost of approximately 40 false or negative LN candidates per patient scan. In this work, positive and negative LN candidate are first sampled up to 200 times with translations and rotations. Afterwards, negative LN samples are randomly re-selected at a lower rate close to the total number of positives. LN candidates are randomly extracted from fields-of-view (FOVs) spanning 35–128 mm in soft tissue window $[-100, 200\text{HU}]$. This allows us to capture multiple spatial scales of image context [43, 44]. The samples are then rescaled to a 64×64 pixel resolution via B-spline interpolation.

Unlike the heart or the liver, lymph nodes have no predetermined anatomic orientation. Hence, the purely random image resampling (with respect to scale, displacement and orientation) and reformatting (the axial, coronal, and sagittal views are in any system randomly resampled coordinates) is a natural choice, which also

Table 8.1 Average number of images in each fold for disease classes, when dividing the dataset in fivefold patient sets

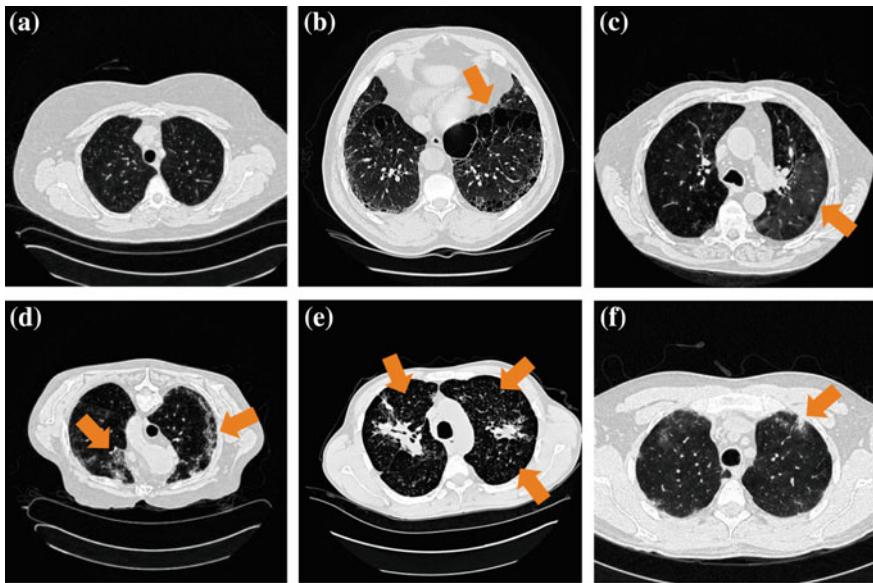
Normal	Emphysema	Ground glass	Fibrosis	Micronodules	Consolidation
30.2	20.2	85.4	96.8	63.2	39.2

happens to yield high CNN performance. Although we integrate three channels of information from three orthogonal views for LN detection, the pixel-wise spatial correlations between or among channels are not necessary. The convolutional kernels in the lower level CNN architectures can learn the optimal weights to linearly combine the observations from the axial, coronal, and sagittal channels by computing their dot products. Transforming axial, coronal, and sagittal representations to RGB also facilitates transfer learning from CNN models trained on ImageNet.

Interstitial Lung Disease Dataset. The publicly available dataset [37] is studied. It contains 905 image slices from 120 patients, with six lung tissue types of annotations: healthy (NM), emphysema (EM), ground glass (GG), fibrosis (FB), micronodules (MN), and consolidation (CD) (Fig. 8.2). At the slice level, the objective is to classify the status of “presence/absence” of any of the six ILD classes for an input axial CT slice [41]. Characterizing an arbitrary CT slice against any possible ILD type, without any manual ROI (in contrast to [38, 39]), can be useful for large-scale patient screening. For slice-level ILD classification, we sampled the slices 12 times with random translations and rotations. After this, we balanced the numbers of CT slice samples for the six classes by randomly sampling several instances at various rates. For patch-based classification, we sampled up to 100 patches of size 64×64 from each ROI. This dataset is divided into five folds with disjoint patient subsets. The average number of CT slices (training instances) per fold is small, as shown in Table 8.1.

In this ILD dataset [37], very few CT slices are labeled as normal or healthy. The remaining CT slices cannot be simply classified as normal, because many ILD disease regions or slices have not yet been labeled. Thus ILD [37] is a partially labeled database as one of its main limitations. Research is being conducted to address this issue. For example, [45] proposes to fully label the ILD dataset pixelwise via segmentation label propagation.

To leverage the CNN architectures designed for color images and to transfer CNN parameters pretrained on ImageNet, we transform all grayscale axial CT slice images via three CT window ranges: lung window range $[-1400, -200\text{HU}]$, high-attenuation range $[-160, 240\text{HU}]$, and low-attenuation range $[-1400, -950\text{HU}]$. We then encode the transformed images into RGB channels (to be aligned with the input channels of CNN models [4, 33] pretrained from natural image datasets [1]). The low-attenuation CT window is useful for visualizing certain texture patterns of lung diseases (especially emphysema). The usage of different CT attenuation channels improves classification results over the usage of a single CT windowing channel, as demonstrated in [41]. More importantly, these CT windowing processes



(a): healthy; (b): emphysema; (c): ground glass; (d): fibrosis; (e): micronodules; (f): consolidation

Fig. 8.1 Some examples of CT image slices with six lung tissue types in the ILD dataset [37]. Disease tissue types are located with *dark orange arrows*

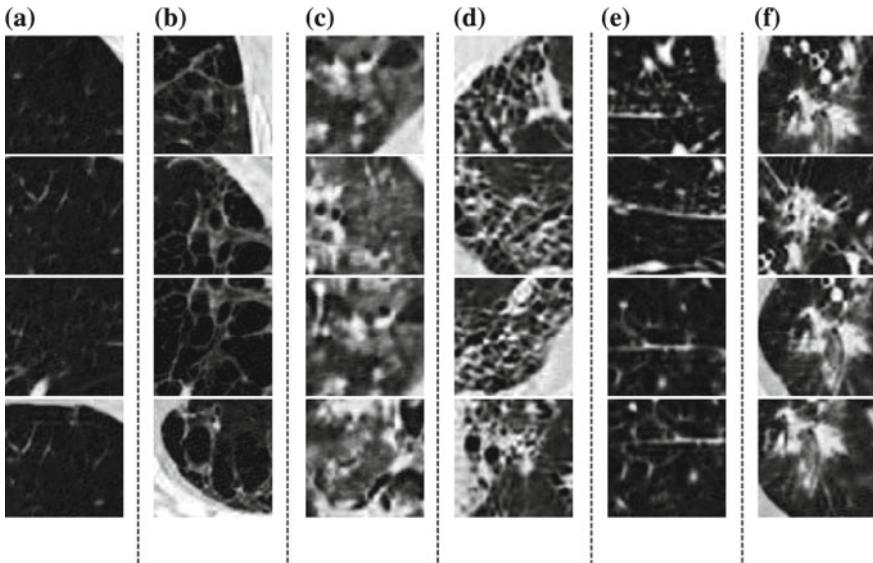


Fig. 8.2 Some examples of 64 × 64 pixel CT image patches for **a** NM, **b** EM, **c** GG, **d** FB, **e** MN
f CD

do not depend on the lung segmentation, which instead is directly defined in the CT HU space.

We empirically compare performance in two scenarios: with or without a rough lung segmentation.¹ There is no significant difference between two setups. For CNN-based image recognition, highly accurate lung segmentation seems not very necessary. The localization of ILD regions within the lung is simultaneously achieved through selectively weighted CNN reception fields in the final convolutional layers during CNN training [47, 48]. Areas outside of the lung appear in both healthy or diseased images and CNN training learns to ignore them by setting very small filter weights around the corresponding regions (Fig. 8.8).

8.3 Methods

In this study, we explore, evaluate, and analyze the influence of various CNN Architectures, dataset characteristics (when we need more training data or better models for object detection [49]) and CNN transfer learning from nonmedical to medical image domains. These three key elements of building effective deep CNN models for CADe problems are described below.

8.3.1 *Convolutional Neural Network Architectures*

We mainly explore three convolutional neural network architectures (CifarNet [5, 27], AlexNet [4] and GoogLeNet [33]) with different model training parameter values. The current deep learning models [27, 50, 51] in medical image tasks are at least $2 \sim 5$ orders of magnitude smaller than even AlexNet [4]. More complex CNN models [27, 50] have only about 150 or 15 K parameters. Roth et al. [27] adopt the CNN architecture tailored to the Cifar-10 dataset [5] and operate on image windows of $32 \times 32 \times 3$ pixels for lymph node detection, while the simplest CNN in [52] has only one convolutional, pooling, and FC layer, respectively.

We use CifarNet [5] as used in [27] as a baseline for the LN detection. AlexNet [4] and GoogLeNet [33] are also modified to evaluate these state-of-the-art CNN architecture from ImageNet classification task [2] to our CADe problems and datasets. A simplified illustration of three CNN architectures exploited is shown in Fig. 8.3. CifarNet always takes $32 \times 32 \times 3$ image patches as input while AlexNet and GoogLeNet are originally designed for the fixed image dimension of $256 \times 256 \times 3$ pixels. We also reduced the filter size, stride, and pooling parameters of AlexNet and GoogLeNet to accommodate a smaller input size of $64 \times 64 \times 3$ pixels. We

¹This can be achieved by segmenting the lung using simple label fusion methods [46]. First, we overlay the target image slice with the average lung mask among the training folds. Second, we perform simple morphology operations to obtain the lung boundary.

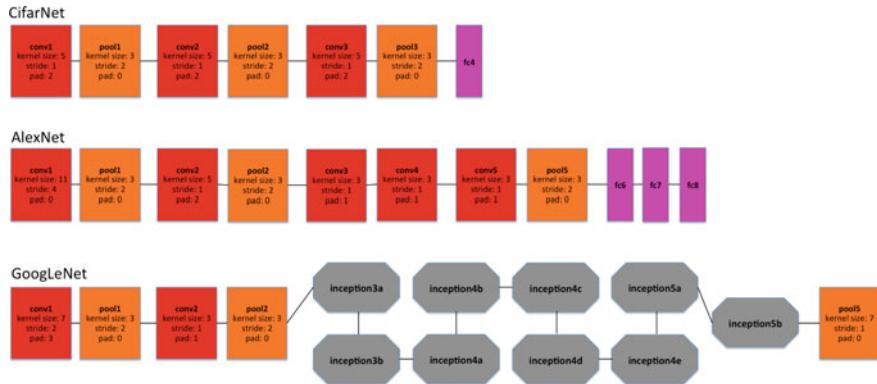


Fig. 8.3 A simplified illustration of the CNN architectures is used. GoogLeNet [33] contains two convolution layers, three pooling layers, and nine inception layers. Each of the inception layer of GoogLeNet consists of six convolution layers and one pooling layer

do so to produce and evaluate “simplified” AlexNet and GoogLeNet versions that are better suited to the smaller scale training datasets common in CADe problems. Throughout the paper, we refer to the models as CifarNet (32×32) or CifarNet (dropping 32×32); AlexNet (256×256) or AlexNet-H (high resolution); AlexNet (64×64) or AlexNet-L (low resolution); GoogLeNet (256×256) or GoogLeNet-H and GoogLeNet (64×64) or GoogLeNet-L (dropping 3 since all image inputs are three channels).

CifarNet

CifarNet, introduced in [5], was the state-of-the-art model for object recognition on the Cifar10 dataset, which consists of 32×32 images of 10 object classes. The objects are normally centered in the images. CifarNet has three convolution layers, three pooling layers, and one fully connected layer. This CNN architecture, also used in [27] has about 0.15 million free parameters. We adopt it as a baseline model for the LN detection.

AlexNet

The AlexNet architecture was published in [4], achieved significantly improved performance over the other non-deep learning methods for ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2012. This success has revived the interest in CNNs [3] in computer vision. ImageNet consists of 1.2 million 256×256 images belonging to 1000 categories. At times, the objects in the image are small and obscure, and thus pose more challenges for learning a successful classification model. More details about the ImageNet dataset will be discussed in Sect. 8.3.2. AlexNet has five convolution layers, three pooling layers, and two fully connected layers with approximately 60 million free parameters. AlexNet is our default CNN architecture for evaluation and analysis in the remainder of the paper.

GoogLeNet

The GoogLeNet model proposed in [33], is significantly more complex and deep than all previous CNN architectures. More importantly, it also introduces a new module called “Inception”, concatenating filters of different sizes and dimensions into a single module. Overall, GoogLeNet has two convolution layers, two pooling layers, and nine “Inception” layers. Each “Inception” layer consists of six convolution layers and one pooling layer. GoogLeNet is the current state-of-the-art CNN architecture for the ILSVRC challenge, where it achieved 5.5% top-5 classification error on the ImageNet challenge, compared to AlexNet’s 15.3% top-5 classification error.

8.3.2 *ImageNet: Large-Scale Annotated Natural Image Dataset*

ImageNet [1] has more than 1.2 million 256×256 images categorized under 1000 object class categories. There are more than 1000 training images per class. The database is organized according to the WordNet [53] hierarchy, which currently contains only nouns in 1000 object categories. The image object labels are obtained largely through crowdsourcing, e.g., Amazon Mechanical Turk, and human inspection. Some examples of object categories in ImageNet are “sea snake”, “sandwich”, “vase”, “leopard”, etc. ImageNet is currently the largest image dataset among other standard datasets for visual recognition. Indeed, the Caltech101, Caltech256 and Cifar10 dataset merely contain 60000 32×32 images and 10 object classes. Furthermore, due to the large number (1000+) of object classes, the objects belonging to each ImageNet class category can be occluded, partial and small, relative to those in the previous public image datasets. This significant intraclass variation poses greater challenges to any data-driven learning system that builds a classifier to fit given data and generalize to unseen data. ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) has become the standard benchmark for large-scale object recognition.

8.3.3 *Training Protocols and Transfer Learning*

When **learned from scratch**, all the parameters of CNN models are initialized with random Gaussian distributions and trained for 30 epochs with the mini-batch size of 50 image instances. Training convergence can be observed within 30 epochs. The other hyperparameters are momentum: 0.9; weight decay: 0.0005; (base) learning rate: 0.01, decreased by a factor of 10 at every 10 epochs. We use the Caffe framework [54] and NVidia K40 GPUs to train the CNNs.

AlexNet and GoogLeNet CNN models can be either learned from scratch or **fine-tuned from pretrained models**. Girshick et al. [6] find that, by applying ImageNet pretrained AlexNet to PASCAL dataset [8], performances of semantic

20-class object detection and segmentation tasks significantly improve over previous methods that use no deep CNNs. AlexNet can be fine-tuned on the PASCAL dataset to surpass the performance of the ImageNet pretrained AlexNet, although the difference is not as significant as that between the CNN and non-CNN methods. Similarly, [55, 56] also demonstrate that better performing deep models are learned via CNN transfer learning from ImageNet to other datasets of limited scales.

Our hypothesis on CNN parameter transfer learning is the following: despite the disparity between natural images and medical images, CNNs comprehensively trained on the large scale well-annotated ImageNet may still be transferred to make medical image recognition tasks more effective. Collecting and annotating large numbers of medical images still poses significant challenges. On the other hand, the mainstream deep CNN architectures (e.g., AlexNet and GoogLeNet) contain tens of millions of free parameters to train, and thus require sufficiently large numbers of labeled medical images.

For transfer learning, we follow the approach of [6, 55] where all CNN layers except the last are fine-tuned at a learning rate 10 times smaller than the default learning rate. The last fully connected layer is random initialized and freshly trained, in order to accommodate the new object categories in our CADe applications. Its learning rate is kept at the original 0.01. We denote the models with random initialization or transfer learning as AlexNet-RI and AlexNet-TL, and GoogLeNet-RI and GoogLeNet-TL. We found that the transfer learning strategy yields the best performance results. Determining the optimal learning rate for different layers is challenging, especially for very deep networks such as GoogLeNet.

We also perform experiments using “**off-the-shelf**” CNN features of AlexNet pretrained on ImageNet and training only the final classifier layer to complete the new CADe classification tasks. Parameters in the convolutional and fully connected layers are fixed and are used as deep image extractors, as in [9, 10, 12]. We refer to this model as AlexNet-ImNet in the remainder of the paper. Note that [9, 10, 12] train support vector machines and random forest classifiers using ImageNet pretrained CNN features. Our simplified implementation is intended to determine whether fine-tuning the “end-to-end” CNN network is necessary to improve performance, as opposed to merely training the final classification layer. This is a slight modification from the method described in [9, 10, 12].

Finally, transfer learning in CNN representation, as empirically verified in previous literature [11, 57–60], can be effective in various cross-modality imaging settings (RGB images to depth images [57, 58], natural images to general CT and MRI images [11], and natural images to neuroimaging [59] or ultrasound [60] data). More thorough theoretical studies on cross-modality imaging statistics and transferability will be needed for future studies.

8.4 Experiments and Discussions

In this section, we evaluate and compare the performances of nine CNN model configurations (CifarNet, AlexNet-ImNet, AlexNet-RI-H, AlexNet-TL-H, AlexNet-RI-L, GoogLeNet-RI-H, GoogLeNet-TL-H, GoogLeNet-RI-L and combined) on two publicly available datasets [27, 37, 40].

8.4.1 Thoracoabdominal Lymph Node Detection

We train and evaluate CNNs using threefold cross-validation (folds are split into disjoint sets of patients), with different CNN architectures. In testing, each LN candidate has multiple random 2.5D views tested by CNN classifiers to generate LN class probability scores. We follow the random view aggregation by averaging probabilities [27]. We first sample the LN image patches at a 64×64 pixel resolution. We then upsample the 64×64 pixel LN images via bilinear interpolation to 256×256 pixels, in order to accommodate AlexNet-RI-L, AlexNet-TL-H, GoogLeNet-RI-H, and GoogLeNet-TL-H. For the modified AlexNet-RI-L at (64×64) pixel resolution, we reduce the number of first layer convolution filters from 96 to 64 and reduce the stride from 4 to 2. For the modified GoogLeNet-RI (64×64) , we decrease the number of first layer convolution filters from 64 to 32, the pad size from 3 to 2, the kernel size from 7 to 5, stride from 2 to 1 and the stride of the subsequent pooling layer from 2 to 1. We slightly reduce the number of convolutional filters in order to accommodate the smaller input image sizes of target medical image datasets [27, 37], while preventing overfitting. This eventually improves performance on patch-based classification. CifarNet is used in [27] to detect LN samples of $32 \times 32 \times 3$ images. For consistency purposes, we downsample $64 \times 64 \times 3$ resolution LN sample images to the dimension of $32 \times 32 \times 3$.

Results for lymph node detection in the mediastinum and abdomen are reported in Table 8.2. FROC curves are illustrated in Fig. 8.4. The area-under-the-FROC-curve (AUC) and true positive rate (TPR, recall or sensitivity) at three false positives per patient (TPR/3FP) are used as performance metrics. Of the nine investigated CNN models, CifarNet, AlexNet-ImNet, and GoogLeNet-RI-H generally yielded the least competitive detection accuracy results. Our LN datasets are significantly more complex (i.e., display much larger within-class appearance variations), especially due to the extracted fields-of-view (FOVs) of (35–128 mm) compared to (30–45 mm) in [27], where CifarNet is also employed. In this experiment, CifarNet is under-trained with respect to our enhanced LN datasets, due to its limited input resolution and parameter complexity. The inferior performance of AlexNet-ImNet implies that using the pretrained ImageNet CNNs alone as “off-the-shelf” deep image feature extractors may not be optimal or adequate for mediastinal and abdominal LN detection tasks. To complement “off-the-shelf” CNN features, [9, 10, 12] all add various handcrafted image features as hybrid inputs for the final classification.

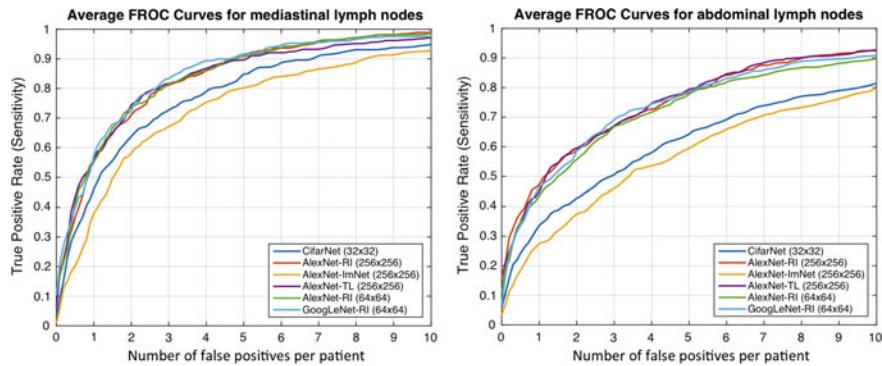


Fig. 8.4 FROC curves averaged on threefold CV for the abdominal (*left*) and mediastinal (*right*) lymph nodes using different CNN models

Table 8.2 Comparison of mediastinal and abdominal LN detection results using various CNN models. Numbers in bold indicate the best performance values on classification accuracy

Region	Mediastinum		Abdomen	
Method	AUC	TPR/3FP	AUC	TPR/3FP
[40]	–	0.63	–	0.70
[27]	0.92	0.70	0.94	0.83
[36]	–	0.78	–	0.78
CifarNet	0.91	0.70	0.81	0.44
AlexNet-ImNet	0.89	0.63	0.80	0.41
AlexNet-RI-H	0.94	0.79	0.92	0.67
AlexNet-TL-H	0.94	0.81	0.92	0.69
GoogLeNet-RI-H	0.85	0.61	0.80	0.48
GoogLeNet-TL-H	0.94	0.81	0.92	0.70
AlexNet-RI-L	0.94	0.77	0.88	0.61
GoogLeNet-RI-L	0.95	0.85	0.91	0.69
Combined	0.95	0.85	0.93	0.70

GoogLeNet-RI-H performs poorly, as it is susceptible to overfitting. No sufficient data samples are available to train GoogLeNet-RI-H with random initialization. Indeed, due to GoogLeNet-RI-H's complexity and 22-layer depth, million image datasets may be required to properly train this model. However, GoogLeNet-TL-H significantly improves upon GoogLeNet-RI-H (0.81 versus 0.61 TPR/3FP in mediastinum; 0.70 versus 0.48 TPR/3FP in abdomen). This indicates that transfer learning offers a much better initialization of CNN parameters than random initialization. Likewise, AlexNet-TL-H consistently outperforms AlexNet-RI-H, though by smaller margins (0.81 versus 0.79 TPR/3FP in mediastinum; 0.69 versus 0.67 TPR/3FP in abdomen). This is also consistent with the findings reported for ILD detection in

Table 8.3. GoogLeNet-TL-H yields results similar to AlexNet-TL-H’s for the mediastinal LN detection, and slightly outperforms Alex-Net-H for abdominal LN detection. AlexNet-RI-H exhibits less severe overfitting than GoogLeNet-RI-H. We also evaluate a simple ensemble by averaging the probability scores from five CNNs: AlexNet-RI-H, AlexNet-TL-H, AlexNet-RI-H, GoogLeNet-TL-H, and GoogLeNet-RI-L. This combined ensemble outputs the classification accuracies matching or slightly exceeding the best performing individual CNN models on the mediastinal or abdominal LN detection tasks, respectively.

Many CNN models achieve notably better (FROC-AUC and TPR/3FP) results than the previous state-of-the-art methods [36] for **mediastinal** LN detection: GoogLeNet-RI-L obtains an AUC = 0.95 and 0.85 TPR/3FP, versus AUC = 0.92 and 0.70 TPR/3FP [27] and 0.78 TPR/3FP [36] which uses stacked shallow learning. This difference lies in the fact that annotated lymph node segmentation masks are required to learn a mid-level semantic boundary detector [36], whereas CNN approaches only need LN locations for training [27]. In **abdominal** LN detection, [27] obtains the best trade-off between its CNN model complexity and sampled data configuration. Our best performing CNN model is GoogLeNet-TL (256×256) which obtains an AUC=0.92 and 0.70 TPR/3FP.

The main difference between our dataset preparation protocol and that from [27] is a more aggressive extraction of random views within a much larger range of FOVs. The usage of larger FOVs to capture more image spatial context is inspired by deep zoom-out features [44] that improve semantic segmentation. This image sampling scheme contributes to our best reported performance results in both mediastinal LN detection (in this paper) and automated pancreas segmentation [61]. Comparing to the scenario that abdominal LNs are surrounded by many other similar looking objects, mediastinal LNs are more easily distinguishable, due to the images’ larger spatial contexts. Finally, from the perspective of the data model trade-off: “*Do We Need More Training Data or Better Models?*” [49], more abdomen CT scans from distinct patient populations need to be acquired and annotated, in order to take full advantage of deep CNN models of high capacity. Nevertheless, deeper and wider CNN models (e.g., GoogLeNet-RI-L and GoogLeNet-TL-H versus Cifar-10 [27]) have shown improved results in the mediastinal LN detection.

Figure 8.5 provides examples of misclassified lymph nodes (in axial view) (both false negatives (**Left**) and false positives(**Right**)), from the Abdomen and Mediastinum datasets. The overall reported LN detection results are clinically significant, as indicated in [62].

8.4.2 Interstitial Lung Disease Classification

The CNN models evaluated in this experiment are (1) AlexNet-RI (training from scratch on the ILD dataset with random initialization); (2) AlexNet-TL (with transfer learning from [4]); (3) AlexNet-ImNet: pretrained ImageNet-CNN model [4] with only the last cost function layer retrained from random initialization, according to

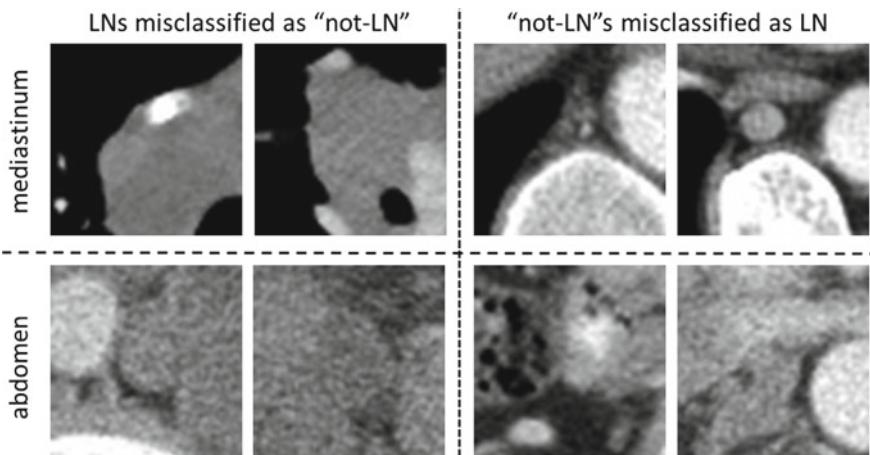


Fig. 8.5 Examples of misclassified lymph nodes (in axial view) of both false negatives (*Left*) and false positives (*Right*). Mediastinal LN examples are shown in the *upper row*, and abdominal LN examples in the *bottom row*

Table 8.3 Comparison of interstitial lung disease classification accuracies on both slice-level (Slice-CV5) and patch-based (Patch-CV5) classification using fivefold CV. Bold numbers indicate the best performance values on classification accuracy

Method	AlexNet- ImNet	AlexNet-RI	AlexNet-TL	GoogLeNet-RI	GoogLeNet-TL	Avg-All
Slice-CV5	0.45	0.44	0.46	0.41	0.57	0.53
Patch-CV5	0.76	0.74	0.76	0.75	0.76	0.79

the six ILD classes (similar to [9] but without using additional handcrafted non-deep feature descriptors, such as GIST and BoVW); (4) GoogLeNet-RI (random initialization); (5) GoogLeNet-TL (GoogLeNet with transfer learning from [33]). All ILD images (patches of 64×64 and CT axial slices of 512×512) are resampled to a fixed dimension of 256×256 pixels.

We evaluate the ILD classification task with fivefold CV on patient-level split, as it is more informative for real clinical performance than LOO. The classification accuracy rates for interstitial lung disease detection are shown in Table 8.3. Two subtasks on ILD patch and slice classifications are conducted. In general, patch-level ILD classification is less challenging than slice-level classification, as far more data samples can be sampled from the manually annotated ROIs (up to 100 image patches per ROI), available from [37]. From Table 8.3, all five deep models evaluated obtain comparable results within the range of classification accuracy rates [0.74, 0.76]. Their averaged model achieves a slightly better accuracy of 0.79.

F1-scores [38, 39, 52] and the confusion matrix (Table 8.5) for patch-level ILD classification using GoogLeNet-TL under fivefold cross-validation (we denote as

Table 8.4 Comparison of interstitial lung disease classification results using F-scores: NM, EM, GG, FB, MN and CD

	NM	EM	GG	FB	MN	CD
Patch-LOO [38]	0.84	0.75	0.78	0.84	0.86	–
Patch-LOO [39]	0.88	0.77	0.80	0.87	0.89	–
Patch-CV10 [52]	0.84	0.55	0.72	0.76	0.91	–
Patch-CV5	0.64	0.81	0.74	0.78	0.82	0.64
Slice-Test [41]	0.40	1.00	0.75	0.80	0.56	0.50
Slice-CV5	0.22	0.35	0.56	0.75	0.71	0.16
Slice-Random	0.90	0.86	0.85	0.94	0.98	0.83

Table 8.5 Confusion matrix for ILD classification (patch-level) with fivefold CV using GoogLeNet-TL

Ground truth	Prediction					
	NM	EM	GG	FB	MN	CD
NM	0.68	0.18	0.10	0.01	0.03	0.01
EM	0.03	0.91	0.00	0.02	0.03	0.01
GG	0.06	0.01	0.70	0.09	0.06	0.08
FB	0.01	0.02	0.05	0.83	0.05	0.05
MN	0.09	0.00	0.07	0.04	0.79	0.00
CD	0.02	0.01	0.10	0.18	0.01	0.68

Patch-CV5) are also computed. F1-scores are reported on patch classification only (32×32 pixel patches extracted from manual ROIs) [38, 39, 52], as shown in Table 8.4. Both [38, 39] use the evaluation protocol of “leave-one-patient-out” (LOO), which is arguably much easier and not directly comparable to tenfold CV [52] or our Patch-CV5. In this study, we classify six ILD classes by adding a consolidation (CD) class to five classes of healthy (normal - NM), emphysema (EM), ground glass (GG), fibrosis (FB), and micronodules (MN) in [38, 39, 52]. Patch-CV10 [52] and Patch-CV5 report similar medium to high F-scores. This implies that the ILD dataset (although one of the mainstream public medical image datasets) may not adequately represent ILD disease CT lung imaging patterns, over a population of only 120 patients. Patch-CV5 yields higher F-scores than [52] and classifies the extra consolidation (CD) class. At present, the most pressing task is to drastically expand the dataset or to explore across-dataset deep learning on the combined ILD and LTRC datasets [63].

Recently, Gao et al. [41] have argued that a new CADe protocol on holistic classification of ILD diseases directly, using axial CT slice attenuation patterns and CNN, may be more realistic for clinical applications. We refer to this as slice-level classification, as image patch sampling from manual ROIs can be completely avoided (hence, no manual ROI inputs will be provided). The experimental results in [41]

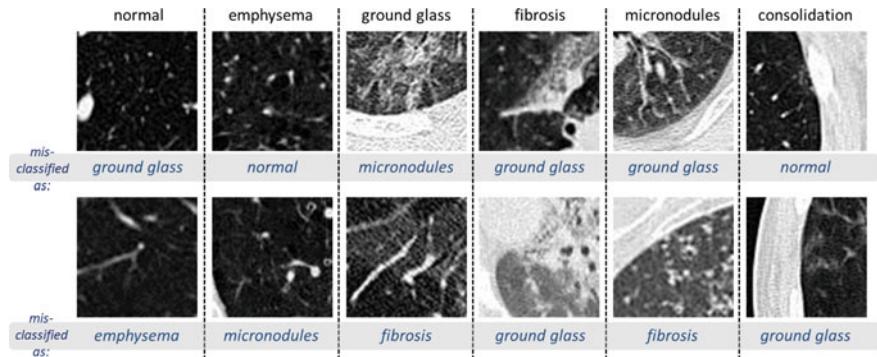


Fig. 8.6 Visual examples of misclassified ILD 64×64 patches (in axial view), with their ground truth labels and inaccurately classified labels

are conducted with a patient-level hard split of 100 (training) and 20 (testing). The method's testing F-scores (i.e., Slice-Test) are given in Table 8.4. Note that the F-scores in [41] are not directly comparable to our results, due to different evaluation criteria. Only Slice-Test is evaluated and reported in [41], and we find that F-scores can change drastically from different rounds of the fivefold CV.

While it is a more practical CADe scheme, slice-level CNN learning [41] is very challenging, as it is restricted to only 905 CT image slices with tagged ILD labels. We only benchmark the slice-level ILD classification results in this section. Even with the help of data augmentation (described in Sect. 8.2), the classification accuracy of GoogLeNet-TL from Table 8.3 is only 0.57. However, transfer learning from ImageNet pretrained model is consistently beneficial, as evidenced by AlexNet-TL (0.46) versus AlexNet-RI (0.44), and GoogLeNet-TL (0.57) versus GoogLeNet-RI (0.41). It especially prevents GoogLeNet from overfitting on the limited CADe datasets. Finally, when the cross-validation is conducted by randomly splitting the set of all 905 CT axial slices into five folds, markedly higher F-scores are obtained (Slice-Random in Table 8.4). This further validates the claim that the dataset poorly generalizes ILDs for different patients. Figure 8.6 shows examples of misclassified ILD patches (in axial view), with their ground truth labels and inaccurately classified labels.

For ILD classification, the most critical performance bottlenecks are the challenge of cross-dataset learning and the limited patient population size. We attempt to overcome these obstacles by merging the ILD [37] and LTRC datasets. Although the ILD [37] and LTRC datasets [63] (used in [19]) were generated and annotated separately, they contain many common disease labels. For instance, the ILD disease classes emphysema (EM), ground glass (GG), fibrosis (FB), and micronodules (MN) belong to both datasets, and thus can be jointly trained and tested to form a larger and unified dataset. In this work, we sample image patches from the slice using the ROIs for the ILD provided in the dataset, in order to be consistent with previous methods in patch-level [38, 39, 52] and slice-level classification [41].

8.4.3 Evaluation of Five CNN Models Using ILD Classification

In this work, we mainly focus on AlexNet and GoogLeNet. AlexNet is the first notably successful CNN architecture on the ImageNet challenge and has rekindled significant research interests on CNN. GoogLeNet is the state-of-the-art deep model, which has outperformed other notable models, such as AlexNet, OverFeat, and VGGNet [64, 65] in various computer vision benchmarks. Likewise, a reasonable assumption is that OverFeat and VGGNet may generate quantitative performance results ranked between AlexNet's and GoogLeNet's. For completeness, we include the Overfeat and VGGNet in the following evaluations, to bolster our hypothesis.

Overfeat

OverFeat is described in [64] as an integrated framework for using CNN for classification, localization and detection. Its architecture is similar to that of AlexNet, but contains far more parameters (e.g., 1024 convolution filters in both “conv4” and “conv5” layers compared to 384 and 256 convolution kernels in the “conv4” and “conv5” layers of AlexNet), and operates more densely (e.g., smaller kernel size of 2 in “pool2” layer “pool5” compared to the kernel size 3 in “pool2” and “pool5” of AlexNet) on the input image. Overfeat is the winning model of the ILSVRC 2013 in detection and classification tasks.

VGGNet

The VGGNet architecture in [65] is designed to significantly increase the depth of the existing CNN architectures with 16 or 19 layers. Very small 3×3 size convolutional filters are used in all convolution layers with a convolutional stride of size 1, in order to reduce the number of parameters in deeper networks. Since VGGNet is substantially deeper than the other CNN models, VGGNet is more susceptible to the vanishing gradient problem [66–68]. Hence, the network may be more difficult to train. Training the network requires far more memory and computation time than AlexNet. The 16 layer variant is used here.

The classification accuracies for ILD slice and patch level classification of five CNN architectures (CifarNet, AlexNet, Overfeat, VGGNet and GoogLeNet) are

Table 8.6 Classification results on ILD classification with LOO

Method	ILD-Slice	Method	ILD-Patch
CifarNet	—	CifarNet	0.799
AlexNet-TL	0.867	AlexNet-TL	0.865
Overfeat-TL	0.877	Overfeat-TL	0.879
VGG-16-TL	0.90	VGG-16-TL	0.893
GoogLeNet-TL	0.902	GoogLeNet-TL	0.911

Table 8.7 Training time and memory requirements of the five CNN architectures on ILD patch-based classification up to 90 epochs

	CifarNet	AlexNet	Overfeat	VGG-16	GoogLeNet
Time	7 m 16 s	1 h 2 m	1 h 26 m	20 h 24 m	2 h 49 m
Memory	2.25 GB	3.45 GB	4.22 GB	9.26 GB	5.37 GB

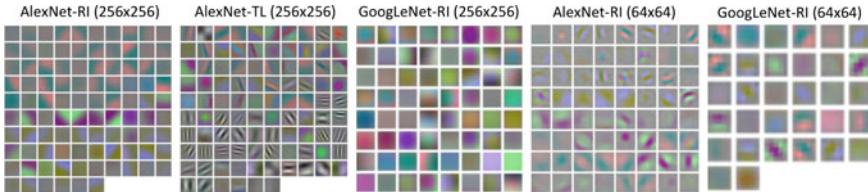


Fig. 8.7 Visualization of first layer convolution filters of CNNs trained on abdominal and mediastinal LNs in RGB color, from random initialization (AlexNet-RI (256×256), AlexNet-RI (64×64), GoogLeNet-RI (256×256) and GoogLeNet-RI (64×64)) and with transfer learning (AlexNet-TL (256×256))

shown in Table 8.6. Based on the analysis in Sect. 8.4.2, transfer learning is only used for the slice level classification task. From Table 8.6, quantitative classification accuracy rates increase as CNN models become more complex (CifarNet, AlexNet, Overfeat, VGGNet and GoogLeNet, in ascending order), for both ILD slice and patch level classification. These results validate our assumption that OverFeat's and VGGNets performance levels fall between AlexNet's and GoogLeNets. CifarNet is designed for images with smaller dimensions (32×32 images), and thus is not catered to classification tasks involving 256×256 images.

CNN training is implemented with the Caffe [54] deep learning framework, using a NVidia K40 GPU on Ubuntu 14.04 Linux OS. All models are trained for up to 90 epochs with early stopping criteria, where a model snapshot with low validation loss is taken for the final model. Other hyperparameters are fixed as follows: momentum: 0.9; weight decay: 0.0005; and a step learning rate schedule with base learning rate of 0.01, decreased by a factor of 10 every 30 epochs. The image batch size is set to 128, except for GoogLeNet's (64) and VGG-16's (32), which are the maximum batch sizes that can fit in the NVidia K40 GPU with 12GB of memory capacity. Table 8.7 illustrates the training time and memory requirements of the five CNN architectures on ILD patch-based classification up to 90 epochs.

8.4.4 Analysis via CNN Learning Visualization

In this section, we determine and analyze, via CNN visualization, the reasons for which transfer learning is beneficial to achieve better performance on CAD applications.

Thoracoabdominal LN Detection. In Fig. 8.7, the first layer convolution filters from five different CNN architectures are visualized. We notice that without transfer learning [6, 55], somewhat blurry filters are learned (AlexNet-RI (256×256), AlexNet-RI (64×64), GoogLeNet-RI (256×256) and GoogLeNet-RI (64×64)). However, in AlexNet-TL (256×256), many higher orders of contrast- or edge-preserving patterns (that enable capturing image appearance details) are evidently learned through fine-tuning from ImageNet. With a smaller input resolution, AlexNet-RI (64×64) and GoogLeNet-RI (64×64) can learn image contrast filters to some degree; whereas, GoogLeNet-RI (256×256) and AlexNet-RI (256×256) have oversmooth low-level filters throughout.

ILD Classification. We analyze visual CNN activations from the ILD dataset since the slice-level setting is most similar to ImageNet's. Both datasets use full-size images. The last pooling layer (pool-5) activation maps of the ImageNet pretrained AlexNet [4] (analogical to AlexNet-ImNet) and AlexNet-TL, obtained by processing two input images of Fig. 8.1b, c, are shown in Fig. 8.8a, b. The last pooling layer activation map summarizes the entire input image by highlighting which relative locations or neural reception fields relative to the image are activated. There are a total of 256 (6×6) reception fields in AlexNet [4]. Pooling units where the relative image location of the disease region is present in the image are highlighted with green boxes. Next, we reconstruct the original ILD images using the process of deconvolution, backpropagating with convolution and un-pooling from the activation maps of the chosen pooling units [69]. From the reconstructed images (Fig. 8.8 bottom), we observe that with fine-tuning, AlexNet-TL detects and localizes objects of interest (ILD disease regions depicted in Fig. 8.1b, c) better than AlexNet-ImNet.

8.4.5 Findings and Observations

Through the experimental study so far in this chapter, our empirical findings are summarized below. These observations may be informative for the design of high-performance image recognition CADe systems.

1. Deep CNN architectures with 8, even 22 layers [4, 33], can be useful even for CADe problems where the available training datasets are limited. Previously, CNN models used in medical image analysis applications have often been $2 \sim 5$ orders of magnitude smaller.
2. The trade-off between using better learning models and using more training data [49] should be carefully considered when searching for an optimal solution to any CADe problem (e.g., mediastinal and abdominal LN detection).
3. Limited datasets can be a bottleneck to further advancement of CADe. Building progressively growing (in scale), well annotated datasets is at least as crucial as developing new algorithms. This has been accomplished, for instance, in the field of computer vision. The well-known scene recognition problem has made

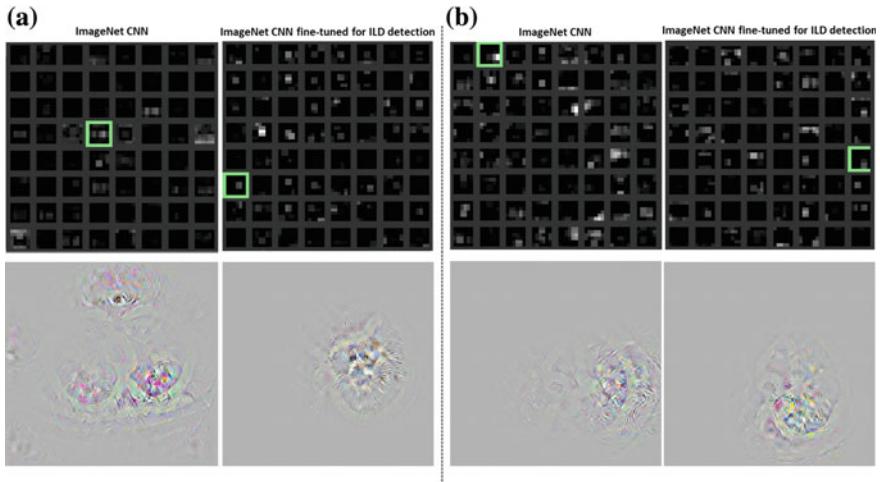


Fig. 8.8 Visualization of the last pooling layer (pool-5) activations (*top*). Pooling units where the relative image location of the disease region is located in the image are highlighted with *green boxes*. The original images reconstructed from the units are shown in the *bottom* [69]. The examples in **a** and **b** are computed from the input ILD images in Fig. 8.1b, c, respectively

tremendous progress, thanks to the steady and continuous development of Scene-15, MIT Indoor-67, SUN-397 and Place datasets [56].

4. Transfer learning from the large-scale annotated natural image datasets (ImageNet) to CADe problems has been consistently beneficial in our experiments. This sheds some light on cross-dataset CNN learning in the medical image domain, e.g., the union of the ILD [37] and LTRC datasets [63], as suggested in this paper.
5. Using the off-the-shelf deep CNN image features to CADe problems can be improved by either exploring the performance complementary properties of hand-crafted features [9, 10, 12], or by training CNNs from scratch and better fine-tuning CNNs on the target medical image dataset, as evaluated in this paper.

8.5 Conclusion

In this paper, we study and exploit three important aspects on deep convolutional neural networks architectures, dataset characteristics, and transfer learning for CADe problems. We evaluate CNN performance on two different computer-aided diagnosis applications: thoracoabdominal lymph node detection and interstitial lung disease classification. The empirical evaluation, CNN model visualization, CNN performance analysis and empirical insights can be generalized to the design of high-performance CAD systems for other medical imaging tasks.

References

1. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: IEEE CVPR
2. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg A, Fei-Fei L (2014) Imagenet large scale visual recognition challenge. [arXiv:1409.0575](https://arxiv.org/abs/1409.0575)
3. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
4. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: NIPS, pp 1097–1105
5. Krizhevsky A (2009) Learning multiple layers of features from tiny images, in Master's Thesis. University of Toronto, Department of Computer Science
6. Girshick R, Donahue J, Darrell T, Malik J (2015) Region-based convolutional networks for accurate object detection and semantic segmentation. In: IEEE Transaction Pattern Analysis Machine Intelligence
7. He K, Zhang X, Ren S, Sun J (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transaction Pattern Analysis Machine Intelligence
8. Everingham M, Eslami SMA, Van Gool L, Williams C, Winn J, Zisserman A (2015) The pascal visual object classes challenge: a retrospective. Int J Comput Vis 111(1):98–136
9. van Ginneken B, Setio A, Jacobs C, Ciompi F (2015) Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans. In: IEEE ISBI, pp 286–289
10. Bar Y, Diamant I, Greenspan H, Wolf L (2015) Chest pathology detection using deep learning with non-medical training. In: IEEE ISBI
11. Shin H, Lu L, Kim L, Seff A, Yao J, Summers R (2015) Interleaved text/image deep mining on a large-scale radiology image database. In: IEEE Conference on CVPR, pp 1–10
12. Ciompi F, de Hoop B, van Riel SJ, Chung K, Scholten E, Oudkerk M, de Jong P, Prokop M, van Ginneken B (2015) Automatic classification of pulmonary peri-fissural nodules in computed tomography using an ensemble of 2d views and a convolutional neural network out-of-the-box. Med Image Anal 26(1):195–202
13. Menze B, Reyes M, Van Leemput K (2015) The multimodal brain tumor image segmentation benchmark (brats). IEEE Trans Med Imaging 34(10):1993–2024
14. Pan Y, Huang W, Lin Z, Zhu W, Zhou J, Wong J, Ding Z (2015) Brain tumor grading based on neural networks and convolutional neural networks. In: IEEE EMBC, pp 699–702
15. Shen W, Zhou M, Yang F, Yang C, Tian J (2015) Multi-scale convolutional neural networks for lung nodule classification. In: IPMI, pp 588–599
16. Carneiro G, Nascimento J, Bradley AP (2015) Unregistered multiview mammogram analysis with pre-trained deep learning models. In: MICCAI, pp 652–660
17. Wolterink JM, Leiner T, Viergever MA, Isgum I (2015) Automatic coronary calcium scoring in cardiac CT angiography using convolutional neural networks. In: MICCAI, pp 589–596
18. Schlegl T, Ofner J, Langs G (2014) Unsupervised pre-training across image domains improves lung tissue classification. In: Medical computer vision: algorithms for big data. Springer, Berlin, pp 82–93
19. Hofmanninger J, Langs G (2015) Mapping visual features to semantic profiles for retrieval in medical imaging. In: IEEE conference on CVPR
20. Carneiro G, Nascimento J (2013) Combining multiple dynamic models and deep learning architectures for tracking the left ventricle endocardium in ultrasound data. IEEE Trans Pattern Anal Mach Intell 35(11):2592–2607
21. Li R, Zhang W, Suk H, Wang L, Li J, Shen D, Ji S (2014) Deep learning based imaging data completion for improved brain disease diagnosis. In: MICCAI
22. Barbu A, Suehling M, Xu X, Liu D, Zhou SK, Comaniciu D (2012) Automatic detection and segmentation of lymph nodes from CT data. IEEE Trans Med Imaging 31(2):240–250

23. Feulner J, Zhou SK, Hammon M, Hornegger J, Comaniciu D (2013) Lymph node detection and segmentation in chest CT data using discriminative learning and a spatial prior. *Med Image Anal* 17(2):254–270
24. Feuerstein M, Glocker B, Kitasaka T, Nakamura Y, Iwano S, Mori K (2012) Mediastinal atlas creation from 3-d chest computed tomography images: application to automated detection and station mapping of lymph nodes. *Med Image Anal* 16(1):63–74
25. Lu L, Devarakota P, Vikal S, Wu D, Zheng Y, Wolf M (2014) Computer aided diagnosis using multilevel image features on large-scale evaluation. In: *Medical computer vision. Large data in medical imaging*. Springer, Berlin, pp 161–174
26. Lu L, Bi J, Wolf M, Salganicoff M (2011) Effective 3d object detection and regression using probabilistic segmentation features in CT images. In: *IEEE CVPR*
27. Roth H, Lu L, Liu J, Yao J, Seff A, Cherry KM, Turkbey E, Summers R (2016) Improving computer-aided detection using convolutional neural networks and random view aggregation. In: *IEEE Transaction on Medical Imaging*
28. Lu L, Barbu A, Wolf M, Liang J, Salganicoff M, Comaniciu D (2008) Accurate polyp segmentation for 3d CT colonography using multi-staged probabilistic binary learning and compositional model. In: *IEEE CVPR*
29. Tajbakhsh N, Gotway MB, Liang J (2015) Computer-aided pulmonary embolism detection using a novel vessel-aligned multi-planar image representation and convolutional neural networks. In: *MICCAI*
30. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. *Int J Comput Vis* 60(2):91–110
31. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: *IEEE CVPR*, vol 1, pp 886–893
32. Torralba A, Fergus R, Weiss Y (2008) Small codes and large image databases for recognition. In: *IEEE CVPR*, pp 1–8
33. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Rabinovich A (2015) Going deeper with convolutions. In: *IEEE conference on CVPR*
34. Chatfield K, Simonyan K, Vedaldi A, Zisserman A (2015) Return of the devil in the details: delving deep into convolutional nets. In: *BMVC*
35. Chatfield K, Lempitsky VS, Vedaldi A, Zisserman A (2011) The devil is in the details: an evaluation of recent feature encoding methods. In: *BMVC*
36. Seff A, Lu L, Barbu A, Roth H, Shin H-C, Summers R (2015) Leveraging mid-level semantic boundary cues for computer-aided lymph node detection. In: *MICCAI*
37. Depeursinge A, Vargas A, Platon A, Geissbuhler A, Poletti P-A, Müller H (2012) Building a reference multimedia database for interstitial lung diseases. *Comput Med Imaging Graph* 36(3):227–238
38. Song Y, Cai W, Zhou Y, Feng DD (2013) Feature-based image patch approximation for lung tissue classification. *IEEE Trans Med Imaging* 32(4):797–808
39. Song Y, Cai W, Huang H, Zhou Y, Feng D, Wang Y, Fulham M, Chen M (2015) Large margin local estimate with applications to medical image classification. *IEEE Transaction on Medical Imaging*
40. Seff A, Lu L, Cherry KM, Roth HR, Liu J, Wang S, Hoffman J, Turkbey EB, Summers R (2014) 2d view aggregation for lymph node detection using a shallow hierarchy of linear classifiers. In: *MICCAI*, pp 544–552
41. Gao M, Bagci U, Lu L, Wu A, Buty M, Shin H-C, Roth H, Papadakis ZG, Depeursinge A, Summers R, Xu Z, Mollura JD (2015) Holistic classification of CT attenuation patterns for interstitial lung diseases via deep convolutional neural networks. In: *MICCAI first workshop on deep learning in medical image analysis*
42. Lu L, Liu M, Ye X, Yu S, Huang H (2011) Coarse-to-fine classification via parametric and nonparametric models for computer-aided diagnosis. In: *ACM conference on CIKM*, pp 2509–2512
43. Farabet C, Couprie C, Najman L, LeCun Y (2013) Learning hierarchical features for scene labeling. *IEEE Trans Pattern Anal Mach Intell* 35(8):1915–1929

44. Mostajabi M, Yadollahpour P, Shakhnarovich G (2014) Feedforward semantic segmentation with zoom-out features. [arXiv:1412.0774](https://arxiv.org/abs/1412.0774)
45. Gao M, Xu Z, Lu L, Nogues I, Summers R, Mollura D (2016) Segmentation label propagation using deep convolutional neural networks and dense conditional random field. In: IEEE ISBI
46. Wang H, Suh JW, Das SR, Pluta JB, Craigie C, Yushkevich P et al (2013) Multi-atlas segmentation with joint label fusion. *IEEE Trans Pattern Anal Mach Intell* 35(3):611–623
47. Oquab M, Bottou L, Laptev I, Sivic J (2015) Is object localization for free?—weakly-supervised learning with convolutional neural networks. In: IEEE CVPR, pp 685–694
48. Oquab M, Bottou L, Laptev I, Josef S (2015) Learning and transferring mid-level image representations using convolutional neural networks. In: IEEE CVPR, pp 1717–1724
49. Zhu X, Vondrick C, Ramanan D, Fowlkes C (2012) Do we need more training data or better models for object detection. In: BMVC
50. Ciresan D, Giusti A, Gambardella L, Schmidhuber J (2013) Mitosis detection in breast cancer histology images with deep neural networks. In: MICCAI
51. Zhang W, Li R, Deng H, Wang L, Lin W, Ji S, Shen D (2015) Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. *NeuroImage* 108:214–224
52. Li Q, Cai W, Wang X, Zhou Y, Feng DD, Chen M (2014) Medical image classification with convolutional neural network. In: IEEE ICARCV, pp 844–848
53. Miller GA (1995) Wordnet: a lexical database for english. *Commun ACM* 38(11):39–41
54. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick RB, Guadarrama S, Darrell T (2014) Caffe: convolutional architecture for fast feature embedding. *ACM Multimed* 2:4
55. Razavian AS, Azizpour H, Sullivan J, Carlsson S (2014) Cnn features off-the-shelf: an astounding baseline for recognition. In: IEEE CVPRW, pp. 512–519
56. Zhou B, Lapedriza A, Xiao J, Torralba A, Oliva A (2014) Learning deep features for scene recognition using places database. In: NIPS, pp 487–495
57. Gupta S, Girshick R, Arbelaez P, Malik J (2014) Learning rich features from rgb-d images for object detection and segmentation. In: ECCV, pp 345–360
58. Gupta S, Arbelaez P, Girshick R, Malik J (2015) Indoor scene understanding with rgb-d images: bottom-up segmentation, object detection and semantic segmentation. *Int J Comput Vis* 112(2):133–149
59. Gupta A, Ayhan M, Maida A (2013) Natural image bases to represent neuroimaging data. In: ICML, pp 987–994
60. Chen H, Dou Q, Ni D, Cheng J, Qin J, Li S, Heng P (2015) Automatic fetal ultrasound standard plane detection using knowledge transferred recurrent neural networks. In: MICCAI, pp 507–514
61. Roth H, Lu L, Farag A, Shin H-C, Liu J, Turkbey E, Summers R (2015) Deeporgan: multi-level deep convolutional networks for automated pancreas segmentation. In: MICCAI
62. Kim L, Roth H, Lu L, Wang S, Turkbey E, Summers R (2014) Performance assessment of retroperitoneal lymph node computer-assisted detection using random forest and deep convolutional neural network learning algorithms in tandem. In: The 102nd annual meeting of radiological society of North America
63. Holmes III D, Bartholmai B, Karwoski R, Zavaleta V, Robb R (2006) The lung tissue research consortium: an extensive open database containing histological, clinical, and radiological data to study chronic lung disease. In: 2006 MICCAI open science workshop
64. Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y (2014) Overfeat: integrated recognition, localization and detection using convolutional networks. In: ICLR
65. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. In: ICLR
66. Hochreiter S (1998) The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int J Uncertain Fuzziness Knowl-Based Syst* 6(02):107–116
67. Hinton GE, Osindero S, Teh Y-W (2006) A fast learning algorithm for deep belief nets. *Neural Comput* 18(7):1527–1554

68. Bengio Y, Simard P, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. *IEEE Trans Neural Netw* 5(2):157–166
69. Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks. In: *ECCV*, pp 818–833

Chapter 9

Cell Detection with Deep Learning Accelerated by Sparse Kernel

Junzhou Huang and Zheng Xu

Abstract As lung cancer is one of the most frequent and serious disease causing death for both men and women, early diagnosis and differentiation of lung cancers is clinically important. Computerized tissue histopathology image analysis and computer-aided diagnosis is very efficient and has become amenable. The cell detection process is the most basic step among the computer-aided histopathology image analysis applications. In this chapter, we study a deep convolutional neural network-based method for the lung cancer cell detection problem. This problem is very challenging due to many reasons, e.g., cell clumping and overlapping, high complexity of the cell detection methods, and the lack of humanly annotated datasets. To address these issues, we introduce a deep learning-based cell detection method for the effectiveness, as the deep learning methods have been demonstrated to be repeatedly successful in various computer vision applications in the last decade. However, this method still takes very long time to detect cells in very small images, e.g., 512×512 , albeit it is very effective in the cell detection task. In order to reduce the overall time cost of this method, we combine this method with the *sparse kernel* technique to significantly accelerate the cell detection process, up to 500 times. With the aforementioned advances, our numerical results confirm that the resulting method is able to outperform most state-of-the-art cell detection methods in terms of both efficiency and effectiveness.

9.1 Introduction

Pathology is defined as the science of the causes and effects of diseases, especially the branch of medicine that deals with the laboratory examination of samples of body tissue for diagnostic or forensic purposes. Conventionally, tissue samples were

J. Huang (✉) · Z. Xu

University of Texas at Arlington, 701 S. Nedderman Dr., Arlington, TX 76013, USA
e-mail: jzhuang@uta.edu

Z. Xu

e-mail: zheng.xu@mavs.uta.edu

taken by the pathologists from the human body, placed under a microscope and were examined for diagnostic purposes. But over the time, this became labor intensive as the diseases started evolving more complex. But the recent technological advances, like Whole Slide Digital scanners are able to digitize the tissue samples, store them as images are helping the pathologists by reducing their manual labor. Another important part of these preparation is the staining. In this process, different components of the tissue are stained with different dyes, so as to give the pathologists a clear idea of what they are looking at.

The primary aim of staining is to reveal cellular components and counterstains are used to provide contrast. One such staining methodology is the Hematoxylin–Eosin (H&E) staining, that has been used by pathologists for over a hundred years. Histopathology images are stained using this methodology. Hematoxylin stains cell nuclei blue, while Eosin stains cytoplasm and connective tissue pink [2]. One other staining methodology is the immunohistochemical (IHC) staining, which is used to diagnose whether the cancer is malignant or benign, to determine the stage of a tumor and to determine which cell type is at the origin of the tumor. After this process of staining, fast slide scanners are used to digitize the sample into images, which provide detailed and critical information at the microscopic level.

Pathologists have been aware of the importance of quantitative analysis of the pathological images. Quantitative analysis provides pathologists clearer insights of the presence or absence of a disease, their progress, nature, grade, etc. This was the point where the need for computer-aided diagnosis (CAD) aroused. Now, CAD has become a research area in the field of medical imaging and diagnostic radiology. It is now possible to use histological tissue patterns with computer-aided image analysis to facilitate disease classification. Thus, quantitative metrics for cancerous nuclei were developed to appropriately encompass the general observations of the experienced pathologist, and were tested on histopathological imagery. Though, several surveys have been published in this topic involving cancer/tumor detection and diagnosis in histopathological images [1–5], this survey aims at elaborating the techniques used for Cell Detection in particular in histopathology images.

9.1.1 Related Work

9.1.1.1 Unsupervised Learning Techniques

Unsupervised learning is the machine-learning task of inferring a function to describe hidden structure from unlabeled data. Since the examples given to the learner are unlabeled, there is no error or reward signal to evaluate a potential solution. This distinguishes unsupervised learning from supervised learning and reinforcement learning. There are actually two approaches to unsupervised learning. The first approach is to teach the algorithm not by giving explicit categorizations, but by using some sort of reward system to indicate success. This type of training will generally fit into the decision problem framework because the goal is not to produce a classification but

to make decisions that maximize rewards. The second type of unsupervised learning is called clustering. In this type of learning, the goal is not to maximize a utility function, but simply to find similarities in the training data. The assumption is often that the clusters discovered will match reasonably well with an intuitive classification. For instance, clustering individuals based on the demographics might result in a clustering of the wealthy in one group and the poor in another. This section and hence this survey focuses on the second method, clustering since this is the one which is most commonly used and is applicable to real world problems.

Correlational Clustering

This paper [6] proposes a cell detection algorithm which is applicable to different cell modalities and cell structures. This method first computes a cell boundary probability map from a trained edge classifier. The resulting predictions are used to obtain superpixels using a watershed transform. As the probability map is a smoothed one, it helps avoiding tiny noninformative superpixels, while still keeping boundaries with low probability separated from background superpixels. and a weighted region adjacency graph. Then an adjacency graph is built using the superpixels. Each graph edge e corresponds to an adjacent superpixel pair. But there arises a problem where negative potentials forces superpixels to be in separate regions, while positive ones favor to merge them. This adjacency graph-partitioning problem is then solved using Correlation Clustering.

SIFT Keypoint Clustering

This unstained cell detection algorithm [7] uses Scale Invariant Feature Transform (SIFT), a self-labeling algorithm, and two clustering steps to achieve effective cell detection in terms of detection accuracy and time. This paper uses bright field and phase contrast microscopic images for detection. This algorithm is heavily weighted on SIFT (a local image feature detector and descriptor) and its related techniques. Each detected keypoint is characterized by its spatial coordinates, a scale, an orientation, a difference of Gaussians (DOG) value, and principal curvatures ratio (PCR) value [7]. The DOG value indicates the keypoint strength and its value is positive for black-on-white blobs and negative for white-on-black blobs. Now, the actual cell detection is done using keypoint clustering and self-labeling. This done using a series of steps viz., (1) Keypoint Extraction, (2) Blob Type Detection, (3) Scale Adaptive Smoothing, (4) Second Keypoint Extraction, (5) Cell/Background Keypoint Clustering, and (6) Cell/Cell Keypoint Clustering. The proposed approach was evaluated on Five cell lines having in total 37 images and 7250 cells were considered for the evaluation: CHO, L929, Sf21, HeLa, and Bovine cells. The F1 measures on these data is between 85.1.

SIFT, Random Forest, and Hierarchical Clustering

This is almost similar to the previous algorithm, except it brings in Random Forest and Hierarchical Clustering instead of Keypoint Clustering [8]. The algorithm goes

like this: First, Keypoint learning is adopted as a calibration step before the actual cell detection procedure. This classifies the keypoints as background and cell keypoints. To check if two given keypoints belong to the same cell, a profile learning methodology is used. This is done by extracting the intensity profiles of the given keypoints. The profile features are extracted from these profiles and these are used to classify the profiles as inner or cross profile.

Now comes the Hierarchical Clustering part. This part combines the problems of Keypoint Learning and Profile Learning to detect cells. This can be achieved by constructing a graph, but this algorithm uses a technique called agglomerative hierarchical clustering (AHC) which uses a similarity measure between two cell keypoints. Finally, Hierarchical Clustering methods such as Linkage Method and Finding-the-hit-point method are used to detect cells. The model has been evaluated over a set of 3500 real and 30,000 simulated images, and the detection error was between 0 and 15.5.

9.1.1.2 Supervised Learning Techniques

Supervised learning is the machine-learning task of inferring a function from labeled training data. The training data consist of a set of training examples. In supervised learning, each example is a pair consisting of a vector and a desired output value. A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances. This requires the learning algorithm to generalize from the training data to unseen situations in a reasonable way. It is the most common technique for training neural networks and decision trees. Both of these techniques are highly dependent on the information given by the predetermined classifications. In the case of neural networks, the classification is used to determine the error of the network and then adjust the network to minimize it, and in decision trees, the classifications are used to determine what attributes provide the most information that can be used to solve the classification puzzle. This section focuses on the supervised learning techniques that are considered to be optimum for the application of cell detection.

Automatic Mitotic Cell Detection

Extracting the mitotic cell from the histopathological image is a very challenging task. This technique [9] consists of three modules viz., discriminative image generation, mitotic cell candidate detection and segmentation, and mitotic cell candidate classification. In the first module, a discriminative image is obtained by linear discriminant analysis. In the second module, classification is performed to detect real mitotic cells. In the third module, a 226-dimension feature is extracted from the mitotic cell candidates and their surrounding regions. An imbalanced classification framework is then applied to perform the classification for the mitotic cell candidates in order to detect the real-mitotic cells. The proposed technique provides 81.5% sensitivity rate

and 33.9% precision rate in terms of detection performance, and 89.3% sensitivity rate and 87.5% precision rate in terms of segmentation performance. Thus, the proposed work is intended to reduce the workload of pathologists when they evaluate the cancer grade of biopsy.

Cell Detection under Nonoverlapping Constraints

Robust cell detection in high-density and low-contrast images is still challenging since cells often touch and partially overlap, forming a cell cluster with blurry inter-cellular boundaries. In such cases, current methods [10] tend to detect multiple cells as a cluster. This also leads to many other problems. To solve the above-mentioned problems, the authors have formulated a series of steps. First, the redundant candidate regions that contain false positives and false negatives are detected. Second, a tree structure is generated in which the candidate regions are nodes, and the relationships between nodes are generated on the basis of information about overlapping. Third, the score for how likely the candidate regions contain the main part of a single cell is computed for each cell using supervised learning. Finally, the optimal set of cell regions from the redundant regions under nonoverlapping constraints is selected. Figure 9.2 shows the overview of the proposed approach. The proposed cell detection method addresses all of the difficulties in detecting dense cells simultaneously under high-density conditions, including the mistaken merging of multiple cells, the segmentation of single cells into multiple regions, and the misdetection of low-intensity cells. The system is evaluated over several types of cells in microscopy images, which achieved an overall F-measure of 0.9 on all types.

Improved Cell Detection using LoG

This is a benchmark algorithm, one of the most widely used papers for comparison. This algorithm [11] consists of a number of concepts executed in a sequential manner. They are (1) The staining phase where for the *in vivo* tissue samples, human breast tissues were stained with hematoxylin and for the *in vitro* tissue samples, frozen blocks of K1735 tumor cells were stained with DAPI. (2) The image capture where Images of hematoxylin or DAPI stained histopathology slides were captured using a Nuance multispectral camera mounted on a Leica epifluorescence microscope. (3) The automatic image binarization where foreground extraction is done using graph-cuts-based binarization. (4) Next, nuclear seed points are detected by a novel method combining multiscale Laplacian-of-Gaussian filtering constrained by distance-map-based adaptive scale selection, which are used to perform an initial segmentation that is refined using a second graph-cuts-based algorithm. (5) Refinement of Initial Nuclear Segmentation using—Expansions and Graph Coloring. The purpose of the refinement is to enhance the initial contours between touching nuclei to better delineate the true edges between them. (6) The last step is the Efficient Computer-Assisted Editing of Automated Segmentation Results. Though automatic segmentation would result in accurate result, some extent of manual intervention may be needed to fix some errors like over-segmentation and under-segmentation.

Fast Cell Detection from High-Throughput Microscopy

High-throughput microscopy has emerged as a powerful tool to analyze cellular dynamics in an unprecedentedly high resolved manner. Available software frameworks are suitable for high-throughput processing of fluorescence images, but they often do not perform well on bright field image data that varies considerably between laboratories, setups, and even single experiments. This algorithm [12] is an image-processing pipeline that is able to robustly segment and analyze cells with ellipsoid morphology from bright field microscopy in a high-throughput, yet time efficient manner. The pipeline comprises two steps: (i) Image acquisition is adjusted to obtain optimal bright field image quality for automatic processing. (ii) A concatenation of fast performing image-processing algorithms robustly identifies single cells in each image. This method allows fully automated processing and analysis of high-throughput bright field microscopy data. The robustness of cell detection and fast computation time will support the analysis of high-content screening experiments, online analysis of time-lapse experiments as well as development of methods to automatically track single-cell genealogies.

9.1.1.3 Deep Learning Techniques

Deep learning is a branch of machine-learning based on a set of algorithms that attempt to model high-level abstractions in data by using multiple processing layers with complex structures, or otherwise composed of multiple nonlinear transformations. Deep learning is part of a broader family of machine-learning methods based on the learning representations of data. An observation can be represented in many ways such as a vector of intensity values per pixel, or in a more abstract way as a set of edges, regions of particular shape, etc. Some representations make it easier to learn tasks (e.g., face recognition or facial expression recognition) from examples. One of the promises of deep learning is replacing handcrafted features with efficient algorithms for unsupervised or semi-supervised feature learning and hierarchical feature extraction. Research in this area attempts to make better representations and create models to learn these representations from large-scale unlabeled data.

Deep Convolutional Neural Network and Maximum-Weight Independent Set

This paper [13] proposes a novel algorithm for general cell detection problem. First, a set of cell detection candidates is generated using different algorithms with varying parameters. Second, each candidate is assigned a score by a trained deep convolutional neural network (DCNN). Finally, a subset of best detection results is selected from all candidates to compose the final cell detection results. The subset selection task is formalized as a maximum-weight independent set problem, which is designed to find the heaviest subset of mutually nonadjacent nodes in a graph.

The proposed algorithm is tested with two datasets: (1) 24 neuroendocrine (NET) tissue microarray (TMA) images and (2) 16 lung cancer images. The results show that the proposed algorithm achieves a precision of 0.91, a recall of 0.90, and

an F1-measure of 0.93 on NET and a precision of 0.90, a recall of 0.88, and an F1-measure of 0.92 on Lung cancer data.

Deep Learning-based Immune Cell Detection

Immunohistochemistry (IHC) staining can be used to determine the distribution and localization of the differentially expressed biomarkers of immune cells (such as T cells or B cells) in cancerous tissue for an immune response study. To manually count each subset of immune cells under a bright field microscope for each piece of IHC stained tissue is usually extremely tedious and time consuming. This makes automatic detection a necessity to find such cells in IHC images, but there are several challenges. A novel method [14] for automatic immune cell counting on digitally scanned images of IHC stained slides is proposed. This method uses a sparse color unmixing technique to separate the IHC image into multiple color channels that correspond to different cell structures.

Sparse Reconstruction and Stacked Denoising Autoencoders

This cell detection algorithm uses the sparse reconstruction with trivial templates and combines it with a stacked denoising autoencoder (sDAE). The sparse reconstruction handles the shape variations by representing a testing patch as a linear combination of shapes in the learned dictionary. Trivial templates are used to model the touching parts. The sDAE, trained with the original data and their structured labels, can be used for cell segmentation. This algorithm [15] achieves a precision of 0.96, a recall of 0.85 and an F-1 measure of 0.90, better than the state-of-the-art methods, such as Laplacian-of-Gaussian (LoG), iterative radial voting (IRV), and image-based tool for counting nuclei (ITCN), and single-pass voting (SPV).

Deep Voting

This is a convolutional neural network (CNN)-based hough voting method to localize nucleus centroids with heavy cluttering and morphological variations in microscopy images. This method [16], called as Deep Voting consists of two steps, (1) assign each local patch of an input image, several pairs of voting offset vectors, (2) collect the weighted votes from all the testing patches and compute final voting density map. The Salient Features of this algorithm are (1) The computation of local density map is similar to that of Parzen-window estimation, (2) This method requires only minimum annotation.

Structured Regression using Convolutional Neural Network

Cell detection is a crucial prerequisite for biomedical image analysis tasks such as cell segmentation. Unfortunately, the success of cell detection is hindered by the nature of microscopic images such as touching cells, background clutters, large variations in the shape and the size of cells, and the use of different image acquisition techniques. To alleviate these problems, a nonoverlapping extremal regions selection method is presented by C. Arteta et al., and achieves state-of-the-art performance on their

data sets. However, this work heavily relies on a robust region detector and thus the application is limited. This algorithm is a novel CNN-based structured regression model, which is able to handle touching cells, inhomogeneous background noises, and large variations in sizes and shapes. The proposed method only requires a few training images with weak annotations.

9.1.2 Challenges

Cell Detection is still an open research area and there is still scope of improvement in this field [17–22]. Indeed, there are open challenges which shall be addressed in future research. The challenges include heterogeneity of the data, where the cells differ in size, shape, orientation, and so on. Another major challenge is the use of different datasets, which arises a need for a single benchmark dataset. Such a benchmark dataset would eliminate the difference in the results caused by the difference in the dataset and type of the data [23–31]. Another important challenge to be looked upon, is the robustness of the algorithms that were proposed and being proposed. Though the datasets are different, the proposed algorithms must be robust to data and environmental changes. This would give clearer insights [32–39]. Also, overlapping and clustered nuclei is another major challenge in cell detection and segmentation [40–44].

9.1.2.1 Pixel-Wise Detector with Its Acceleration

In this chapter, we study a fully automatic lung cancer cell detection method using DCNN with its acceleration variant. In the proposed method, the training process is only performed on the local patches centered at the weakly annotated dot in each cell area with the non-cell area patches of the same amount as the cell areas. This means only weak annotation of cell area (a single dot near the center of cell area) are required during labeling process, significantly relieving the manual annotation burden. This training technique also decreases the training time cost as it usually feeds less than one percent pixel patches of a training images to the proposed model, even when the cell density is high. Another benefit for this technique is to reduce the over-fitting effect and make the proposed method general enough to detect the rough cell shape information in the training image, providing the benefit for further applications, e.g., cell counting, segmentation and tracking.

During testing stage, the very first strategy is using the conventional sliding window manner to perform the pixel-wise cell detection. However, the conventional sliding window manner for all local pixel patches is inefficient due to the considerable redundant convolution computation. To accelerate the testing process for each testing image, we present a fast-forwarding technique in DCNN framework. Instead of performing DCNN forwarding in each pixel patch, the proposed method performs convolution computation in the entire testing image, with a modified sparse

convolution kernel. This technique almost eliminates all redundant convolution computation compared to the conventional pixel-wise classification, which significantly accelerates the DCNN forwarding procedure. Experimental result reports the proposed method only requires around 0.1 s to detect lung cancer cells in a 512×512 image, while the state-of-the-art DCNN requires around 40 s.

To sum up, we propose a novel DCNN-based model for lung cancer cell detection in this paper. Our contributions are summarized as three parts: (1) We built-up a deep learning-based framework in lung cancer cell detection with modified sliding window manner in both training and testing stage. (2) We modify the training strategy by only acquiring weak annotations in the samples, which decreases both labeling and training cost. (3) We present a novel accelerated DCNN forwarding technology by reducing the redundant convolution computation, accelerating the testing process several hundred times than the traditional DCNN-based sliding window method. To the best of our knowledge, this is the first study to report the application of accelerated DCNN framework for lung cancer cell detection.

9.2 Pixel-Wise Cell Detector

9.2.1 Overview

An overview of our method is presented in Fig. 9.1. The proposed framework [45] is based on the pixel-wise lung cancer cell detection over CNN. Slides of hematoxylin and eosin stained lung biopsy tissue are scanned at $40\times$ magnification. Since the extreme high resolution of the original slide, smaller image patches are randomly extracted as the datasets. There are two parts, training part and cell detection testing part. In the training part, the provided training images are preprocessed into image patches at first. There are two types of image patches, positive image patches and negative image patches, which are produced according to the human annotation of

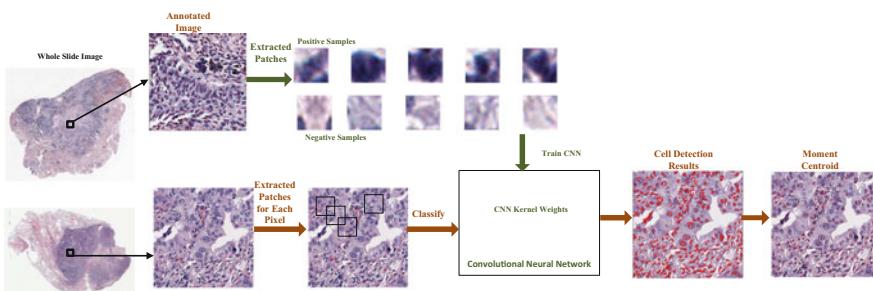


Fig. 9.1 Overview of the proposed framework. Both local appearance and holistic architecture features are extracted from image tile from the whole slide image

the training images. Then, the convolutional neural network is trained using the manually annotated image patches of the training data. In the cell detection testing part, the testing image is input into the trained CNN to get the output possibility map, and then the image moment analysis method is applied to get the lung cancer cell centroids.

9.2.2 Deep Convolutional Neural Network

Deep Convolutional Neural Network (DCNN) have been developed to be a powerful framework in the domains of image processing, DCNN is a type of feed forward artificial neural network, which is very effective in the extraction of the hierarchical feature expressions for the image classification and recognition tasks without any prior knowledge of domain-specific image feature. Convolution layer and pooling layer are DCNN typical layers. The convolution layer performs the convolution of input feature maps with filter, followed by an optional a point-wise nonlinear function to produce the output feature maps. The filter is a rectangular kernel, which extracts the same type of local features in every possible position of the input map. The pooling layer takes just one value from a subwindow of the input map, which makes the resolution of the feature maps decrease to make the output feature maps keep invariance to local deformations. Max-pooling and average-pooling are most commonly used operations. After extracting features with several pairs of convolution layer and pooling layer, fully connected layers mix the output features into the complete feature vector. The output layer is the fully connected layer with the classification prediction as the output result. Therefore, DCNN is a classification system that performs automatic feature extraction and classification procedures together.

Our DCNN model is a 7-layer network(excluding input) with 2 convolutional layers(C), 2 max-pooling layers(MP), 1 fully connected layer(FC), 1 rectified linear unit layer followed by the output layer, which is the special case of fully connected layer with the softmax function as the activation function with two output classes(lung cancer cell or non-cell), as shown in Fig. 9.2. The architecture and mapped proximity patches of our proposed DCNN model is illustrated in Fig. 9.2. The detailed configuration of our DCNN is: Input $(20 \times 20 \times 3)$ - C $(16 \times 16 \times 20)$ - MP $(8 \times 8 \times 20)$ - C $(4 \times 4 \times 50)$ - MP $(2 \times 2 \times 50)$ - FC (500) - ReLu (500) - FC (2) . the sizes of the layers are defined as width*height*depth, where depth represents the number of feature maps and width*height represents the dimensionality of the feature map. the filter size of the convolution layer is 5×5 , the max-pooling layer is 2×2 with the stride of 2. For the output layer, the activation function is softmax function, i.e., $\text{softmax}(x) = \exp(x)/Z$, where Z is a normalization constant. This function convert the computation result x into positive values (the summation to the values is one), so as to be interpreted as a probability distribution.

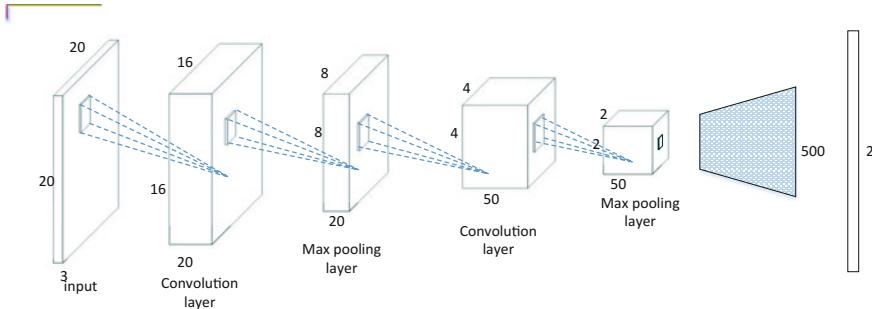


Fig. 9.2 DCNN architecture used for the experiments

9.2.3 Implementation

The training set is composed of the image patches abstracted from the weakly annotated images. All the true lung cancer cell centers are manually annotated in the training images. The task of training data preparation is to generate image patches according to the annotated lung cancer cell centers. We extract the square image patches from the training images within the distance d_1 from the lung cancer cell center as positive training samples. The square image patches from the training images that locate far from the lung cancer cell centers are negative training samples. As the number of pixels marked as the negative is far more than the number as positive, we randomly selected the same number of negative image patches avoiding the over-fitting problems.

Our implementation is base on Caffe framework [46], Caffe provides a clean and modifiable framework for state-of-the-art deep learning algorithms and a collection of reference models. It also supports the use of GPU to accelerate the execution of deep learning algorithms.

In the testing stage, we first abstract the image patches centered at each pixel of the testing image, then apply the trained DCNN classifier to these patches producing the output probability labels for the patch.

Our method is also able to approximate the location of cell nuclei based on the rough shape information provided by the proposed patch-by-patch manner. For each cell area, we estimate the centroid of the lung cancer cell area as the nuclei location via the *image raw moments*.

$$M_{p,q,i} = \sum_x \sum_y x^p y^q I_i(x, y), \quad (9.1)$$

where x, y indicate the pixel location coordinate, i denotes the i th cell area. $I_i(x, y)$ is the image intensity function for the binary image at i th cell area. $I_i(x, y) = 1$ if pixel located at (x, y) is in the i th cell area, otherwise $I_i(x, y) = 0$. With image raw moments calculated, we are able to approximate the centroid of i th cell area:

$$(x_i, y_i) = \left(\frac{M_{1,0,i}}{M_{0,0,i}}, \frac{M_{0,1,i}}{M_{0,0,i}} \right) \quad (9.2)$$

Given an input lung cancer histopathological image I , the problem is to find a set $D = \{d_1, d_2, \dots, d_N\}$ of detections, each reporting the centroid coordinates for a single-cell area. The problem is solved by training a detector on training images with given weakly annotated ground-truth information $G = \{g_1, g_2, \dots, g_M\}$, each representing the manually annotated coordinate near the center of each cell area. In the testing stage, each pixel is assigned one of two possible classes, *cell* or *non-cell*, former to pixels in cell areas, the latter to all other pixels. Our detector is a DCNN-based pixel-wise classifier. For each given pixel p , the DCNN predicts its class using raw RGB values in its local square image patch centered on p .

9.3 Sparse Kernel Acceleration of the Pixel-Wise Cell Detector

9.3.1 *Training the Detector*

Using the weakly annotated ground-truth data G , we label each patch centered on the given ground-truth g_m as positive(*cell*) sample. Moreover, we randomly sample the negative(*non-cell*) samples from the local pixel patches whose center are outside of the boundary of positive patches. The amount of negative sample patches is the same as the positive ones. If a patch window lies partly outside of the image boundary, the missing pixels are fetched in the mirror padded image.

For these images, we only feed very few patches into the proposed model for training, therefore extremely accelerating the training stage. Besides, this technique also partly eliminates the effect of over-fitting due to the under-sampling usage of sample images (Fig. 9.3).

9.3.2 *Deep Convolution Neural Network Architecture*

Our DCNN model [47] contains two pairs of convolution and max-pooling layers, followed by a fully connected layer, rectified linear unit layer and another fully connected layer as output. Figure 9.4 illustrates the network architecture for training stage. Each **convolution layer** performs a 2D-convolution operation with a square filter. If the activation from previous layer contains more than one map, they are summed up first and then convoluted. In the training process, the stride of **max-pooling layer** is set the same as its kernel size to avoid overlap, provide more non-linearity and reduce dimensionality of previous activation map. The **fully connected layer** mixes the output from previous map into the feature vector. A **rectified linear**

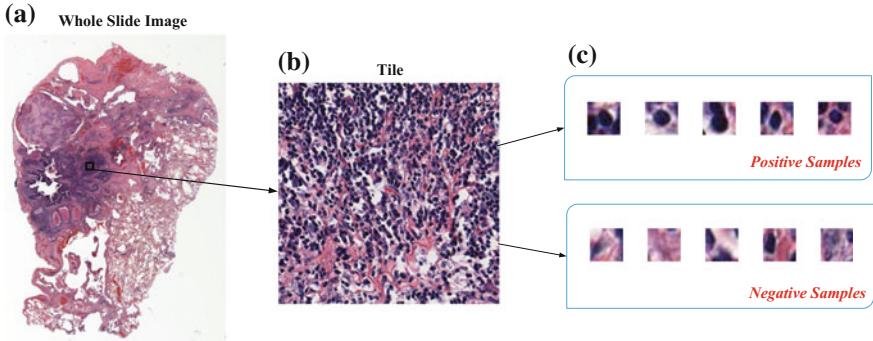


Fig. 9.3 The illustration of generation of training samples: **a** Tiles are randomly sampled from the whole slide images. **b** The sampled tiles are manually annotated by well-trained pathologists, which construct the weakly annotated information. **c** We only feed the local pixels patches center on the annotated pixels and the randomly sampled non-cell patches of the same amount as the cell ones

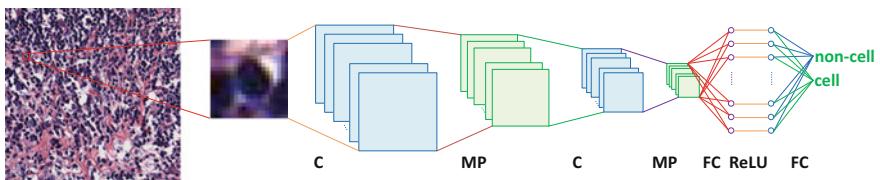


Fig. 9.4 The DCNN architecture used in the training process of the proposed framework. C, MP, FC, ReLU represents the convolution layer, max-pooling layer, fully connected layer, and rectified linear unit layer, respectively

unit layer is followed because of its superior nonlinearity. The output layer is simply another fully connected layer with just two neurons(one for cell class, the other for non-cell class), activated by a softmax function to provide the final possibility map for the two classes. We detail the layer type, neuron size, filter size, and filter number parameters of the proposed DCNN framework in the left of Table 9.1.

9.3.3 Acceleration of Forward Detection

The traditional sliding window manner requires the patch-by-patch scanning for all the pixels in the same image. It sequentially and independently feeds patches to DCNN and the forward propagation is repeated for all the local pixel patches. However, this strategy is time consuming due to the fact that there exists a lot of redundant convolution operations among adjacent patches when computing the sliding windows.

To reduce the redundant convolution operations, we utilize the relations between adjacent local image patches. In the proposed acceleration model, at the testing stage,

Table 9.1 Backward (left) and accelerated forward (right) network architecture. M : the number of patch samples, N : the number of testing images. Layer type: I - Input, C - Convolution, MP - Max Pooling, ReLU - Rectified Linear Unit, FC - Fully Connected

Type	Maps and neurons	Filter size	Filter num	Stride	Type	Maps and neurons	Filter size	Filter num	Stride
I	$3 \times 20 \times 20M$	–	–	–	I	$3 \times 531 \times 531N$	–	–	–
C	$20 \times 16 \times 16M$	5	20	1	C	$20 \times 527 \times 527N$	5	20	1
MP	$20 \times 8 \times 8M$	2	–	2	MP	$20 \times 526 \times 526N$	2	–	1
C	$50 \times 4 \times 4M$	5	50	1	C	$50 \times 518 \times 518N$	9	50	1
MP	$50 \times 2 \times 2M$	2	–	2	MP	$50 \times 516 \times 516N$	3	–	1
FC	$500M$	1	–	–	FC(C)	$500 \times 512 \times 512N$	5	–	1
ReLU	$500M$	1	–	–	ReLU	$500 \times 512 \times 512N$	1	–	–
FC	$2M$	1	–	–	FC(C)	$2 \times 512 \times 512N$	1	–	–

the proposed model takes the whole input image as input and can predict the whole label map with just one pass of the accelerated forward propagation. If a DCNN takes $n \times n$ image patches as inputs, a testing image of size $h \times w$ should be padded to size $(h + n - 1) \times (w + n - 1)$ to keep the size consistency of the patches centered at the boundary of images. The proposed method, in the testing stage, uses the exact weights solved in the training stage to generate the exactly same result as the traditional sliding window method does. To achieve this goal, we involve the k -sparse kernel technique [48] for convolution and max-pooling layers into our approach. The k -sparse kernels are created by inserting all-zero rows and columns into the original kernels to make every two original neighboring entries k -pixel away. To accelerate the forward process of fully connect layer, we treat fully connected layer as a special convolution layer. Then, the fully connect layer could be accelerated by the modified convolution layer. The proposed fast forwarding network is detailed in Table 9.1 (right). Experimental results show that around 400 times speedup is achieved on 512×512 testing images for forward propagation (Fig. 9.5).

9.4 Experiments

9.4.1 Materials and Experiment Setup

Data Set

The proposed method is evaluated on part of the National Lung Screening Trial (NLST) data set [49]. Totally 215 tile images of size 512×512 are selected from the original high-resolution histopathological images. The nuclei in these tiles are manually annotated by the well-trained pathologist. The selected dataset contains a total of 83245 nuclei objects.

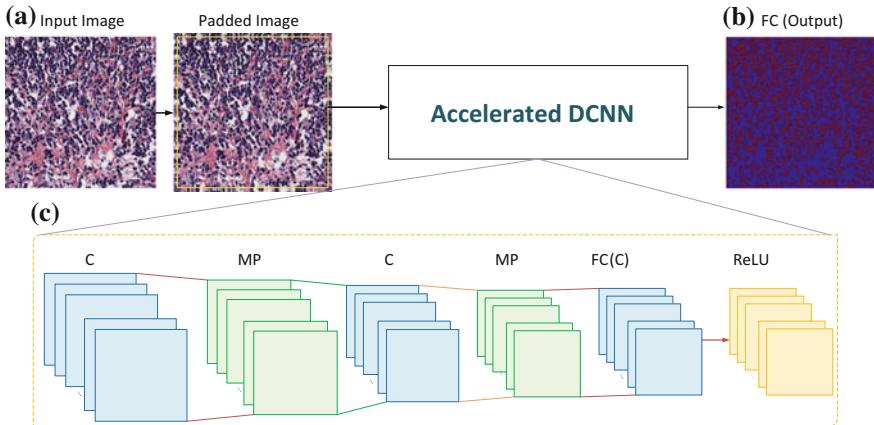


Fig. 9.5 The illustration of acceleration forward net: 1 The proposed method takes the whole image as input in testing stage. 2 The input image is mirror padded as the sampling process in the training stage. 3 The padded image is then put into the accelerated forward network which generates the whole label map in the rightmost. Note that the fully connected layer is implemented via a modified convolution layer to achieve acceleration

Experiments Setup

We partition the 215 images into three subsets: training set (143 images), validation set (62 images) and evaluation set (10 images). The evaluation result is reported on evaluation subset containing 10 images. We compare the proposed method with the state-of-the-art method in cell detection [50] and the traditional DCNN-based sliding window method [51]. For fair comparisons, we download the code from their websites and follow their default parameter settings carefully.

Infrastructure and Implementation Details

All experiments in this paper are conducted on a Workstation with Intel(R) Xeon(R) CPU E5-2620 v2 @ 2.10GHz CPU, 32 gigabyte RAM. The computation involved GPU computing is performed on a nVidia Tesla K40c GPU with 12 gigabytes memory. The training process of proposed method is implemented based on the Caffe framework [46].

Evaluation Metrics

For quantitative analysis, we define the ground-truth areas as circular regions within 8 pixels for every annotated cell center [50]. Since the proposed method detects the cell area shape, we calculate the **raw image moment** centroid as its approximate nuclei location. A detected cell centroid is considered to be a true positive (*TP*) sample if the circular area of radius 8 centered at the detected nuclei contains the ground-truth annotation; otherwise, it is considered as False Positive (*FP*). Missed ground-truth dots are counted as False Negatives (*FN*). The results are reported in terms of F_1 score $F_1 = 2PR/(P + R)$, where precision $P = TP/(TP + FP)$ and recall $R = TP/(TP + FN)$.

9.4.2 Results

Training Time Cost

The mean training time for the proposed method is 229 s for the training set described below. The unaccelerated version with the same training strategy costs the same time as the proposed method. Besides, the state-of-the-art MSER-based method [50] costs more than 40,0000 s, roughly 5 days for training 143 images of size 512×512 . The proposed method is able to impressively reduce several thousand times time cost of training stage than the state-of-the-art MSER-based method due to the proposed training strategy.

Accuracy of Testing

Table 9.2 reports the F_1 score metric comparison between the proposed method and MSER-based method. The proposed method outperforms the state-of-the-art method in almost all of the evaluation images in terms of F_1 scores. We also visually compares our results with the MSER-based method in Fig. 9.6. The proposed method detects almost all of the cell regions even in images with intensive cells.

Testing Time Cost

As shown in Fig. 9.7, the proposed method only costs around 0.1 s for a single 512×512 tile image, which is the fastest among the three methods. The proposed method accelerates the forwarding procedure around 400 times compared with the traditional pixel-wise sliding-window method, which is due to the accelerated forwarding technique.

9.5 Discussion

The aforementioned LeNet- based methods grant us some basic frameworks for the cell detection task. However, there are still many more challenges left for us to solve.

Huge Data Scale

When the histopathological image scale grows to some insane level, e.g., 10^{10} pixels. The current framework, even with the sparse kernel acceleration, is not able to classify

Table 9.2 F_1 scores on the evaluation set

	1	2	3	4	5	6	7	8	9	10	Mean
MSER [50]	0.714	0.633	0.566	0.676	0.751	0.564	0.019	0.453	0.694	0.518	0.559
Proposed	0.790	0.852	0.727	0.807	0.732	0.804	0.860	0.810	0.770	0.712	0.786

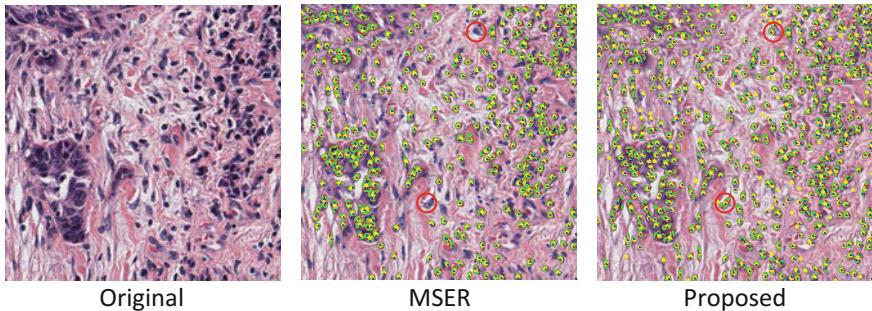


Fig. 9.6 Visual Comparison between the proposed method and MSER-based method [50]. The green area denotes the detected cell area by the corresponding method. Blue dots denote the ground-truth annotation. The proposed method is able to detect the cell area missed by the MSER-based method as denoted in red circle. Better viewed in $\times 4$ pdf

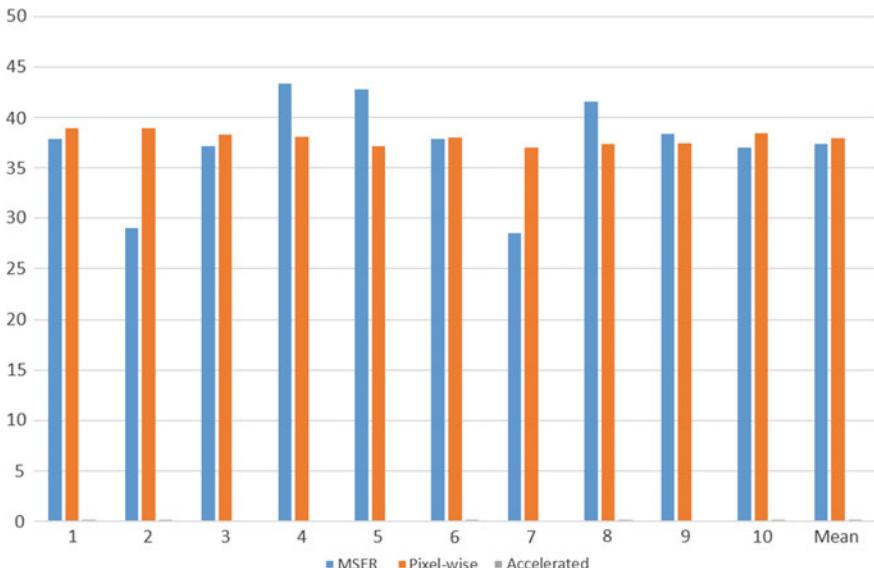


Fig. 9.7 Mean time cost comparison on the evaluation set

all cell pixels within a reasonable time frame. However, as we observe some most recent success from the distributed computing and parallel GPU computing, the actual performance of the huge-scale pixel-wise cell detection could benefit from these techniques from the distributed computing.

Hardware I/O Burden

As mentioned before, the data scale can insanely grow to a huge level. One other problem is the disk I/O cost, which is usually the bottleneck of the whole slide cell

detection process. As a result, choosing the appropriate I/O strategy is one of the major task in optimizing the overall performance of the pixel level cell detection. Therefore, dealing with it could be a future work. A potential direction could be seen in [52].

Network Structure

In this chapter, we only study a very fundamental neural network structure, i.e., LeNet [51]. There are several other advanced pixel-wise classifiers to study further, e.g., AlexNet [53], GoogLeNet [54]. We might also try more complicated tasks, e.g., subtype detection [55].

Optimization Strategy

At present, most deep learning methods are still using basic stochastic gradient descent to train their neural network. While we are aware of some recent advance in the stochastic optimization area [56], we can expect to accelerate the training process.

Model Compression

Recently, the research community leans to develop very deep neural networks to significantly improve the performance. However, the hardware memory usually limits this trend. To resolve this problem, we can involve some model compression techniques to compress the model [57] so that the current memory can fit in deeper models.

9.6 Conclusion

In this chapter, we have discussed a DCNN-based cell detection method with its sparse kernel acceleration. The proposed method is designed based on the DCNN framework [46], which is able to provide state-of-the-art accuracy with only weakly annotated ground truth. For each cell area, only one local patch containing the cell area is fed into the detector for training. The training strategy significantly reduces the time cost of training procedure due to the fact that only around one percent of all pixel labels are used. In the testing stage, we modified the training network to accept the input of the whole image. By utilizing the relation of adjacent patches, in the forwarding propagation, the proposed method provides the exact same result within a few hundredths time. Experimental results clearly demonstrate the efficiency and effectiveness of the proposed method for large-scale lung cancer cell detection.

References

1. Demir C, Yener B (2005) Automated cancer diagnosis based on histopathological images: a systematic survey. Tech. Rep., Rensselaer Polytechnic Institute
2. Gurcan MN, Boucheron LE, Can A, Madabhushi A, Rajpoot NM, Yener B (2009) Histopathological image analysis: a review. *IEEE Rev Biomed Eng* 2:147–171
3. Irshad H, Veillard A, Roux L, Racocian D (2014) Methods for nuclei detection, segmentation, and classification in digital histopathology: a review current status and future potential. *IEEE Rev Biomed Eng* 7:97–114
4. Rathore S, Hussain M, Ali A, Khan A (2013) A recent survey on colon cancer detection techniques. *IEEE/ACM Trans Comput Biol Bioinform* 10(3):545–563
5. Veta M, Pluim JPW, van Diest PJ, Viergever MA (2014) Breast cancer histopathology image analysis: a review. *IEEE Trans Biomed Eng* 61(5):1400–1411
6. Zhang C, Yarkony J, Hamprecht FA (2014) Cell detection and segmentation using correlation clustering. In: Medical image computing and computer-assisted intervention—MICCAI. Springer, Berlin, pp 9–16
7. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. *Int J Comput Vis* 60(2):91–110
8. Mualla F, Scholl S, Sommerfeldt B, Maier A, Horngger J (2013) Automatic cell detection in bright-field microscope images using sift, random forests, and hierarchical clustering. *IEEE Trans Med Imaging* 32(12):2274–2286
9. Lu C, Mandal M (2014) Toward automatic mitotic cell detection and segmentation in multi-spectral histopathological images. *IEEE J Biomed Health Inform* 18(2):594–605
10. Bise R, Sato Y (2015) Cell detection from redundant candidate regions under nonoverlapping constraints. *IEEE Trans Med Imaging* 34(7):1417–1427
11. Al-Kofahi Y, Lassoued W, Lee W, Roysam B (2010) Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Trans Biomed Eng* 57(4):841–852
12. Buggenthin F, Marr C, Schwarzfischer M, Hoppe PS, Hilsenbeck O, Schroeder T, Theis FJ (2013) An automatic method for robust and fast cell detection in bright field images from high-throughput microscopy. *BMC Bioinform* 14(1):297
13. Liu F, Yang L (2015) A novel cell detection method using deep convolutional neural network and maximum-weight independent set. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 349–357
14. Chen T, Chefchotel C (2014) Deep learning based automatic immune cell detection for immunohistochemistry images. In: Machine learning in medical imaging. Springer, Berlin, pp 17–24
15. Su H, Xing F, Kong X, Xie Y, Zhang S, Yang L (2015) Robust cell detection and segmentation in histopathological images using sparse reconstruction and stacked denoising autoencoders. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 383–390
16. Xie Y, Kong X, Xing F, Liu F, Su H, Yang L (2015) Deep voting: a robust approach toward nucleus localization in microscopy images. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 374–382
17. Afridi MJ, Liu X, Shapiro E, Ross A (2015) Automatic in vivo cell detection in mri. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 391–399
18. Chaudhury KN, Puspoki Z, Muñoz-Barrutia A, Sage D, Unser M (2010) Fast detection of cells using a continuously scalable mexican-hat-like template. In: 2010 IEEE international symposium on biomedical imaging: from nano to macro. IEEE, pp 1277–1280
19. Mayerich D, Kwon J, Panchal A, Keyser J, Choe Y (2011) Fast cell detection in high-throughput imagery using gpu-accelerated machine learning. In: 2011 IEEE international symposium on biomedical imaging: from nano to macro. IEEE, pp 719–723
20. Massoudi A, Semenovich D, Sowmya A (2012) Cell tracking and mitosis detection using splitting flow networks in phase-contrast imaging. In: 2012 annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE, pp 5310–5313

21. Dong B, Shao L, Da Costa M, Bandmann O, Frangi AF (2015) Deep learning for automatic cell detection in wide-field microscopy zebrafish images. In: 2015 IEEE 12th international symposium on biomedical imaging (ISBI). IEEE, pp 772–776
22. Xu Y, Mo T, Feng Q, Zhong P, Lai M, Chang EI et al (2014) Deep learning of feature representation with multiple instance learning for medical image analysis. In: 2014 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 1626–1630
23. Verbancsics P, Harguess J (2015) Image classification using generative neuro evolution for deep learning. In: 2015 IEEE winter conference on applications of computer vision (WACV). IEEE, pp 488–493
24. Liang M, Li Z, Chen T, Zeng J (2015) Integrative data analysis of multi-platform cancer data with a multimodal deep learning approach. *IEEE/ACM Trans Comput Biol Bioinform (TCBB)* 12(4):928–937
25. Xie Y, Xing F, Kong X, Su H, Yang L (2015) Beyond classification: Structured regression for robust cell detection using convolutional neural network. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 358–365
26. Yu Z, Chen H, You J, Wong H-S, Liu J, Li L, Han G (2014) Double selection based semi-supervised clustering ensemble for tumor clustering from gene expression profiles. *IEEE/ACM Trans Comput Biol Bioinform (TCBB)* 11(4):727–740
27. Ibrahim R, Yousri NA, Ismail MA, El-Makky NM (2014) Multi-level gene/mirna feature selection using deep belief nets and active learning. In: 2014 36th annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE, pp 3957–3960
28. Yu Z, Chen H, You J, Liu J, Wong H-S, Han G, Li L (2015) Adaptive fuzzy consensus clustering framework for clustering analysis of cancer data. *IEEE/ACM Trans Comput Biol Bioinform* 12(4):887–901
29. Li W, Zhang J, McKenna SJ (2015) Multiple instance cancer detection by boosting regularised trees. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 645–652
30. Azizi S, Imani F, Zhuang B, Tahmasebi A, Kwak JT, Xu S, Uniyal N, Turkbey B, Choyke P, Pinto P et al (2015) Ultrasound-based detection of prostate cancer using automatic feature selection with deep belief networks. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 70–77
31. Kandemir M, Wojek C, Hamprecht FA (2015) Cell event detection in phase-contrast microscopy sequences from few annotations. In: Medical image computing and computer-assisted intervention—MICCAI 2015. Springer, Berlin, pp 316–323
32. Xing F, Su H, Neltner J, Yang L (2014) Automatic ki-67 counting using robust cell detection and online dictionary learning. *IEEE Trans Biomed Eng* 61(3):859–870
33. Fu H, Qiu G, Shu J, Ilyas M (2014) A novel polar space random field model for the detection of glandular structures. *IEEE Trans Med Imaging* 33(3):764–776
34. Lou X, Schiegg M, Hamprecht FA (2014) Active structured learning for cell tracking: algorithm, framework, and usability. *IEEE Trans Med Imaging* 33(4):849–860
35. Litjens G, Debats O, Barentsz J, Karssemeijer N, Huisman H (2014) Computer-aided detection of prostate cancer in mri. *IEEE Trans Med Imaging* 33(5):1083–1092
36. Lo C-M, Chen R-T, Chang Y-C, Yang Y-W, Hung M-J, Huang C-S, Chang R-F (2014) Multi-dimensional tumor detection in automated whole breast ultrasound using topographic watershed. *IEEE Trans Med Imaging* 33(7):1503–1511
37. Cameron A, Khalvati F, Haider M, Wong A (2015) A quantitative radiomics approach for prostate cancer detection, maps: a quantitative radiomics approach for prostate cancer detection
38. Cruz-Roa AA, Ovalle JEA, Madabhushi A, Osorio FAG (2013) A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In: Medical image computing and computer-assisted intervention—MICCAI 2013. Springer, Berlin, pp 403–410
39. Marasco WA, Phan SH, Krutzsch H, Showell HJ, Feltner DE, Nairn R, Becker EL, Ward PA (1984) Purification and identification of formyl-methionyl-leucyl-phenylalanine as the major peptide neutrophil chemotactic factor produced by escherichia coli. *J Biol Chem* 259(9):5430–5439

40. Mualla F, Schöll S, Sommerfeldt B, Maier A, Steidl S, Buchholz R, Hornegger J (2014) Unsupervised unstained cell detection by sift keypoint clustering and self-labeling algorithm. In: Medical image computing and computer-assisted intervention—MICCAI 2014. Springer, Berlin, pp 377–384
41. Xing F, Su H, Yang L (2013) An integrated framework for automatic ki-67 scoring in pancreatic neuroendocrine tumor. In: Medical image computing and computer-assisted intervention—MICCAI 2013. Springer, Berlin, pp 436–443
42. Chakraborty A, Roy-Chowdhury AK (2015) Context aware spatio-temporal cell tracking in densely packed multilayer tissues. *Med Image Anal* 19(1):149–163
43. Veta M, Van Diest PJ, Willems SM, Wang H, Madabhushi A, Cruz-Roa A, Gonzalez F, Larsen AB, Vestergaard JS, Dahl AB et al (2015) Assessment of algorithms for mitosis detection in breast cancer histopathology images. *Med Image Anal* 20(1):237–248
44. Ali S, Lewis J, Madabhushi A (2013) Spatially aware cell cluster (spaccl) graphs: predicting outcome in oropharyngeal p16+ tumors. In: Medical image computing and computer-assisted intervention—MICCAI 2013. Springer, Berlin, pp 412–419
45. Pan H, Xu Z, Huang J (2015) An effective approach for robust lung cancer cell detection. In: International workshop on patch-based techniques in medical imaging. Springer, Berlin, pp 87–94
46. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the ACM international conference on multimedia. ACM, pp 675–678
47. Xu Z, Huang J (2015) Efficient lung cancer cell detection with deep convolution neural network. In: International workshop on patch-based techniques in medical imaging. Springer, Berlin, pp 79–86
48. Li H, Zhao R, Wang X (2014) Highly efficient forward and backward propagation of convolutional neural networks for pixelwise classification. [arXiv:1412.4526](https://arxiv.org/abs/1412.4526)
49. National Lung Screening Trial Research Team et al. (2011) The national lung screening trial: overview and study design. *Radiology*
50. Arteta C, Lempitsky V, Noble JA, Zisserman A (2012) Learning to detect cells using non-overlapping extremal regions. In: Medical image computing and computer-assisted intervention—MICCAI 2012. Springer, Berlin, pp 348–356
51. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324
52. Xu Z, Huang J (2016) Detecting 10,000 cells in one second. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, pp 676–684
53. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
54. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
55. Wang S, Yao J, Xu Z, Huang J (2016) Subtype cell detection with an accelerated deep convolution neural network. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, pp 640–648
56. Recht B, Re C, Wright S, Niu F (2011) Hogwild: a lock-free approach to parallelizing stochastic gradient descent. In: Advances in neural information processing systems, pp 693–701
57. Han S, Pool J, Tran J, Dally W (2015) Learning both weights and connections for efficient neural network. In: Advances in neural information processing systems, pp 1135–1143

Chapter 10

Fully Convolutional Networks in Medical Imaging: Applications to Image Enhancement and Recognition

Christian F. Baumgartner, Ozan Oktay and Daniel Rueckert

Abstract Convolutional neural networks (CNNs) are hierarchical models that have immense representational capacity and have been successfully applied to computer vision problems including object localisation, classification and super-resolution. A particular example of CNN models, known as fully convolutional network (FCN), has been shown to offer improved computational efficiency and representation learning capabilities due to simpler model parametrisation and spatial consistency of extracted features. In this chapter, we demonstrate the power and applicability of this particular model on two medical imaging tasks, image enhancement via super-resolution and image recognition. In both examples, experimental results show that FCN models can significantly outperform traditional learning-based approaches while achieving real-time performance. Additionally, we demonstrate that the proposed image classification FCN model can be used in organ localisation task as well without requiring additional training data.

10.1 Introduction

With the advent of efficient parallel computing power in form of GPUs, deep neural networks and in particular deep convolutional neural networks (CNNs), have recently gained considerable research interest in computer vision and medical image analysis. In addition to the improved computational power, the availability of large image datasets and annotations have allowed deep neural network models to achieve state-of-the-art performance in many different tasks.

C.F. Baumgartner (✉) · O. Oktay · D. Rueckert
Biomedical Image Analysis Group, Department of Computing, Imperial College London,
180 Queen's Gate, London SW7 2AZ, UK
e-mail: c.baumgartner@imperial.ac.uk

O. Oktay
e-mail: o.oktay13@imperial.ac.uk

D. Rueckert
e-mail: d.rueckert@imperial.ac.uk

A notable scientific breakthrough in the computer vision domain was made by Krizhevsky et al. [1] in the ImageNet LSVRC-2010 classification challenge. In that work, the authors used a feature extractor consisting of a number of convolutional layers and a classifier based on the multiple fully connected layers to address the challenge of image recognition. Many of the new benchmarks that followed used a similar architecture with a trend towards deeper model architectures and smaller kernel sizes such as VGG-net [2] and residual networks [3].

The considerable success of CNNs can be attributed to two main reasons: (I) Its scalable feature learning architecture that tunes model parameters for a given particular task and relies very little on feature-engineering or prior knowledge, and (II) the end-to-end model training strategy which allows to simultaneously optimise all components of a particular image-processing pipeline. These advantages over traditional learning-based algorithms have recently also led to a wide adoption of CNNs in the medical image analysis domain. In the last 3 years, there has already been a considerable amount of work, which demonstrated the successful applications of CNNs in medical imaging problems. In many of these applications, the CNN-based approaches have been shown to outperform many traditional methods based on the hand-engineered analysis frameworks and image features. A few well-known applications include semantic segmentation in microscopy [4] and cardiac images [5], anatomical landmark localisation [6, 7], spatial image alignment [8] and abnormality classification [9].

In this chapter, we discuss two specific applications of CNNs in the medical imaging domain in detail, namely an approach for cardiac image enhancement via super resolution (SR) and a technique for real-time image recognition and organ localisation. The former approach addresses the clinical difficulties that arise when imaging cardiac volumes using stacked 2D MR acquisitions, which suffer from low resolution (LR) in the through plane direction. The discussed SR model accurately predicts a high-resolution (HR) isotropic volume from a given LR clinical image. The second technique discussed in this chapter aims to improve fetal mid-pregnancy abnormality scans by providing robust real-time detection of a number of standard views in a stream of 2D ultrasound (US) data. Moreover, the method provides a localisation of the fetal target anatomy via bounding boxes in frames containing such views, without needing bounding box annotations during training.

Both of the presented methods employ fully convolutional network architectures (FCN), that is, network architectures which consist solely of convolutional and max-pooling layers and forgo the fully connected layers traditionally used in the classification step. Generally, the use of fully connected layers restricts the model to fixed image sizes which must be decided during training. In order to obtain predictions for larger, rectangular input images during test time, typically the network is evaluated multiple times for overlapping patches of the training image size, and it usually hinders real-time performance of the algorithm. FCNs can be used to calculate the output

to arbitrary image sizes much more efficiently in a single forward pass. The second attribute that makes FCNs particularly suitable for the techniques discussed in this chapter, is that they allow to design networks which retain a spatial correspondence between the input image and the network output. Therefore, whenever classification speed or spatial correspondence of the input and output are desired, it can be beneficial to employ FCNs. Recent applications of such networks include semantic segmentation [10], natural image super-resolution [11, 12], and object localisation [13].

The chapter is structured as follows: Sect. 10.2 presents an overview of related work in medical image super-resolution, as well as examples of the use of residual learning and multi-input CNN models on cardiac image super-resolution problem. Section 10.3 describes the automatic scan plane detection approach. Finally, the advantages of CNNs on these particular problems are discussed and future research directions are given.

10.2 Image Super-Resolution

In this section, we present an example use of fully convolution neural networks for the medical image super resolution task. The presented neural network model predicts a 3D high-resolution cardiac MR image from a given input low-resolution stack of 2D image slices, which is experimentally shown to be useful for subsequent image analysis and qualitative assessment.

10.2.1 Motivation

3D magnetic resonance (MR) imaging with near isotropic resolution provides a good visualisation of cardiac morphology, and enables accurate assessment of cardiovascular physiology. However, 3D MR sequences usually require long breath-hold and repetition times, which leads to scan times that are infeasible in clinical routine, and 2D multi-slice imaging is used instead. Due to limitations on signal-to-noise ratio (SNR), the acquired slices are usually thick compared to the in-plane resolution and thus negatively affect the visualisation of anatomy and hamper further analysis. Attempts to improve image resolution are typically carried out either during the acquisition stage (sparse k-space filling) or retrospectively through super-resolution (SR) of single/multiple image acquisitions.

Related Work: Most of the SR methods recover the missing information through the examples observed in training images, which are used as a prior to link low and high-resolution (LR–HR) image patches. Single image SR methods, based on the way they utilise training data, fall into two categories: non-parametric and parametric. The former aims to recover HR patches from LR ones via a cooccurrence prior between

the target image and external training data. Atlas-based approaches such as the patchmatch method [14] and non-local means-based single image SR [15] methods are two examples of this category. These approaches are computationally demanding as the candidate patches have to be searched in the training dataset to find the most suitable HR candidate. Instead, compact and generative models can be learned from the training data to define the mapping between LR and HR patches. Parametric generative models, such as coupled-dictionary learning-based approaches, have been proposed to upscale MR brain [16] and cardiac [17] images. These methods benefit from sparsity constraint to express the link between LR and HR. Similarly, random forest-based nonlinear regressors have been proposed to predict HR patches from LR data and have been successfully applied on diffusion tensor images [18]. Recently, CNN models [11, 12] have been put forward to replace the inference step as they have enough capacity to perform complex nonlinear regression tasks. Even by using a shallow network composed of a few layers, these models [12] achieved superior results over other state-of-the-art SR methods.

Contributions: In the work presented here, we extend the SR-CNN proposed by [11, 12] with an improved layer design and training objective function, and show its application to cardiac MR images. In particular, the proposed approach simplifies the LR–HR mapping problem through residual learning and allows training a deeper network to achieve improved performance. Additionally, the new model can be considered more data-adaptive since the initial upscaling is performed by learning a deconvolution layer instead of a fixed kernel [12]. More importantly, a multi-input image extension of the SR-CNN model is proposed and exploited to achieve a better SR image quality. By making use of multiple images acquired from different slice directions one can further improve and constrain the HR image reconstruction. Similar multi-image SR approaches have been proposed in [19, 20] to synthesise HR cardiac images; however, these approaches did not make use of available large training datasets to learn the appearance of anatomical structures in HR. Compared to the state-of-the-art image SR approaches [12, 14], the proposed method shows improved performance in terms of peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) [21]. Additionally, the experimental results show that cardiac image segmentation can benefit from SR-CNN as the segmentations generated from super-resolved images are shown to be similar to the manual segmentations on HR images in terms of volume measures and surface distances. Lastly, it is shown that cardiac motion tracking results can be improved using SR-CNN as it visualises the basal and apical parts of the myocardium more clearly compared to the conventional interpolation methods (see Fig. 10.1).

10.2.2 Methodology

The SR image generation is formulated as an inverse problem that recovers the high-dimensional data through the MR image acquisition model [22], which has

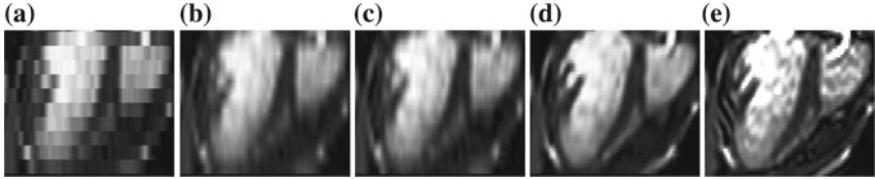


Fig. 10.1 The low-resolution image **a** is upscaled using linear **b** and cubic spline **c** interpolations, and the proposed method **d** which shows a high correlation with the ground-truth high-resolution image **e** shown on the *rightmost*

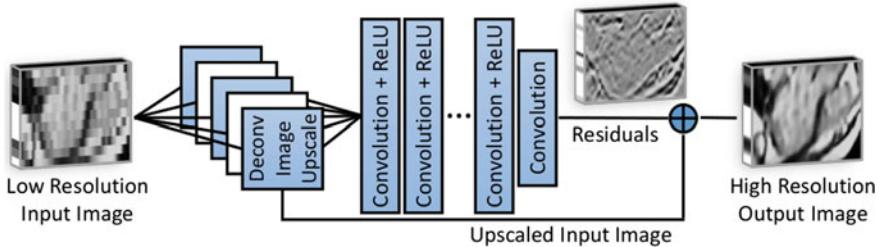


Fig. 10.2 The proposed single image super resolution network model

been the starting point of approaches in [14, 17, 19]. The model links the HR volume $\mathbf{y} \in \mathbb{R}^M$ to the low-dimensional observation $\mathbf{x} \in \mathbb{R}^N$ ($N \ll M$) through the application of a series of operators as: $\mathbf{x} = \mathbf{DBSM}\mathbf{y} + \boldsymbol{\eta}$ where M defines the spatial displacements caused due to respiratory and cardiac motion, S is the slice selection operator, \mathbf{B} is a point-spread function (PSF) used to blur the selected slice, \mathbf{D} is a decimation operator, and $\boldsymbol{\eta}$ is the Rician noise model. The solution to this inverse problem estimates a conditional distribution $p(\mathbf{y}|\mathbf{x})$ that minimises the cost function Ψ defined by \mathbf{y} and its estimate $\Phi(\mathbf{x}, \boldsymbol{\Theta})$ obtained from LR input data. The estimate is obtained through a CNN parameterised by $\boldsymbol{\Theta}$ that models the distribution $p(\mathbf{y}|\mathbf{x})$ via a collection of hidden variables. For the smooth ℓ_1 norm case, the loss function is defined as $\min_{\boldsymbol{\Theta}} \sum_i \Psi_{\ell_1}(\Phi(\mathbf{x}_i, \boldsymbol{\Theta}) - \mathbf{y}_i)$, where $\Psi_{\ell_1}(r) = \{0.5 r^2 \text{ if } |r| < 1, |r| - 0.5 \text{ otherwise}\}$ and $(\mathbf{x}_i, \mathbf{y}_i)$ denote the training samples. The next section describes the proposed CNN model.

Single Image Network: The proposed model, shown in Fig. 10.2, is formed by concatenating a series of convolutional layers (Conv) and rectified linear units (ReLU) [23] to estimate the nonlinear mapping Φ , as proposed in [12] to upscale natural images. The intermediate feature maps $h_j^{(n)}$ at layer n are computed through Conv kernels (hidden units) w_{kj}^n as $\max \left(0, \sum_{k=1}^K h_k^{(n-1)} * w_{kj}^n \right) = h_j^n$ where $*$ is the convolution operator. As suggested by [2], in order to obtain better nonlinear estimations, the proposed architecture uses small Conv kernels ($3 \times 3 \times 3$) and a large number of Conv+ReLU layers. Such approach allows training of a deeper network. Different to the models proposed in [11, 12], we include an initial upscaling operation

within the model as a deconvolution layer (Deconv) ($\mathbf{x} \uparrow U) * w_j = h_j^0$ where \uparrow is a zero-padding upscaling operator and $U = M/N$ is the upscaling factor. In this way, upsampling filters can be optimised for SR applications by training the network in an end-to-end manner. This improves the image signal quality in image regions closer to the boundaries. Instead of learning to synthesise a HR image, the CNN model is trained to predict the residuals between the LR input data and HR ground-truth information. These residuals are later summed up with the linearly upscaled input image (output of Deconv layer) to reconstruct the output HR image. In this way, a simplified regression function Φ is learned where mostly high-frequency signal components, such as edges and texture, are predicted (see Fig. 10.2). At training time, the correctness of reconstructed HR images is evaluated based on the $\Psi_{\ell_1}(.)$ function, and the model weights are updated by backpropagating the error defined by that function. In [24] the ℓ_1 norm was shown to be a better metric than the ℓ_2 norm for image restoration and SR problems. This is attributed to the fact that the weight updates are not dominated by the large prediction errors.

Multi-image Network: The single image model is extended to multi-input image SR by creating multiple input channels (MC) from given images which are resampled to the same spatial grid and visualise the same anatomy. In this way, the SR performance is enhanced by merging multiple image stacks, e.g. long-axis (LAX) and short axis (SAX) stacks, acquired from different imaging planes into a single SR volume. However, when only a few slices are acquired, a mask or distance map is required as input to the network to identify the missing information. Additionally, the number of parameters is supposed to be increased so that the model can learn to extract in image regions where the masks are defined, which increases the training time accordingly. For this reason, a Siamese network [25] is proposed as a third model (see Fig. 10.3) for comparison purposes, which was used in similar problems such as shape recognition from multiple images [26]. The first stage of the network resamples, the input images into a fixed HR spatial grid. In the second stage, the same type of image features are extracted from each channel which are sharing the same filter weights. In the final stage, the features are pooled and passed to another Conv network to reconstruct

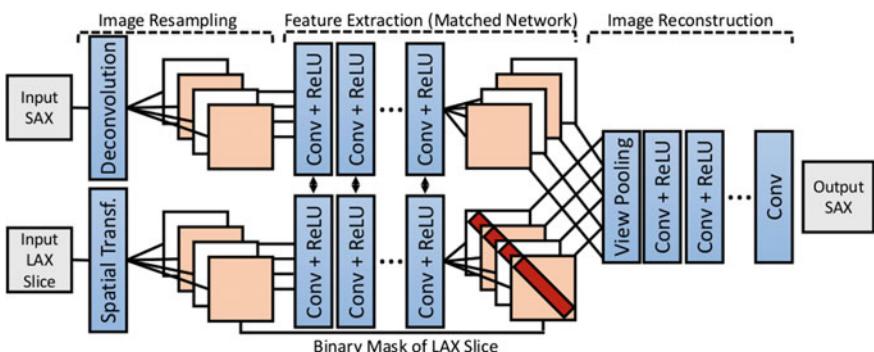


Fig. 10.3 The proposed Siamese multi-image super resolution network model

the output HR image. The view pooling layer averages the corresponding features from all channels over the areas where the images are overlapping. The proposed models are initially pre-trained with small number of layers to better initialise the final deeper network training, which improves the network performance [11].

10.2.3 Results

The models are evaluated on end-diastolic frames of cine cardiac MR images acquired from 1233 healthy adult subjects. The images are upscaled in the direction orthogonal to the SAX plane. The proposed method is compared against linear, cubic spline, and multi-atlas patchmatch (MAPM) [14] upscaling methods in four different experiments: image quality assessment for (a–b) single and multi-input cases, (c) left-ventricle (LV) segmentation, (d) LV motion tracking.

Experimental Details: In the first experiment, an image dataset containing 1080 3D SAX cardiac volumes with voxel size $1.25 \times 1.25 \times 2.00$ mm, is randomly split into two subsets and used for single image model training (930) and testing (150). The images are intensity normalised and cropped around the heart. Synthetic LR images are generated using the acquisition model given in Sect. 10.2.2, which are resampled to a fixed resolution $1.25 \times 1.25 \times 10.00$ mm. The PSF is set to be a Gaussian kernel with a full-width at half-maximum equal to the slice thickness [22]. For the LR/HR pairs, multiple acquisitions could be used as well, but an unbalanced bias would be introduced near sharp edges due to spatial misalignments. For the evaluation of multi-input models, a separate clinical dataset of 153 image pairs of LAX cardiac image slices and SAX image stacks are used, of which 10 pairs are split for evaluation. Spatial misalignment between SAX and LAX images are corrected using image registration [27]. For the single/multi-image model, seven consecutive Conv layers are used after the upscaling layer. In the Siamese model, the channels are merged after the fourth Conv layer.

Image Quality Assessment: The upscaled images are compared with the ground-truth HR 3D volumes in terms of PSNR and SSIM [21]. The latter measure assesses the correlation of local structures and is less sensitive to image noise. The results

Table 10.1 Quantitative comparison of different image upsampling methods

Exp (a)	PSNR (dB)	SSIM	# Filters/atlasses
Linear	20.83 ± 1.10	0.70 ± 0.03	–
CSpline	22.38 ± 1.13	0.73 ± 0.03	–
MAPM	22.75 ± 1.22	0.73 ± 0.03	350
sh-CNN	23.67 ± 1.18	0.74 ± 0.02	64, 64, 32, 1
CNN	24.12 ± 1.18	0.76 ± 0.02	64, 64, 32, 16, 8, 4, 1
de-CNN	24.45 ± 1.20	0.77 ± 0.02	64, 64, 32, 16, 8, 4, 1

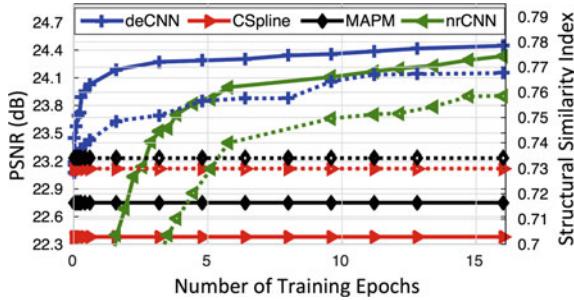


Fig. 10.4 Results on the testing data, PSNR (*solid*) and SSIM (*dashed*)

Table 10.2 Image quality results obtained with three different models: single image de-CNN, Siamese, and multi-channel (MC) that uses multiple input images

Exp (b)	de-CNN(SAX)	Siamese(SAX/4CH)	MC(SAX/4CH)	MC(SAX/2/4CH)
PSNR (dB)	24.76 ± 0.48	25.13 ± 0.48	25.15 ± 0.47	25.26 ± 0.37
SSIM	0.807 ± 0.009	0.814 ± 0.013	0.814 ± 0.012	0.818 ± 0.012
p - values	0.005	0.016	0.017	-

in Table 10.1 show that learning the initial upscaling kernels (de-CNN) can improve ($p = 0.007$) the quality of generated HR image compared to convolution only network (CNN) using the same number of trainable parameters. Additionally, the performance of 7-layer network is compared against the 4-layer shallow network from [12] (sh-CNN). Addition of extra Conv layers to the 7-layer model is found to be ineffective due to increased training time and negligible performance improvement. In Fig. 10.4, we see that CNN-based methods can learn better HR synthesis models even after a small number of training epochs. On the same figure, it can be seen that the model without the residual learning (nrCNN) underperforms and requires a large number of training iterations.

Multi-input Model: In the second experiment, we show that the single image SR model can be enhanced by providing additional information from two and four chamber (2/4CH) LAX images. The results given in Table 10.2 show that by including LAX information in the model, a modest improvement in image visual quality can be achieved. The improvement is mostly observed in image regions closer to areas, where the SAX-LAX slices overlap, as can be seen in Fig. 10.5a–d. Also, the results show that the multi-channel (MC) model performs slightly better than Siamese model as it is given more degrees-of-freedom, whereas the latter is more practical as it trains faster and requires fewer trainable parameters.

Segmentation Evaluation: As a subsequent image analysis, 18 SAX SR images are segmented using a state-of-the-art multi-atlas method [28]. The SR images generated from clinical 2D stack data with different upscaling methods are automatically segmented and those segmentations are compared with the manual annotations per-

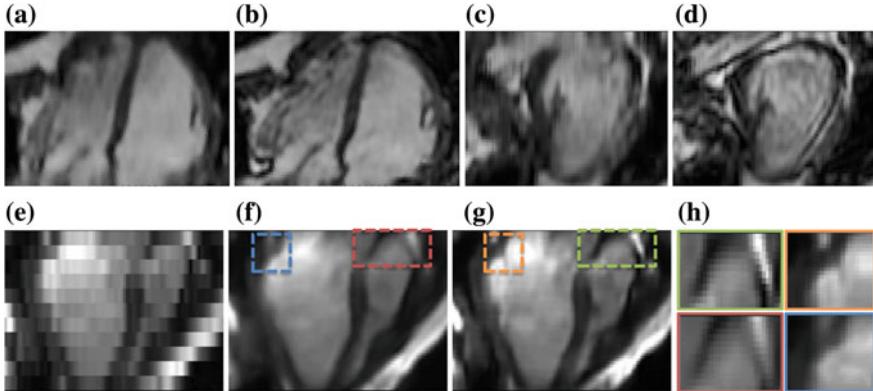


Fig. 10.5 The LV is better visualised by using multi-input images (**b, d**) compared to single image SR (**a, c**). Also, the proposed method (**g**) performs better than MAPM [14] (**f**) in areas where uncommon shapes are over-smoothed by atlases

Table 10.3 Segmentation results for different upsampling methods, CSpline ($p = 0.007$) and MAPM ($p = 0.009$). They are compared in terms of mean and Hausdorff distances (MYO) and LV cavity volume differences (w.r.t. manual annotations)

		Linear	CSpline	MAPM	de-CNN	High Res
Exp (c)	LV Vol Diff (ml)	11.72 ± 6.96	10.80 ± 6.46	9.55 ± 5.42	9.09 ± 5.36	8.24 ± 5.47
	Mean Dist (mm)	1.49 ± 0.30	1.45 ± 0.29	1.40 ± 0.29	1.38 ± 0.29	1.38 ± 0.28
	Haus Dist (mm)	7.74 ± 1.73	7.29 ± 1.63	6.83 ± 1.61	6.67 ± 1.77	6.70 ± 1.85

formed on ground-truth HR 3D images. Additionally, the HR images are segmented with the same method to show the lower error bound. The quality of segmentations are evaluated based on the LV cavity volume measure and surface-to-surface distances for myocardium (MYO). The results in Table 10.3 show that CNN upscaled images can produce segmentation results similar to the ones obtained from HR images. The main result difference between the SR methods is observed in image areas where thin and detailed boundaries are observed (e.g. apex). As can be seen in Fig. 10.5e–h, the MAPM over-smooths areas closer to image boundaries. Inference of the proposed model is not as computationally demanding as brute-force searching (MAPM), which requires hours for a single image, whereas SR-CNN can be executed in 6.8 s on GPU or 5.8 mins CPU on average per image. The shorter runtime makes the SR methods more applicable to subsequent analysis, as they can replace the standard interpolation methods.

Motion Tracking: The clinical applications of SR can be extended to MYO tracking as it can benefit from SR as a pre-processing stage to better highlight the ventricle

boundaries. End-diastolic MYO segmentations are propagated to end-systolic (ES) phase using B-Spline FFD registrations [29]. ES meshes generated with CNN and linear upscaling methods are compared with tracking results obtained with 10 3D-SAX HR images based on Hausdorff distance. The proposed SR method produces tracking results (4.73 ± 1.03 mm) more accurate ($p = 0.01$) than the linear interpolation (5.50 ± 1.08 mm). We observe that the images upscaled with the CNN model follow the apical boundaries more accurately, which is shown in the supplementary material: www.doc.ic.ac.uk/~oo2113/publication/miccai16/.

10.2.4 Discussion and Conclusion

The results show that the proposed SR approach outperforms conventional upscaling methods both in terms of image quality metrics and subsequent image analysis accuracy. Also, it is computationally efficient and can be applied to image analysis tasks such as segmentation and tracking. The experiments show that these applications can benefit from SR images since 2D stack image analysis with SR-CNN can achieve similar quantitative results as the analysis on isotropic volumes without requiring long acquisition time. We also show that the proposed model can be easily extended to multiple image input scenarios to obtain better SR results. SR-CNN's applicability is not only limited to cardiac images but to other anatomical structures as well. In the proposed approach, inter-slice and stack spatial misalignments due to motion are handled using a registration method. However, we observe that large slice misplacements can degrade SR accuracy. Future research will focus on that aspect of the problem.

10.3 Scan Plane Detection

In this section, we show how a simple convolution neural network can be used for very accurate detection of fetal standard scan planes in real-time 2D US data.

10.3.1 Motivation

Abnormal fetal development is a leading cause of perinatal mortality in both industrialised and developing countries [30]. Although many countries have introduced fetal screening programmes based on the mid-pregnancy ultrasound scans at around 20 weeks of gestational age, detection rates remain relatively low. For example, it is estimated that in the UK approximately 26% of fetal anomalies are not detected during pregnancy [31]. Detection rates have also been reported to vary considerably across different institutions [32] which suggests that, at least in part, differences in

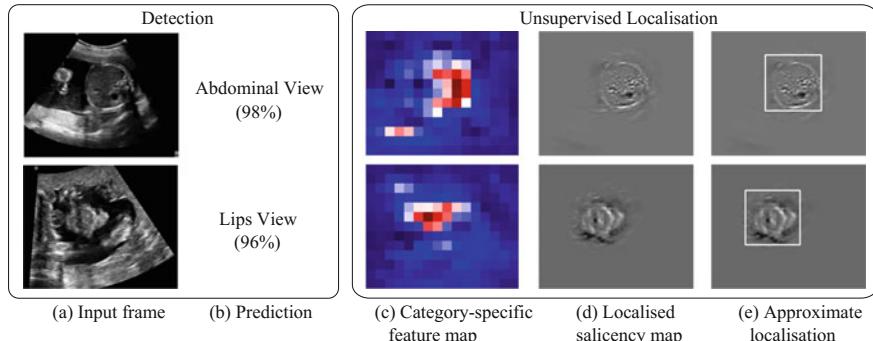


Fig. 10.6 Overview of the proposed framework for two standard view examples. Given a video frame (a) the trained convolutional neural network provides a prediction and confidence value (b). By design, each classifier output has a corresponding low-resolution feature map (c). Backpropagating the error from the most active feature neurons results in a saliency map (d). A bounding box can be derived using thresholding (e)

training may be responsible for this variability. Moreover, according to the WHO, it is likely that worldwide many US scans are carried out by individuals with little or no formal training [30].

Biometric measurements and identification of abnormalities are performed on a number of standardised 2D US view planes acquired at different locations in the fetal body. In the UK, guidelines for selecting these planes are defined in [33]. Standard scan planes are often hard to localise even for experienced sonographers and have been shown to suffer from low reproducibility and large operator bias [34]. Thus, a system automating or aiding with this step could have significant clinical impact particularly in geographic regions where few highly skilled sonographers are available. It is also an essential step for further processing such as automated measurements or automated detection of anomalies.

In this section, we show how a real-time system which can automatically detect 12 commonly acquired standard scan planes in clinical free-hand 2D US data can be implemented using a relatively simple convolutional neural network. We demonstrate the detection framework for (1) real-time annotations of US data to assist sonographers, and (2) for the retrospective retrieval of standard scan planes from recordings of the full examination. Furthermore, we extend this architecture to obtain saliency maps highlighting the part of the image that provides the highest contribution to a prediction (see Fig. 10.6). Such saliency maps provide a localisation of the respective fetal anatomy and can be used as starting point for further automatic processing. This localisation step is unsupervised and does not require ground-truth bounding box annotations during training.

Related Work: Standard scan plane classification of 7 planes was proposed for a large fetal *image* database [35]. This differs significantly from the present work since in that scenario it is already known that every image is in fact a standard plane whilst

in video data the majority of frames does not show standard planes. A number of papers have proposed methods to detect fetal anatomy in videos of fetal 2D US sweeps (e.g. [36]). In those works, the authors were aiming at detecting the presence of fetal structures such as the skull, heart or abdomen rather specific standardised scan planes.

Automated fetal standard scan plane detection has been demonstrated for 1–3 standard planes in 2D fetal US sweeps [37–39]. US sweeps are acquired by moving the US probe from the cervix upwards in one continuous motion [38]. However, not all standard views required to determine the fetus’ health status are adequately visualised using a sweep protocol. For example, visualising the femur or the lips normally requires careful manual scan plane selection. Furthermore, data obtained using the sweep protocol are typically only 2–5 s long and consist of fewer than 50 frames [38]. To the best of our knowledge, fetal standard scan plane detection has never been performed on true free-hand US data which typically consist of 10,000+ frames. Moreover, none of related works were demonstrated to run in real-time, typically requiring multiple seconds per frame.

Note that the majority of the related works followed a traditional machine learning approach in which a set of fixed features (e.g. Haar-like features) are extracted from the data and are then used to train a machine learning algorithm such as random forests. The only exceptions are [37, 38], who also employed convolutional neural networks and performed an end-to-end training.

10.3.2 Materials and Methods

Data and Pre-processing: Our dataset consists of 1003 2D US scans of consented volunteers with gestational ages between 18 and 22 weeks which have been acquired by a team of expert sonographers using GE Voluson E8 systems. For each scan, a screen capture video of the entire procedure was recorded. Additionally, for each case the sonographers saved multiple “freeze frames” of a number of standard views. A large fraction of these frames have been annotated allowing us to infer the correct ground-truth (GT) label. All video frames and images were downsampled to a size of 225×273 pixels.

We considered 12 standard scan planes based on the guidelines in [33]. In particular, we selected the following: two brain views at the level of the ventricles (Vt.) and the cerebellum (Cb.), the standard abdominal view, the transverse kidney view, the coronal lip, the median profile, and the femur and sagittal spine views. We also included four commonly acquired cardiac views: the left and right ventricular outflow tracts (LVOT and RVOT), the three vessel view (3VV) and the four chamber view (4CH). Examples, of each category are shown in Fig. 10.7.

Since this work aims at *detection* of scan planes in real video data, rather than categorisation we need a robust way to model the challenging background class, i.e. the “not a standard scan plane” category. Due to the large variations in appearance that free-hand ultrasound a large amount of images is required to adequately represent

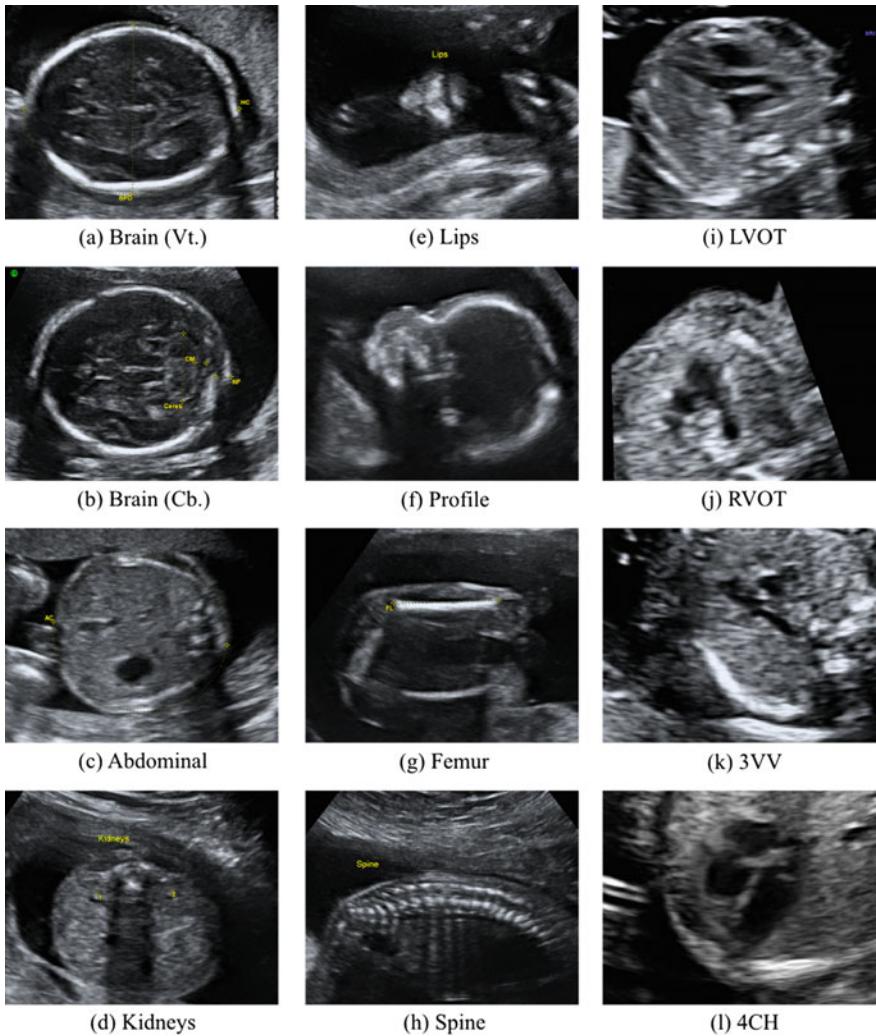


Fig. 10.7 Overview of the modelled standard scan planes

this class. Therefore, we additionally sampled 50 random frames from each video, that is in total 50150 images, to model the background.

Network Architecture: The architecture of the CNN discussed in this section is summarised in Fig. 10.8. The architecture is inspired by the AlexNet [1], but is designed with lower complexity for optimal speed. However, in contrast to the AlexNet and following recent advances in computer vision, we opted for a fully convolutional network architecture which replaces traditional fully connected layers with convolution layers using a 1×1 kernel [40, 41]. In the final convolutional layer (C6), the

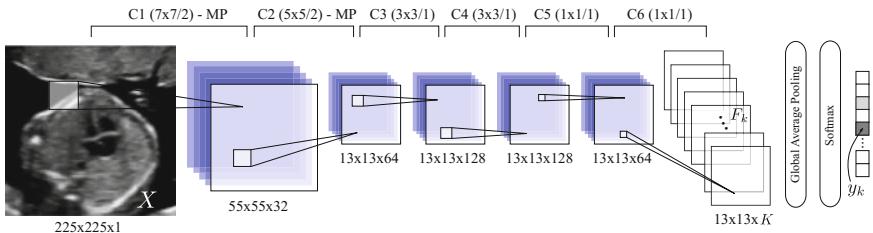


Fig. 10.8 Overview of the discussed network architecture. The size and stride of the convolutional kernels are indicated at the *top* (notation: *kernel size/stride*). Max-pooling steps are indicated by *MP* (2×2 bins, stride of 2). The activation functions of all convolutions except *C6* are rectified nonlinear units (ReLUs). *C6* is followed by a global average pooling step. The sizes at the *bottom* of each image/feature map refer to the training phase and will be slightly larger during inference due to larger input images

input is reduced to K 13×13 feature maps F_k , where K is the number of classes. Each of these feature maps is then averaged to obtain the input to the final Softmax layer.

Note that independent of the input image size the final output will always be mapped to the same dimension due to the average pool layer. This allows training on square patches of images as is common for convolutional neural networks [1] and beneficial for data augmentation [2, 3]. Networks with a densely connected classification layer need to be evaluated multiple times for rectangular input images, see, for example [38]. In a fully convolutional architecture, rectangular images can be evaluated in a single forward pass which allows for significantly more efficient operation and is crucial for achieving real-time performance.

A key aspect of the network architecture presented in this section is that we enforce a one-to-one correspondence between each feature map F_k and the respective prediction y_k . Since each neuron in the feature maps F_k has a receptive field in the original image, during training, the neurons will learn to activate only if an object of class k is in that field. This allows to interpret F_k as a spatially encoded confidence map for class k [40]. In this paper, we take advantage of this fact to generate localised saliency maps as described below.

Training: We split the dataset into a test set containing 20% of the subjects and a training set containing 80%. We use 10% of the training data as validation set to monitor the training progress. In total, we model 12 standard view planes, plus one background class resulting in $K = 13$ categories.

We train the model in an end-to-end fashion using mini-batch gradient descent, using the categorical cross-entropy cost function and the Adam optimiser proposed in [42]. In order to prevent overfitting we add 50% dropout after the C5 and C6 layers. To account for the significant class imbalance introduced by the background category, we create stratified mini-batches with even class-sampling.

We sample random square patches of size 225×225 from the input images and, additionally, augment each batch by a factor of 5 by transforming them with a small

random rotation and flips along the vertical axis. Taking random square subimages allows to introduce more variation to the augmented batches compared to training on the full field of view. This helps to reduce the overfitting of the network. We train the network for 50 epochs and choose the network parameters with the lowest error on the validation set.

Frame Annotation and Retrospective Retrieval: After training we feed the network with video frames containing the full field of view (225×273 pixels) of the input videos. This results in larger category-specific feature maps of 13×16 . The prediction y_k and confidence c_k of each frame are given by the prediction with the highest probability and the probability itself.

For retrospective frame retrieval, for each subject we calculate and record the confidence for each class over the entire duration of an input video. Subsequently, we retrieve the frame with the highest confidence for each class.

Saliency Maps and Unsupervised Localisation: After obtaining the category y_k of the current frame X from a forward pass through the network, we can examine the feature map F_k (i.e. the output of the C6 layer) corresponding to the predicted category k . Two examples of feature maps are shown in Fig. 10.6c. The F_k could already be used to make an approximate estimate of the location of the respective anatomy similar to [41].

Here, instead of using the feature maps directly, we present a method to obtain localised saliency with the resolution of the original input images. For each neuron $F_k^{(p,q)}$ at the location p, q in the feature map it is possible calculate how much each original input pixel $X^{(i,j)}$ contributed to the activation of this neuron. This corresponds to calculating the partial derivatives

$$S_k^{(i,j)} = \frac{\partial F_k^{(p,q)}}{\partial X^{(i,j)}},$$

which can be solved efficiently using an additional backwards pass through the network. Reference [43] proposed a method for performing this backpropagation in a *guided* manner by allowing only error signals which contribute to an increase of the activations in the higher layers (i.e. layers closer to the network output) to backpropagate. In particular, the error is only backpropagated through each neuron's ReLU unit if the input to the neuron x , as well as the error in the higher layer δ_ℓ are positive. That is, the backpropagated error $\delta_{\ell-1}$ of each neuron is given by $\delta_{\ell-1} = \delta_\ell \sigma(x) \sigma(\delta_\ell)$, where $\sigma(\cdot)$ is the unit step function.

In contrast to [43] who backpropagated from the final output, in this work we take advantage of the spatial encoding in the category-specific feature maps and only backpropagate the errors for the 10% most active feature map neurons, i.e. the spatial locations where the fetal anatomy is predicted. The resulting saliency maps are significantly more localised compared to [43] (see Fig. 10.9).

These saliency maps can be used as starting point for various image analysis tasks such as automated segmentation or measurements. Here, we demonstrate how they can be used for approximate localisation using basic image processing. We blur the

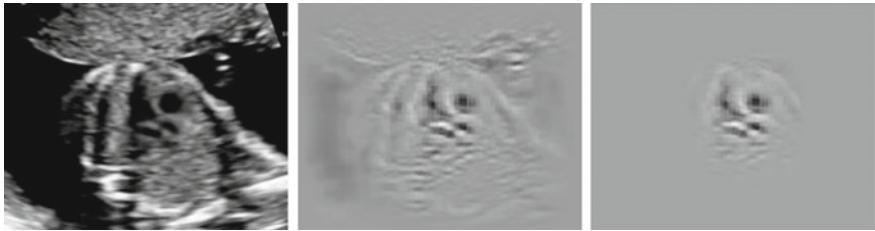


Fig. 10.9 Saliency maps obtained from the input frame (LVOT class) shown on the *left*. The *middle* map was obtained using guided backpropagation from the average pool layer output [43]. The map on the *right* was obtained using the discussed method

absolute value image of a saliency map $|S_k|$ using a 25×25 Gaussian kernel and apply a thresholding using Otsu's method [44]. Finally, we compute the minimum bounding box of the components in the threshold image.

10.3.3 Experiments and Results

Real-Time Frame Annotation: We evaluated the ability of the method to detect standard frames by classifying the test data including the randomly sampled background class. We report the achieved precision (pc) and recall (rc) scores in Table 10.4. The lowest scores were obtained for cardiac views, which are also the most difficult to scan for expert sonographers. This fact is reflected in the low detection rates for serious cardiac anomalies (e.g. only 35% in the UK).

Chen et al. [37] have recently reported pc/rc scores of 0.75/0.75 for the abdominal standard view, and 0.77/0.61 for the 4CH view in US sweep data. We obtained comparable values for the 4CH view and considerably better values for the abdominal view. However, with 12 modelled standard planes and free-hand US data the problem here is significantly more complex. Using, a Nvidia Tesla K80 graphics processing unit (GPU) we were able to classify 113 frames per second (FPS) on average, which significantly exceeds the recording rate of the ultrasound machine of 25 FPS. An

Table 10.4 Detection scores: precision $pc = TP/(TP + FP)$ and recall $rc = TP/(TP + FN)$ for the classification of the modelled scan planes and the background class

View	pc	rc	View	pc	rc	View	pc	rc
Brain (Vt.)	0.96	0.90	Lips	0.85	0.88	LVOT	0.63	0.63
Brain (Cb.)	0.92	0.94	Profile	0.71	0.82	RVOT	0.40	0.46
Abdominal	0.85	0.80	Femur	0.79	0.93	3VV	0.46	0.60
Kidneys	0.64	0.87	Spine	0.51	0.99	4CH	0.61	0.74
Background	0.96	0.93						

Table 10.5 Retrieval accuracy: percentage of correctly retrieved frames for each standard view for all 201 test subjects

View	(%)	View	(%)	View	(%)
Brain (Vt.)	0.95	Lips	0.77	LVOT	0.73
Brain (Cb.)	0.89	Profile	0.76	RVOT	0.70
Abdominal	0.79	Femur	0.75	3VV	0.66
Kidneys	0.87	Spine	0.77	4CH	0.78

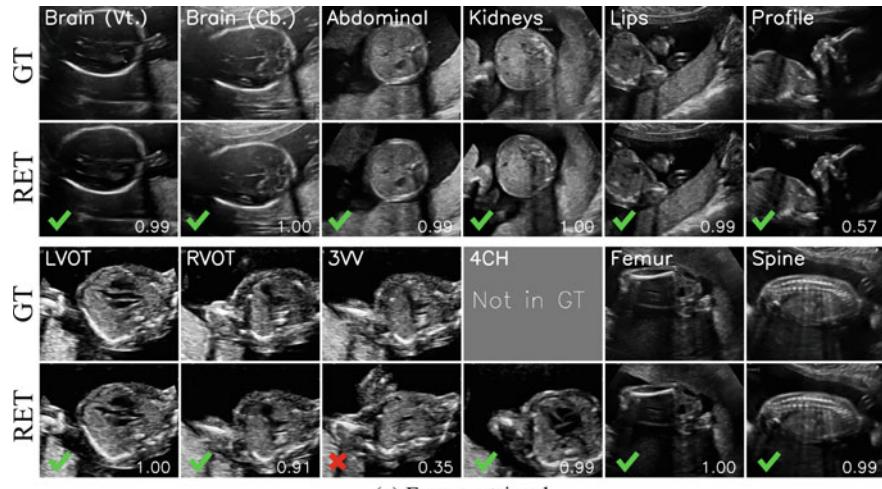
example of a annotated video can be viewed at <https://www.youtube.com/watch?v=w4tfvRFGhvE>.

Retrospective Frame Retrieval: We retrieved the standard views from videos of all test subjects and manually evaluated whether the retrieved frames corresponded to the annotated GT frames for each category. Several cases did not have GTs for all views because they were not manually included by the sonographer in the original scan. For those cases, we did not evaluate the retrieved frame. The results are summarised in Table 10.5. We show examples of the retrieved frames for two volunteers in Fig. 10.10. Note that in many cases the retrieved planes match the expert GT almost exactly. Moreover, some planes which were not annotated by the experts were nevertheless found correctly. As before, most cardiac views achieved lower scores compared to other views.

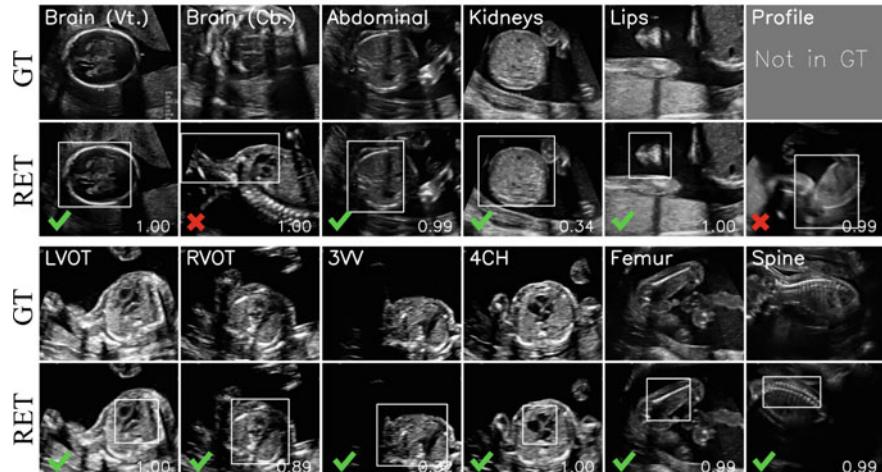
Localisation: We show results for the approximate localisation of the respective fetal anatomy in the retrieved frames for one representative case in Fig. 10.10b and in the supplemental video. We found that performing the localisation reduced the frame rate to 39 FPS on average.

10.3.4 Discussion and Conclusion

In this section, we have introduced a system for the automatic detection of 12 fetal standard scan planes from real clinical fetal US scans. The employed fully CNN architecture allowed for robust real-time inference. Furthermore, we have shown a method to obtain localised saliency maps by combining the information in category-specific feature maps with a guided backpropagation step. To the best of our knowledge, this approach is the first to model a large number of fetal standard views from a substantial population of free-hand US scans. We have shown that the method can be used to robustly annotate US data with classification scores exceeding values reported in related work for some standard planes, but in a much more challenging scenario. A system based on the presented approach could potentially be used to assist or train inexperienced sonographers. We have also shown how the framework can be used to retrieve standard scan planes retrospectively. In this manner, relevant key frames could be extracted from a video acquired by an inexperienced operator and sent for



(a) Frame retrieval



(b) Frame retrieval and localisation

Fig. 10.10 Retrieved standard frames (*RET*) and *GT* frames annotated and saved by expert sonographers for two volunteers. Correctly retrieved and incorrectly retrieved frames are denoted with a green check mark or red cross, respectively. Frames with no *GT* annotation are indicated. The confidence is shown in the lower right of each image. The frames in **b** additionally contain the results of the localisation method (boxes)

further analysis to an expert. We have also demonstrated how the localised saliency maps can be used to extract an approximate bounding box of the fetal anatomy. This is an important stepping stone for further, more specialised image processing.

10.4 Discussion and Conclusion

In this chapter, two methods were presented which outperform the respective states-of-the-art by employing fully convolutional networks and an end-to-end optimisation.

The network described in Sect. 10.2 was designed to learn to predict high-resolution images from low-resolution versions of the same images. To this end, a low-resolution input image was first upsampled to the higher resolution using a deconvolution layer and then the residual image information was estimated in a number of convolutional layers to reconstruct the output volume. Importantly, this entire pipeline was learned in an end-to-end fashion allowing to optimise all the steps simultaneously. The experimental results showed that FCNs can learn high level representations that enable more accurate regression performance in synthesising the high-resolution volumes. In addition to the standard single input SR-CNN model, a multi-input model extension was presented to make use of additional intensity information provided by scans acquired from different imaging planes.

In Sect. 10.3, a different fully convolutional network was described which can accurately detect 12 fetal standard scan planes in complex free-hand ultrasound data. In contrast to Sect. 10.2 the output in this case was a single prediction rather than an image. This could have been achieved using a standard architecture with a fully connected classification layer. However, constructing the network exclusively with convolutions allowed to obtain a class-specific feature map which can indicate an approximate location of the structure of interest in the image. From those maps, very accurate category-specific saliency maps were obtained which can be used to provide operator an explanation behind the prediction and also for unsupervised localisation of the target structures.

A number of related scan plane detection works rely on extracting a fixed set of features from the data and training a classifier such as AdaBoost or random forests on them [35, 39]. However, similar to the work described in Sect. 10.2, optimising the feature extraction and classification simultaneously generally leads to better results than optimising each step separately and more recent-related works has also adopted this strategy [37, 38].

In conclusion, fully convolutional networks offer an attractive solution when one is interested in an output that bears some spatial correspondence to the input image. On the one hand, this may be the case if the output is an image as well. In this chapter, the work on super resolution fell into this category. Another example of the same nature which was not covered in this chapter is image segmentation [4, 10, 45] in which case the output is a segmentation mask. On the other hand, a spatial correspondence can be “forced” by making the network output artificially large and aggregating those feature maps using average pooling. This produces spatial confidence maps in an unsupervised fashion. In either case, such networks can be optimised end-to-end as a whole which has been shown to generally produce very good results in medical imaging and computer vision tasks alike.

References

1. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems 25 (NIPS 2012). pp 1097–1105
2. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
3. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. arXiv preprint [arXiv:1512.03385](https://arxiv.org/abs/1512.03385)
4. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, pp 234–241
5. Tran PV (2016) A fully convolutional neural network for cardiac segmentation in short-axis MRI. arXiv preprint [arXiv:1604.00494](https://arxiv.org/abs/1604.00494)
6. Ghisu FC, Georgescu B, Mansi T, Neumann D, Hornegger J, Comaniciu D (2016) An artificial agent for anatomical landmark detection in medical images. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, pp 229–237
7. Payer C, Štern D, Bischof H, Urschler M (2016) Regressing heatmaps for multiple landmark localization using CNNs. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, pp 230–238
8. Miao S, Wang ZJ, Liao R (2016) A CNN regression approach for real-time 2D/3D registration. IEEE Trans Med Imaging 35(5):1352–1363
9. Jamaludin A, Kadir T, Zisserman A (2016) SpineNet: automatically pinpointing classification evidence in spinal MRIs. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, pp 166–175
10. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 3431–3440
11. Dong C, Deng Y, Change Loy C, Tang X (2015) Compression artifacts reduction by a deep convolutional network. In: IEEE CVPR. pp 576–584
12. Dong C, Loy CC, He K, Tang X (2016) Image super-resolution using deep convolutional networks. IEEE PAMI 38(2):295–307
13. Redmon J, Divvala S, Girshick R, Farhadi A (2015) You only look once: unified, real-time object detection. arXiv preprint [arXiv:1506.02640](https://arxiv.org/abs/1506.02640)
14. Shi W, Caballero J, Ledig C, Zhuang X, Bai W, Bhatia K, de Marvao A, Dawes T, ORegan D, Rueckert D (2013) Cardiac image super-resolution with global correspondence using multi-atlas patchmatch. In: MICCAI. pp 9–16
15. Manjón JV, Coupé P, Buades A, Fonov V, Collins DL, Robles M (2010) Non-local MRI upsampling. MedIA 14(6):784–792
16. Rueda A, Malpica N, Romero E (2013) Single-image super-resolution of brain MR images using overcomplete dictionaries. MedIA 17(1):113–132
17. Bhatia KK, Price AN, Shi W, Rueckert D (2014) Super-resolution reconstruction of cardiac MRI using coupled dictionary learning. In: IEEE ISBI. pp 947–950
18. Alexander DC, Zikic D, Zhang J, Zhang H, Criminisi A (2014) Image quality transfer via random forest regression: applications in diffusion MRI. In: MICCAI. Springer, Berlin, pp 225–232
19. Odille F, Bustin A, Chen B, Vuissoz PA, Felblinger J (2015) Motion-corrected, super-resolution reconstruction for high-resolution 3D cardiac cine MRI. In: MICCAI. Springer, Berlin, pp 435–442
20. Plenge E, Poot D, Niessen W, Meijering E (2013) Super-resolution reconstruction using cross-scale self-similarity in multi-slice MRI. In: MICCAI. pp 123–130
21. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE TIP 13(4):600–612

22. Greenspan H (2009) Super-resolution in medical imaging. *Comput J* 52(1):43–63
23. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324
24. Zhao H, Gallo O, Frosio I, Kautz J (2015) Is L2 a good loss function for neural networks for image processing? *arXiv preprint arXiv:1511.08861*
25. Bromley J, Guyon I, LeCun Y, Sckinger E, Shah R (1994) Signature verification using a Siamese time delay neural network. In: *NIPS*. pp 737–744
26. Su H, Maji S, Kalogerakis E, Learned-Miller E (2015) Multi-view convolutional neural networks for 3D shape recognition. In: *IEEE CVPR*. pp 945–953
27. Lötjönen J, Pollari M, Lauerma K (2004) Correction of movement artifacts from 4-D cardiac short-and long-axis MR data. In: *MICCAI*. Springer, Berlin, pp 405–412
28. Bai W, Shi W, O'Regan DP, Tong T, Wang H, Jamil-Copley S, Peters NS, Rueckert D (2013) A probabilistic patch-based label fusion model for multi-atlas segmentation with registration refinement: application to cardiac MR images. *IEEE TMI* 32(7):1302–1315
29. Rueckert D, Sonoda LI, Hayes C, Hill DL, Leach MO, Hawkes DJ (1999) Nonrigid registration using free-form deformations: application to breast MR images. *IEEE TMI* 18(8):712–721
30. Salomon L, Alfirevic Z, Bergella V, Bilardo C, Leung KY, Malinger G, Munoz H et al (2011) Practice guidelines for performance of the routine mid-trimester fetal ultrasound scan. *Ultrasound Obst Gyn* 37(1):116–126
31. Kurinczuk J, Hollowell J, Boyd P, Oakley L, Brocklehurst P, Gray R (2010) The contribution of congenital anomalies to infant mortality. National Perinatal Epidemiology Unit, University of Oxford
32. Bull C et al (1999) Current and potential impact of fetal diagnosis on prevalence and spectrum of serious congenital heart disease at term in the UK. *The Lancet* 354(9186):1242–1247
33. NHS screening programmes (2015) Fetal anomalie screen programme handbook. pp 28–35
34. Chan L, Fung T, Leung T, Sahota D, Lau T (2009) Volumetric (3D) imaging reduces inter-and intraobserver variation of fetal biometry measurements. *Ultrasound Obst Gyn* 33(4):447–452
35. Yaquib M, Kelly B, Papageorgiou A, Noble J (2015) Guided random forests for identification of key fetal anatomy and image categorization in ultrasound scans. In: *Proceedings of the MICCAI*. Springer, Berlin, pp 687–694
36. Maraci M, Napolitano R, Papageorgiou A, Noble J (2014) Searching for structures of interest in an ultrasound video sequence. In: *Proceedings of the MLMI*. pp 133–140
37. Chen H, Dou Q, Ni D, Cheng JZ, Qin J, Li S, Heng PA (2015) Automatic fetal ultrasound standard plane detection using knowledge transferred recurrent neural networks. In: *Proceedings of the MICCAI*. Springer, Berlin, pp 507–514
38. Chen H, Ni D, Qin J, Li S, Yang X, Wang T, Heng P (2015) Standard plane localization in fetal ultrasound via domain transferred deep neural networks. *IEEE J Biomed Health Inform* 19(5):1627–1636
39. Ni D, Yang X, Chen X, Chin CT, Chen S, Heng PA, Li S, Qin J, Wang T (2014) Standard plane localization in ultrasound by radial component model and selective search. *Ultrasound Med Biol* 40(11):2728–2742
40. Lin M, Chen Q, Yan S (2013) Network in network. *arXiv:1312.4400*
41. Oquab M, Bottou L, Laptev I, Sivic J (2015) Is object localization for free?-weakly-supervised learning with convolutional neural networks. In: *IEEE proceedings of the CVPR*. pp 685–694
42. Kingma D, Ba J (2014) Adam: a method for stochastic optimization. *arXiv:1412.6980*
43. Springenberg J, Dosovitskiy A, Brox T, Riedmiller M (2014) Striving for simplicity: the all convolutional net. *arXiv:1412.6806*
44. Otsu N (1975) A threshold selection method from gray-level histograms. *Automatica* 11(285–296):23–27
45. Kamnitsas K, Ledig C, Newcombe V, Simpson J, Kane A, Menon D, Rueckert D, Glocker B (2017) Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Med Image Anal* 36:61–78

Chapter 11

On the Necessity of Fine-Tuned Convolutional Neural Networks for Medical Imaging

Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst,
Christopher B. Kendall, Michael B. Gotway and Jianming Liang

Abstract This study aims to address two central questions. First, *are fine-tuned convolutional neural networks (CNNs) necessary for medical imaging applications?* In response, we considered four medical vision tasks from three different medical imaging modalities, and studied the necessity of fine-tuned CNNs under varying amounts of training data. Second, *to what extent the knowledge is to be transferred?* In response, we proposed a layer-wise fine-tuning scheme to examine how the extent or depth of fine-tuning contributes to the success of knowledge transfer. Our experiments consistently showed that the use of a pre-trained CNN with adequate fine-tuning outperformed or, in the worst case, performed as well as a CNN trained from scratch. The performance gap widened when reduced training sets were used for training and fine-tuning. Our results further revealed that the required level of fine-tuning

N. Tajbakhsh (✉) · J.Y. Shin · J. Liang

Department of Biomedical Informatics, Arizona State University,
13212 E. Shea Blvd., Scottsdale, AZ 85259, USA
e-mail: Nima.Tajbakhsh@asu.edu

J.Y. Shin

e-mail: Sejong@asu.edu

J. Liang

e-mail: Jianming.Liang@asu.edu

S.R. Gurudu

Division of Gastroenterology and Hepatology, Mayo Clinic,
13400 E. Shea Blvd., Scottsdale, AZ 85259, USA
e-mail: Gurudu.Suryakanth@mayo.edu

R. Todd Hurst · C.B. Kendall

Division of Cardiovascular Diseases, Mayo Clinic,
13400 E. Shea Blvd., Scottsdale, AZ 85259, USA
e-mail: Hurst.R@mayo.edu

C.B. Kendall

e-mail: Kendall.Christopher@mayo.edu

M.B. Gotway

Department of Radiology, Mayo Clinic,
13400 E. Shea Blvd., Scottsdale, AZ 85259, USA
e-mail: Gotway.Michael@mayo.edu

differed from one application to another, suggesting that neither shallow tuning nor deep tuning may be the optimal choice for a particular application. Layer-wise fine-tuning may offer a practical way to reach the best performance for the application at hand based on the amount of available data. We conclude that knowledge transfer from natural images is necessary and that the level of tuning should be chosen experimentally.

11.1 Introduction

Convolutional Neural Networks (CNNs) have recently shown tremendous success for various computer vision tasks in natural images. However, training a deep CNN from scratch (or full training) requires a large amount of labeled training data—a requirement that may be difficult to meet in the medical imaging domain where expert annotation is expensive and the diseases (e.g., lesions) are rare in the datasets. A promising alternative to training from scratch is to fine-tune a pre-trained network [1–3]. The idea is to transfer the knowledge from a source domain with a tremendous set of labeled images to a target domain where only limited labeled data are available. Fine-tuned CNNs have recently shown promising performance for medical imaging applications [4–8]; however, their potentials have not been systematically studied yet.

In this study, we address the following questions in the context of CNNs and medical imaging: *Is knowledge transfer necessary for high performance? If so, what level of knowledge transfer is needed?* To answer the above central questions, we propose a layer-wise fine-tuning scheme and study knowledge transfer to the following 4 medical imaging applications (see Fig. 11.1): (1) polyp detection in colonoscopy videos, (2) image quality assessment in colonoscopy videos, (3) pulmonary embolism detection in computed tomography (CT) images, and (4) intima-media boundary segmentation in ultrasound images. These applications are chosen to cover different imaging modality systems (i.e., CT, ultrasound, and optical endoscopy) and the most common medical vision tasks (i.e., lesion detection, organ segmentation, and image classification). Our experiments demonstrated that knowledge transfer was necessary for achieving high performance systems particularly given limited training data. We also discovered that layer-wise fine-tuning is a practical way to reach the best performance for the application at hand based on the amount of available training data.

11.2 Related Works

As summarized in Table 11.1, CNNs have been extensively used for solving a variety of medical vision tasks. However, literature on knowledge transfer from natural images to the medical imaging domain is not as significant. The related research

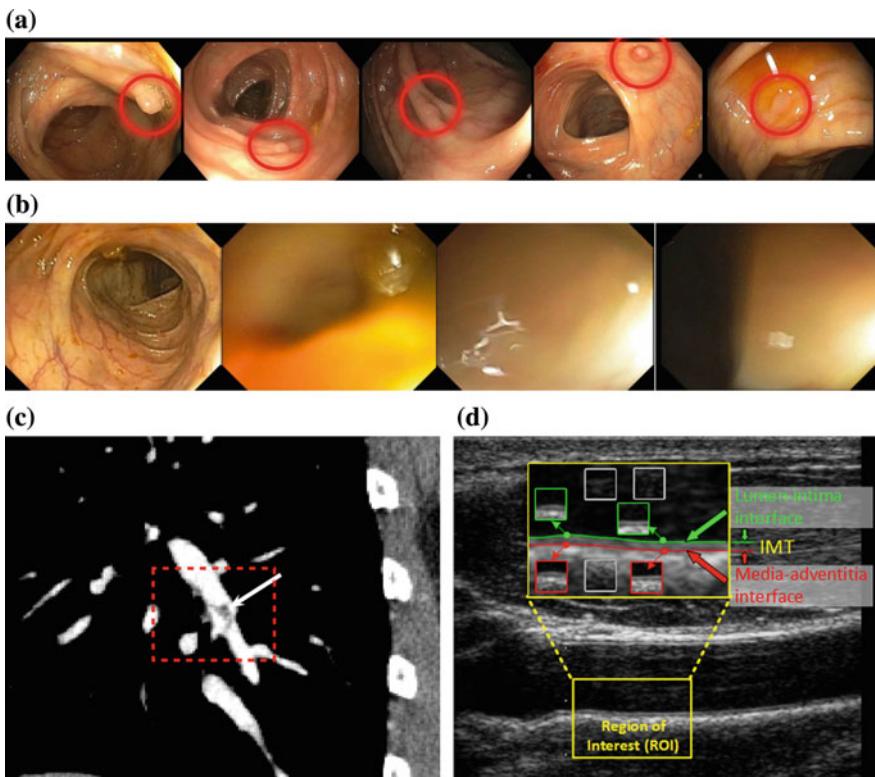


Fig. 11.1 Studied applications. **a** polyp detection in colonoscopy videos. Polyps are highlighted with the *red circles*. **b** Frame classification for quality monitoring of colonoscopy procedures. The very left image is an informative image but the rest are non-informative. **c** Pulmonary embolism (PE) detection in CT datasets. Dark PE is marked by the *arrow* in the bright vessel. **d** lumen-intima interface (*green* boundary) and media-adventitia interface (*red* boundary) segmentation in ultrasound images

on knowledge transfer can be categorized into two groups. The first group [4, 5, 7] consists of works wherein a pre-trained CNN is used as a feature generator: the CNN features (outputs of a certain layer) are extracted and then used to train a new pattern classifier. The second group consists of works that fine-tune a pre-trained CNN. This has been accomplished by means of shallow fine-tuning of a pre-trained CNN where only the last layer is trained [9] or by means of deep tuning where all convolutional layers in a pre-trained CNN are trained [6, 8, 10]. Shallow tuning requires only limited medical imaging data but may not achieve the desired performance. On the other hand, deep tuning may better adapt the pre-trained CNN to the application at hand but may require more medical imaging data. It would be interesting to study how different levels of fine-tuning contribute to knowledge transfer in various medical imaging applications.

Table 11.1 A brief review of the CNN-related research in medical imaging

Reference	Task
	Focal pattern detection
[7, 11–13]	Nodule detection
[14–16]	Polyp detection
[17]	Pulmonary embolism detection
[18]	Lymph node detection
[19, 20]	Cell detection
	Segmentation
[21]	Cartilage segmentation
[22]	Pancreas segmentation
[23–25]	Brain (tumor) segmentation
[26]	Tissue segmentation
	Image classification and registration
[27]	MRI acquisition plane recognition
[5]	Chest pathology identification
[6]	Fetal ultrasound standard plane detection
[28]	Radiology image registration

11.3 Contributions

One of our key contributions is a systematic study of layer-wise knowledge transfer to 4 medical imaging applications with varying distances to natural images. The selected applications are from three different medical imaging modalities, involving image classification, object detection, and boundary segmentation. For each application, we further study the choice between fine-tuning and full training under different amount of training data. Our findings may add to the state of the art, where conclusions are solely based on one medical imaging application and are derived for only shallow or deep fine-tuning.

11.4 Applications and Results

For consistency and also ease of comparison between applications, we use the AlexNet architecture for all experiments in this study. For each application, we used a stratified training set by down-sampling the majority class. For training AlexNet from scratch, we used different learning rates ranging from 0.0001 to 0.01, and found out that a learning rate of 0.001 led to a proper convergence for all applications. For fine-tuning the pre-trained AlexNet, we used a learning rate of 0.01 for the last fully connected layer and a learning rate of 0.001 for the previous layers. To exclude a

Table 11.2 Learning parameters used for training and fine-tuning of AlexNet in our experiments. μ is the momentum, α is the learning rate of the weights in each convolutional layer, and γ determines the rate by which α decreases at the end of each epoch. Note that “Fine-tuned AlexNet:layer1-layer2” indicates that all the layers between and including these two layers undergo fine-tuning

CNNs	Parameters									
	μ	α_{conv1}	α_{conv2}	α_{conv3}	α_{conv4}	α_{conv5}	α_{fc6}	α_{fc7}	α_{fc8}	γ
Fine-tuned AlexNet:conv1-fc8	0.9	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.01	0.95
Fine-tuned AlexNet:conv2-fc8	0.9	0	0.001	0.001	0.001	0.001	0.001	0.001	0.01	0.95
Fine-tuned AlexNet:conv3-fc8	0.9	0	0	0.001	0.001	0.001	0.001	0.001	0.01	0.95
Fine-tuned AlexNet:conv4-fc8	0.9	0	0	0	0.001	0.001	0.001	0.001	0.01	0.95
Fine-tuned AlexNet:conv5-fc8	0.9	0	0	0	0	0.001	0.001	0.001	0.01	0.95
Fine-tuned AlexNet:fc6-fc8	0.9	0	0	0	0	0	0.001	0.001	0.01	0.95
Fine-tuned AlexNet:fc7-fc8	0.9	0	0	0	0	0	0	0.001	0.01	0.95
Fine-tuned AlexNet:only fc8	0.9	0	0	0	0	0	0	0	0.01	0.95
AlexNet scratch	0.9	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.95

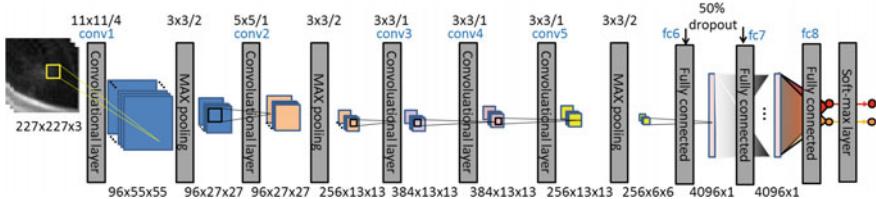


Fig. 11.2 Schematic overview of AlexNet used in our experiments

layer from the tuning process, we set the corresponding learning rate to 0. We used the notation “CNN (FT):layer1-layer3” to indicate that all the layers from Layer 1 to Layer 3 undergo fine-tuning. For the fine-tuning scenario, we employ the pre-trained AlexNet model provided in the Caffe library [29]. Table 11.2 summarizes the learning parameters used for the training and fine-tuning of AlexNet in all of our experiments (Fig. 11.2).

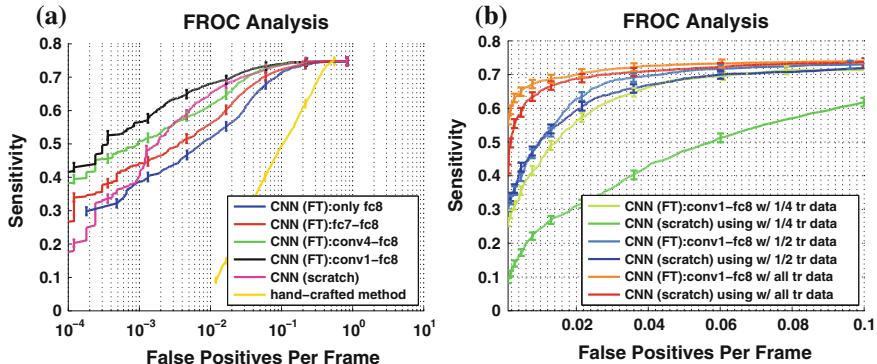


Fig. 11.3 FROC analysis for polyp detection. **a** Comparison between incremental fine-tuning, training from scratch, and a hand crafted approach [30]. **b** Effect of reduction in the training data on the performance of CNNs. Note that the sensitivity never reaches 100% due to false negatives of the candidate generator

11.4.1 Polyp Detection

Polyps are colon wall protrusions that are often missed during colonoscopy. From a vision perspective, polyp detection is difficult due to their large variations in shape, color, and size (see Fig. 11.1a). We use the polyp database released through the polyp detection challenge,¹ consisting of 40 short colonoscopy videos. The training videos contain 3800 frames with polyps and 15,100 frames without polyps, and the test videos contain 5,700 frames with polyps and 13,200 frames without polyps. Training and test patches were collected with data augmentation using the bounding boxes of polyp and non-polyp candidates generated by the system suggested in [30]. For training and fine-tuning the CNNs, we collect a stratified set of training patches by down-sampling the negative patches. For evaluation, we performed a free-response ROC (FROC) analysis.

Figure 11.3a compares the FROC curves for polyp detection. To avoid clutter in the figure, we have shown only a subset of representative FROC curves. As seen, the handcrafted approach [30] is significantly outperformed by all CNN-based scenarios ($p < 0.05$). This result is probably because the handcrafted approach used only geometric information to remove false positive candidates. For fine-tuning, we obtained the lowest performance with (FT:only fc8), but observed incremental performance improvement as we included more convolutional layers in the fine-tuning process. Also, as seen in Fig. 11.3a, fine-tuning the last few convolutional layers was sufficient to outperform AlexNet trained from scratch in low false positive rates (FT:conv4-fc8). This superiority becomes even more evident when the training set is reduced to 25% at poly-level. These findings suggest that deep fine-tuning is necessary for high-performance polyp detection.

¹<http://polyp.grand-challenge.org/>.

11.4.2 Pulmonary Embolism Detection

PEs are blood clots that block pulmonary arteries (see Fig. 11.1c). PEs are hard to diagnose, but CAD system have shown to be effective in reducing PE miss-rates. We used a private database consisting of 121 CTPA volumes with a total of 326 PEs. We first performed a candidate generation method [31] to obtain a set of PE candidates and then divided the candidates at the volume level into a training set covering 199 unique PEs and a test set with 127 PEs. The training and test patches were extracted from the candidate locations with data augmentation according to a novel image representation suggested in [17]. For evaluation, we performed an FROC analysis.

Figure 11.4a shows the representative FROC curves for PE detection. The most notable increase in performance was observed after updating the fully connected layers (CNN (FT):fc6-fc8). Fine-tuning the remaining convolutional layers led to marginal improvements although their accumulation resulted in a significant performance gain (CNN (FT):conv1-fc8). Therefore, deep fine-tuning is necessary to achieve the best knowledge transfer. We also observed that the deeply fine-tuned CNN performed on a par with the CNN trained from scratch, but neither of them outperformed the meticulously designed handcrafted approach. We found it interesting that end-to-end learning machines could learn such a sophisticated set of handcrafted features with minimal engineering effort. By reducing training data at volume level, as seen in Fig. 11.4b, we observed significant performance degradation for the CNN trained from scratch and to less extent for the deeply tuned CNN. This highlights the importance of large training sets for effective training and fine-tuning of CNNs.

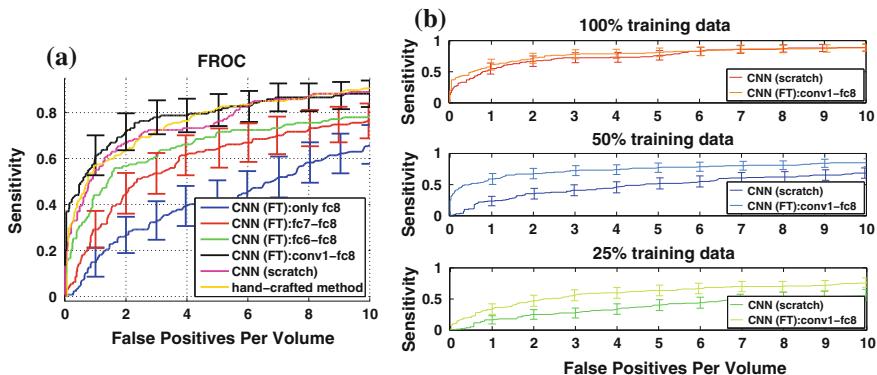


Fig. 11.4 FROC analysis for pulmonary embolism detection. **a** Comparison between incremental fine-tuning, training from scratch, and a handcrafted approach [31]. **b** Effect of reduction in the training data on the performance of CNNs

11.4.3 Colonoscopy Frame Classification

Typically, a colonoscopy video contains a large number of non-informative images, which are not suitable for inspecting the colon or performing therapeutic actions. The larger the fraction of non-informative images in a video, the lower the quality of colon visualization, and thus the lower the quality of colonoscopy. Therefore, one way to assess the quality of colonoscopy is to monitor the quality of images captured during the procedures. Technically, image quality assessment at colonoscopy can be viewed as an image classification task whereby an input image is labeled as either *informative* or *non-informative*. Figure 11.1b shows an example of informative frame and 3 examples of non-informative frames.

To develop our image classification system, we use a balanced dataset of 4,000 frames from six entire-length colonoscopy videos. Each frame is labeled as informative or non-informative. We divide the frames at the video-level into training and test sets, each containing approximately 2000 colonoscopy frames. For data augmentation, we extract 200 sub-images of size 227×227 pixels from random locations in each 500×350 colonoscopy frame, resulting in a stratified training set with approximately 40,000 sub-images. The extracted patches were labeled according to the images they were selected from. During the test stage, the probability of each frame being informative is computed as the average probabilities assigned to its randomly cropped sub-images. We used an ROC analysis for performance evaluation.

Figure 11.5a shows ROC curves for colonoscopy frame classification. We compared the performance curves at 3 operating points corresponding to 10, 15, and 20% false positive rates. We observed that all CNN-based scenarios significantly outperformed the handcrafted approach in at least one of the above 3 operating points. We also observed that fine-tuning the pre-trained CNN halfway through the network (FT:conv4-fc8 and FT:conv5-fc8) not only significantly outperformed shallow tuning but also was superior to a deeply fine-tuned CNN (FT:conv1-fc8) at 10

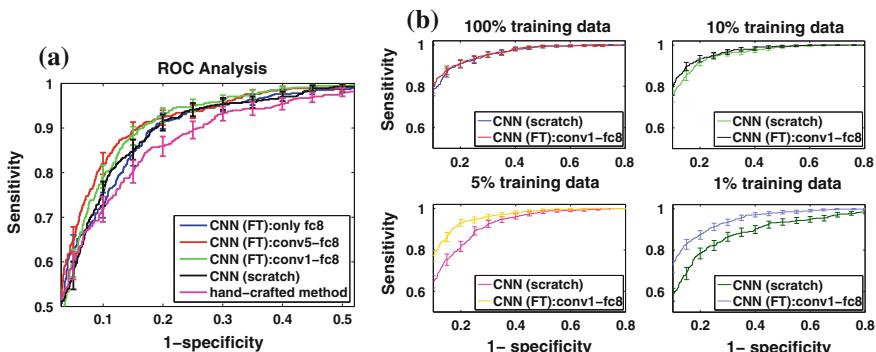


Fig. 11.5 ROC analysis for image quality assessment. **a** Comparison between incremental fine-tuning, training from scratch, and a hand-crafted approach [32]. **b** Effect of reduction in the training data on the performance of CNNs

and 15% false positive rates. Therefore, deep tuning was not as effective as intermediate tuning. This was probably because the kernels learned in the early layers of the CNN were suitable for image quality assessment and thus their fine-tuning was unnecessary. Furthermore, while the CNN trained from scratch outperformed the pre-trained CNN with shallow fine-tuning (FT:only fc8), it was outperformed by the pre-trained CNN with a moderate level of fine-tuning (FT:conv5-fc8). Therefore, the fine-tuning scheme was superior to the full training scheme from scratch. Figure 11.5b compares the deeply fine-tuned CNN and the CNN trained from scratch using reduced training sets. With 10% of the original training set, both models showed insignificant performance degradation; however, further reduction in the size of the training set substantially degraded the performance of fully trained CNNs and, to a largely less extent, the performance of deeply fine-tuned CNNs. The robustness of the deeply tuned CNN to sample size can be attributed to the similarity between ImageNet and the colonoscopy frames in our database. Specifically, both databases use high-resolution images and share similar low-level image information.

11.4.4 *Intima-Media Boundary Segmentation*

Carotid intima-media thickness (IMT) has proven to be valuable for predicting the risk of cardiovascular diseases. The IMT is defined as the average distance between the lumen-intima and media-adventitia interfaces in a region of interest (ROI) (Fig. 11.1d). The IMT measurement is performed by a sonographer who manually traces the lumen-intima and media-adventitia interfaces. This, however, is a time-consuming and tedious task. An automatic image segmentation method can accelerate the CIMT measurements.

Automatic segmentation of lumen-intima interface (LII) and media-adventitia interface (MAI) is essential for objective measurement of IMT, as shown in Fig. 11.1d. Technically, this segmentation problem can be viewed as a three-class classification task wherein the goal is to classify every pixel in the ROI into MAI, LIA, and background; and hence a CNN-based approach can be adopted. We used a database consisting of 276 ROIs from 23 patients with annotated LII and MAI. We divided the ROIs at the patient-level into a training set with 144 ROIs and a test set with 132 ROIs. For training and fine-tuning the CNNs, we extracted training patches without data augmentation from the background and annotated boundaries. This was because each ROI allowed us to extract a large number of image patches from a dense grid of points. During the test stage, the trained CNN was applied to each image in a convolutional fashion. We then found the maximum probability for MAI and LII in each column, yielding a 1-pixel thick boundary around each interface. To measure segmentation accuracy, we computed the distance between the annotated and segmented interfaces.

Figure 11.6 shows the box plots of segmentation error for each of the two interfaces. The whiskers are plotted according to Tukey's method. For fine-tuning, holding all the layers fixed, except the very last layer (fc8), resulted in the lowest performance.

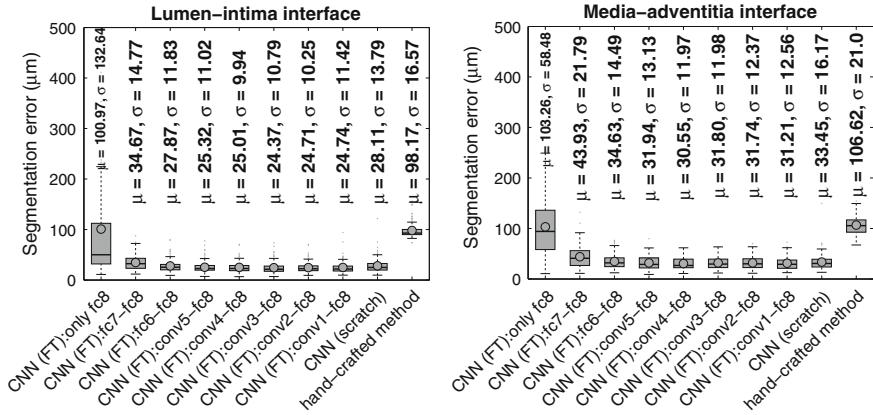


Fig. 11.6 Box plots of segmentation error for (*left*) the lumen–intima interface and (*right*) the media–adventitia interface

However, segmentation error decreased as more layers were fine-tuned. Specifically, the highest drop in segmentation error was observed when layers fc7 and fc6 underwent fine-tuning. For both interfaces, the lowest segmentation error was obtained using intermediate fine-tuning (FT:conv4-fc8) with a slight increase in segmentation error as more layer underwent fine-tuning. Therefore, deep fine-tuning was unnecessary.

11.5 Discussion

In this study, to ensure generalizability of our findings, we considered four common medical imaging problems from three different imaging modality systems. Specifically, we chose pulmonary embolism detection as representative of computer-aided lesion detection in 3D volumetric images, polyp detection as representative of computer-aided lesion detection in 2D videos, intima-media boundary segmentation as representative of machine-learning based medical image segmentation, and colonoscopy image quality assessment as representative of medical image classification. These applications are also different because they require solving problems at different image scales. For instance, while intima-media boundary segmentation and pulmonary embolism detection may require the examination of a small sub-region within the images, polyp detection and frame classification demand far larger receptive fields. Therefore, we believe that the chosen applications encompass a variety of applications relevant to the field of medical imaging.

We thoroughly investigated the potential for fine-tuned CNNs in the context of medical image analysis as an alternative to training deep CNNs from scratch. We performed our analyses using both large sets of training and reduced training datasets.

When using complete datasets, we observed that the shallow tuning of the pre-trained CNNs most often leads to a performance inferior to CNNs trained from scratch, whereas with deeper fine-tuning, we obtained performance superior to CNNs trained from scratch. The performance gap between fine-tuned CNNs and those trained from scratch widened when reducing the size of training sets, which led us to conclude that fine-tuned CNNs should always be the preferred option regardless of the size of training sets available.

We observed that the depth of fine-tuning is fundamental to achieving accurate image classifiers. For instance, while intermediate fine-tuning was sufficient to achieve the optimal performance for intima-media segmentation and colonoscopy frame classification, deep fine-tuning was essential to achieving the optimal performance for polyp detection and pulmonary embolism detection. This behavior was in contrast to the studied applications in the field of computer vision where shallow fine-tuning of pre-trained CNNs achieved the state-of-the-art performance. These observations can be explained by similarities between vision applications and differences between vision and medical applications.

In this study, we based our experiments on the AlexNet architecture. Alternatively, deeper architectures such as VGGNet [33] or GoogleNet [34], could have been utilized. We surmise that similar conclusion can be derived for these architectures. We should also emphasize that the objective of this study was not to achieve the highest performance for a number of different medical imaging tasks, but rather to study the effectiveness of knowledge transfer from natural to medical images in the presence and absence of sufficient labeled medical data. Therefore, AlexNet could be a reasonable architectural choice.

11.6 Conclusion

In this paper, we studied the necessity of fine-tuning and the effective level of knowledge transfer to 4 medical imaging applications. Our experiments demonstrated medical imaging applications were conducive to transfer learning and that fine-tuned CNNs were necessary to achieve high performance particularly with limited training datasets. We also showed that the desired level of fine-tuning differed from one application to another. While deeper levels of fine-tuning were suitable for polyp and PE detection, intermediate fine-tuning worked the best for interface segmentation and colonoscopy frame classification. Our findings led us to conclude that layer-wise fine-tuning is a practical way to reach the best performance based on the amount of available data.

Acknowledgements This research has been supported by NIH (R01HL128785) and ASU-Mayo seed program (pulmonary embolism); Mayo discovery translation program (carotid intima-media thickness); and ASU-Mayo seed program (colonoscopy). The content is solely the responsibility of the authors and does not necessarily represent the official views of NIH or ASU-Mayo funding programs.

References

1. Azizpour H, Razavian AS, Sullivan J, Maki A, Carlsson S (2014) From generic to specific deep representations for visual recognition. arXiv preprint [arXiv:1406.5774](https://arxiv.org/abs/1406.5774)
2. Penatti OA, Nogueira K, dos Santos JA (2015) Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 44–51
3. Razavian AS, Azizpour H, Sullivan J, Carlsson S (2014) CNN features off-the-shelf: an astounding baseline for recognition. In: 2014 IEEE conference on computer vision and pattern recognition workshops (CVPRW). IEEE, pp 512–519
4. Arevalo J, Gonzalez F, Ramos-Pollan R, Oliveira J, Guevara Lopez M (2015) Convolutional neural networks for mammography mass lesion classification. In: 2015 37th Annual international conference of the IEEE engineering in medicine and biology society (EMBC), pp 797–800. doi:[10.1109/EMBC.2015.7318482](https://doi.org/10.1109/EMBC.2015.7318482)
5. Bar Y, Diamant I, Wolf L, Greenspan H (2015) Deep learning with non-medical training used for chest pathology identification. In: SPIE medical imaging. International Society for Optics and Photonics, pp 94,140V–94,140V
6. Chen H, Dou Q, Ni D, Cheng JZ, Qin J, Li S, Heng PA (2015) Automatic fetal ultrasound standard plane detection using knowledge transferred recurrent neural networks. In: Medical image computing and computer-assisted intervention–MICCAI 2015. Springer, Berlin, pp 507–514
7. van Ginneken B, Setio AA, Jacobs C, Ciompi F (2015) Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans. In: 2015 IEEE 12th International symposium on biomedical imaging (ISBI), pp 286–289
8. Shin HC, Lu L, Kim L, Seff A, Yao J, Summers RM (2015) Interleaved text/image deep mining on a very large-scale radiology database. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1090–1099
9. Carneiro G, Nascimento J, Bradley A (2015) Unregistered multiview mammogram analysis with pre-trained deep learning models. In: Navab N, Hornegger J, Wells WM, Frangi AF (eds) Medical image computing and computer-assisted intervention MICCAI 2015. Lecture notes in computer science, vol 9351. Springer International Publishing, Berlin, pp 652–660
10. Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans Med Imaging PP(99):1–1. doi:[10.1109/TMI.2016.2528162](https://doi.org/10.1109/TMI.2016.2528162)
11. Ciompi F, de Hoop B, van Riel SJ, Chung K, Scholten ET, Oudkerk M, de Jong PA, Prokop M, van Ginneken B (2015) Automatic classification of pulmonary peri-fissural nodules in computed tomography using an ensemble of 2D views and a convolutional neural network out-of-the-box. Med Image Anal 26(1):195–202
12. Hua KL, Hsu CH, Hidayati SC, Cheng WH, Chen YJ (2015) Computer-aided classification of lung nodules on computed tomography images via deep learning technique. Onco Targets Ther 8
13. Shen W, Zhou M, Yang F, Yang C, Tian J (2015) Multi-scale convolutional neural networks for lung nodule classification. In: Ourselin S, Alexander DC, Westin CF, Cardoso MJ (eds) Information processing in medical imaging. Lecture notes in computer science, vol 9123. Springer International Publishing, pp 588–599. doi:[10.1007/978-3-319-19992-4_46](https://doi.org/10.1007/978-3-319-19992-4_46)
14. Roth H, Lu L, Liu J, Yao J, Seff A, Cherry K, Kim L, Summers R (2015) Improving computer-aided detection using convolutional neural networks and random view aggregation. IEEE Trans Med Imaging
15. Tajbakhsh N, Gurudu SR, Liang J (2015) Automatic polyp detection in colonoscopy videos using an ensemble of convolutional neural networks. In: 2015 IEEE 12th international symposium on biomedical imaging (ISBI). IEEE, pp 79–83

16. Tajbakhsh N, Gurudu SR, Liang J (2015) A comprehensive computer-aided polyp detection system for colonoscopy videos. In: Information processing in medical imaging. Springer, Berlin, pp 327–338
17. Tajbakhsh N, Gotway MB, Liang J (2015) Computer-aided pulmonary embolism detection using a novel vessel-aligned multi-planar image representation and convolutional neural networks. In: Medical image computing and computer-assisted intervention MICCAI 2015
18. Roth H, Lu L, Seff A, Cherry K, Hoffman J, Wang S, Liu J, Turkbey E, Summers R (2014) A new 2.5D representation for lymph node detection using random sets of deep convolutional neural network observations. In: Golland P, Hata N, Barillot C, Horngger J, Howe R (eds) Medical image computing and computer-assisted intervention MICCAI 2014. Lecture notes in computer science, vol 8673. Springer International Publishing, Berlin, pp 520–527. doi:[10.1007/978-3-319-10404-1_65](https://doi.org/10.1007/978-3-319-10404-1_65)
19. Chen T, Chefdhotel C (2014) Deep learning based automatic immune cell detection for immunohistochemistry images. In: Machine learning in medical imaging. Springer, Berlin, pp 17–24
20. Cireşan DC, Giusti A, Gambardella LM, Schmidhuber J (2013) Mitosis detection in breast cancer histology images with deep neural networks. In: Medical image computing and computer-assisted intervention—MICCAI 2013. Springer, Berlin, pp 411–418
21. Prasoon A, Petersen K, Igel C, Lauze F, Dam E, Nielsen M (2013) Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. In: Medical image computing and computer-assisted intervention—MICCAI 2013. Springer, Berlin, pp 246–253
22. Roth HR, Farag A, Lu L, Turkbey EB, Summers RM (2015) Deep convolutional networks for pancreas segmentation in CT imaging. In: SPIE medical imaging. International Society for Optics and Photonics, pp 94,131G–94,131G
23. Haweai M, Davy A, Warde-Farley D, Biard A, Courville A, Bengio Y, Pal C, Jodoin PM, Larochelle H (2015) Brain tumor segmentation with deep neural networks. arXiv preprint [arXiv:1505.03540](https://arxiv.org/abs/1505.03540)
24. Kleesiek J, Urban G, Hubert A, Schwarz D, Maier-Hein K, Bendszus M, Biller A (2016) Deep MRI brain extraction: a 3D convolutional neural network for skull stripping. NeuroImage 129:460
25. Zhang W, Li R, Deng H, Wang L, Lin W, Ji S, Shen D (2015) Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. NeuroImage 108:214–224
26. Xu J, Luo X, Wang G, Gilmore H, Madabhushi A (2016) A deep convolutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images. Neurocomputing 191:214
27. Margeta J, Criminisi A, Cabrera Lozoya R, Lee DC, Ayache N (2015) Fine-tuned convolutional neural nets for cardiac MRI acquisition plane recognition. Comput Methods Biomed Eng Imaging Vis 1–11
28. Miao S, Wang WJ, Liao R (2016) A CNN regression approach for real-time 2D/3D registration. IEEE Trans Med Imaging PP(99):1–1. doi:[10.1109/TMI.2016.2521800](https://doi.org/10.1109/TMI.2016.2521800)
29. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: convolutional architecture for fast feature embedding. arXiv preprint [arXiv:1408.5093](https://arxiv.org/abs/1408.5093)
30. Tajbakhsh N, Gurudu SR, Liang J (2016) Automated polyp detection in colonoscopy videos using shape and context information. IEEE Trans Med Imaging 35(2):630–644
31. Liang J, Bi J (2007) Computer aided detection of pulmonary embolism with tobogganing and multiple instance classification in CT pulmonary angiography. In: Information processing in medical imaging. Springer, Berlin, pp 630–641
32. Tajbakhsh N, Chi C, Sharma H, Wu Q, Gurudu SR, Liang J (2014) Automatic assessment of image informativeness in colonoscopy. In: Abdominal imaging. Computational and clinical applications. Springer, Berlin, pp 151–158
33. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
34. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2014) Going deeper with convolutions. arXiv preprint [arXiv:1409.4842](https://arxiv.org/abs/1409.4842)

Part III

Segmentation

Chapter 12

Fully Automated Segmentation Using Distance Regularised Level Set and Deep-Structured Learning and Inference

Tuan Anh Ngo and Gustavo Carneiro

Abstract We introduce a new segmentation methodology that combines the structured output inference from deep belief networks and the delineation from level set methods to produce accurate segmentation of anatomies from medical images. Deep belief networks can be used in the implementation of accurate segmentation models if large annotated training sets are available, but the limited availability of such large datasets in medical image analysis problems motivates the development of methods that can circumvent this demand. In this chapter, we propose the use of level set methods containing several shape and appearance terms, where one of the terms consists of the result from the deep belief network. This combination reduces the demand for large annotated training sets from the deep belief network and at the same time increases the capacity of the level set method to model more effectively the shape and appearance of the visual object of interest. We test our methodology on the Medical Image Computing and Computer Assisted Intervention (MICCAI) 2009 left ventricle segmentation challenge dataset and on Japanese Society of Radiological Technology (JSRT) lung segmentation dataset, where our approach achieves the most accurate results of the field using the semi-automated methodology and state-of-the-art results for the fully automated challenge.

This work is an extension of the papers published by the same authors at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014) [1] and the IEEE International Conference on Image Processing (ICIP 2015) [2].

T.A. Ngo (✉)

Department of Computer Science, Faculty of Information Technology,
Vietnam National University of Agriculture, Hanoi, Vietnam
e-mail: ntanh@vnu.edu.vn

G. Carneiro

Australian Centre for Visual Technologies, The University of Adelaide,
Adelaide, SA, Australia
e-mail: gustavo.carneiro@adelaide.edu.au

12.1 Introduction

The segmentation of anatomies from medical images is an important stage in the process of analysing the health of a particular organ. For instance, the segmentation of the endocardium and epicardium from the left ventricle (LV) of the heart using cardiac cine Magnetic Resonance (MR) [3, 4], as shown in Fig. 12.1a, is necessary for the assessment of the cardiovascular system function and structure. The main challenges in the LV segmentation from MR are related to the need to process the various slices from the short axis view, where the area of the LV changes considerably, and to be robust to trabeculations and papillary muscles. Another example is the segmentation of the lung from digital chest X-ray (CXR) [5], as displayed in Fig. 12.1b, which is needed for computing lung volume or estimating shape irregularities [6] for screening and detecting pulmonary pathologies. The lung segmentation problem is challenging due to the large shape and appearance variations of the lung, and the presence clavicle bones and rib cage. One of the main challenges involved in these medical image analysis segmentation problems is that the usefulness of a system is related to the accuracy of its segmentation results, which is usually correlated to the size of the annotated training set available to build the segmentation model. However, large annotated training sets are rarely available for medical image analysis segmentation problems, so it is important to develop methods that can circumvent this demand.

Currently, the main approaches explored in medical image segmentation problems are the following: active contour models, machine learning models, and hybrid active contour and machine learning models. One of most successful methodologies explored in the field is the active contour models [7, 8] that is generally represented by an optimisation that minimises an energy functional which varies the shape of a contour using internal and external hand-crafted constraints. Internal constraints are represented by terms that associate cost with contour bending, stretching or shrinking, and the external constraints use the image data to move the contour towards (or away from) certain features, such as edges. These constraints usually rely on shape or appearance models that require small training sets. The main challenges

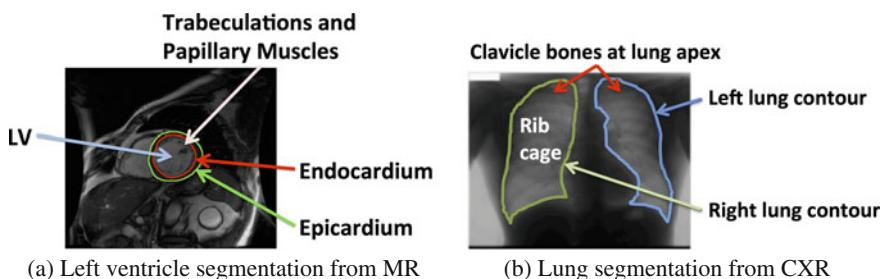


Fig. 12.1 LV segmentation from cardiac cine MR imaging [4] (a), and lung segmentation from digital chest X-ray [5] (b)

faced by active contour models are their inability to model robustly the shape and appearance variations presented by the visual object of interest.

Machine learning methods allow a more robust modelling of the shape and appearance of visual objects [9, 10], which generally translates into more accurate segmentation results. However, the challenges presented in medical image applications in terms of segmentation accuracy requirements and large shape and appearance variations of the visual object of interest imply that the models must have high capacity, requiring a large and rich annotated training set. This means that the acquisition of comprehensive annotated training sets is one of the main foci in the design of machine learning models, which is a complicated task, particularly in medical image analysis. More recent machine learning methodologies are based on models with less capacity, which reduces the need for large and rich training sets, where the idea lies in the combination of active contour models and Markov random fields (MRF) [11–13]. However, the main issue of these approaches is that MRF models present large memory complexity, which limits the size of the input image (or volume) to be segmented.

We propose a new methodology that combines an active contour model (distance regularised level sets) [14] with a machine learning approach (deep belief network) [15]. Deep belief networks (DBN) are represented by a high capacity model that needs large amounts of training data to be robust to the appearance and shape variations of the object of interest, but the two-stage training (consisting of a pre-training based on a large un-annotated training set, followed by a fine-tuning that relies on a relatively small annotated training set) [15] reduces the need for annotated training images. Nevertheless, medical image analysis datasets are generally too small to produce robust DBN models, so its use as a shape term in a level set method can compensate for its lack of robustness and at the same time can improve the accuracy of the level set method. In addition, this combination does not present the large memory complexity faced by MRF models. We show the effectiveness of our approach on two distinct datasets: the Medical Image Computing and Computer Assisted Intervention (MICCAI) 2009 LV segmentation challenge dataset [4] and the Japanese Society of Radiological Technology (JSRT) lung segmentation dataset [16]. Our experiments show that our approach produces the best result in the field when we rely on a semi-automated segmentation (i.e., with manual initialisation) for both datasets. Also, our fully automated approach produces a result that is on par with the current state of the art on the MICCAI 2009 LV segmentation challenge dataset.

12.2 Literature Review

The proposed segmentation methodology can be used in various medical image analysis problems, but we focus on two applications that are introduced in this section. The first application is the segmentation of the endocardial and epicardial borders of the LV from short axis cine MR images, and the second application is the lung segmentation from CXR images. The LV segmentation (see Fig. 12.1a) is challenging

due to the lack of grey-level homogeneity in the imaging of the LV, which happens because of blood flow, papillary muscles and trabeculations, and the low resolution of the apical and basal images [3]. It is possible to categorise LV segmentation approaches with three properties: (1) segmentation method (region and edge based, pixel classification, deformable models, active appearance and shape models), (2) prior information (none, weak, and strong), and (3) automated localisation of the heart (time-based or object detection). According to Petitjean et al.'s analysis [3] of the MICCAI 2009 challenge results [4], the highest accuracy is obtained from image-based methodologies [17, 18] based on thresholding or dynamic programming applied to image segmentation results. However, these methods usually require user interaction and show difficulties in segmenting the LV in all cardiac phases. These drawbacks have been addressed by more sophisticated methods [19–21], but their segmentation accuracy is not as high as the simpler image-based methods above. Moreover, the use of techniques specific to the LV segmentation problem [17, 18, 22] produces more accurate results when compared to more general approaches [19, 23]. The main conclusion reached by Petitjean et al. [3] is that Jolly's methodology [21] is the most effective because it is fully automatic and offers the best compromise between accuracy and generalisation. The most effective methodology in the MICCAI 2009 challenge for the semi-automated case (i.e., that requires a user input in terms of the initialisation for the segmentation contour) has been developed by Huang et al. [18].

The challenges in the lung segmentation problem (see Fig. 12.1b) are related to the presence of strong edges at the rib cage and clavicle, the lack of a consistent lung shape among different cases, and the appearance of the lung apex. Current techniques are based on methods that combine several methodologies, such as landmark learning and active shape and appearance models [24, 25] or MRF and non-rigid registration [5]. Although presenting state-of-the-art segmentation results, these methods show some drawbacks: landmark learning is a hard problem that is based on hand-crafted feature detector and extractor, active shape and appearance models make strong assumptions about the distribution of landmarks, and MRF inference has high memory complexity that limits the input image size.

Finally, it is important to note that image segmentation can be posed as a structured output learning and inference problem [26], where the classification is represented by a multidimensional binary vector. Traditionally, structured output models use a large margin learning formulation [27], but a natural way to represent a structured learning is with a multi-layer perceptron, where the output layer consists of a multidimensional binary vector denoting the segmentation [28]. The recent renaissance of deep learning methods originated from the development of an efficient learning algorithm for training DBN [15], which allowed the development of structured inference and learning with DBN [29–32]. Similarly, the method proposed by Farabet et al. [30] parses a scene into several visual classes using convolutional neural networks. Nevertheless, the papers above show that DBNs can work solidly in structured output problems only with the availability of large annotated training sets that allows the modelling of a robust DBN.

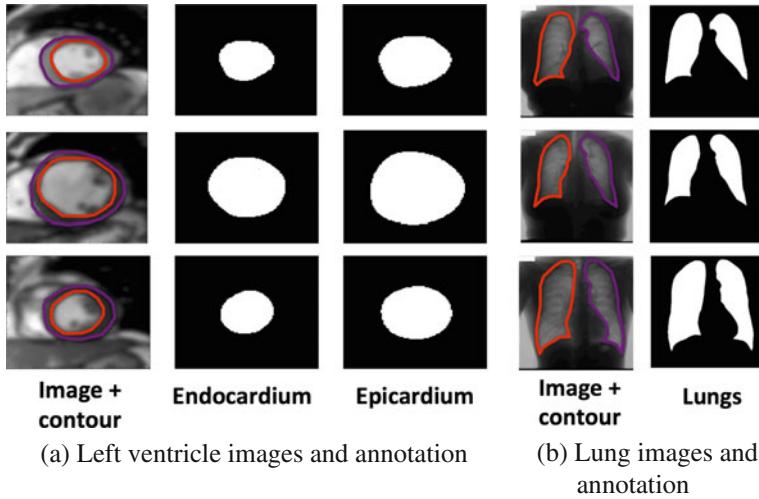


Fig. 12.2 Left ventricle images \mathbf{v} with overlaid endocardial and epicardial segmentation contours \mathbf{c} and respective segmentation maps \mathbf{y} in **(a)**, and lung images with overlaid left and right segmentation contours and respective segmentation maps in **(b)**

12.3 Methodology

In order to explain the segmentation algorithm, let us assume that we have an annotated dataset (Fig. 12.2), represented by $\mathcal{D} = \{(\mathbf{v}, \mathbf{c}, \mathbf{y})_i\}_{i=1}^{|\mathcal{D}|}$, where $\mathbf{v} : \Omega \rightarrow \mathbb{R}$ represents an image of the visual object of interest (with $\Omega \subseteq \mathbb{R}^2$ denoting the image lattice), $\mathbf{c} : [0, 1] \rightarrow \Omega$ denotes the explicit contour representation of the segmentation, and the binary segmentation map is represented by $\mathbf{y} : \Omega \rightarrow \{0, 1\}$, where 1 represents the foreground (i.e., points inside the contour \mathbf{c}) and 0 denotes the background (i.e., points outside the contour \mathbf{c}). Below, we first explain the segmentation method based on the distance regularised level set (DRLS), then we describe the DBN model and the shape prior.

The main segmentation algorithm is based on the distance regularised level set (DRLS) method [14], where the energy functional is represented by:

$$\mathcal{E}(\phi, \phi_{\text{DBN}}, \phi_{\text{PRIOR}}) = \mu \mathcal{R}_p(\phi) + \mathcal{E}_{\text{ext}}(\phi, \phi_{\text{DBN}}, \phi_{\text{PRIOR}}), \quad (12.1)$$

where $\phi : \Omega \rightarrow \mathbb{R}$ represents the signed distance function, defined by

$$\phi(\mathbf{x}) = \begin{cases} -d(\mathbf{x}, \Omega^{out}), & \text{if } \mathbf{x} \in \Omega^{in} \\ +d(\mathbf{x}, \Omega^{in}), & \text{if } \mathbf{x} \in \Omega^{out} \end{cases}, \quad (12.2)$$

where $\Omega^{in} = \{\mathbf{x} \in \Omega | \mathbf{y}(\mathbf{x}) = 1\}$, $\Omega^{out} = \{\mathbf{x} \in \Omega | \mathbf{y}(\mathbf{x}) = 0\}$, and $d(\mathbf{x}, \Omega) = \inf_{\mathbf{z} \in \Omega} \|\mathbf{x} - \mathbf{z}\|_2$, assuming that \mathbf{y} denotes the segmentation map. Also in (12.1), the distance

regularisation $\mathcal{R}_p(\phi) = \int_{\Omega} 0.5(|\nabla\phi(\mathbf{x})| - 1)^2 d\mathbf{x}$ guarantees that $|\nabla\phi(\mathbf{x})| \approx 1$, which avoids the re-initialisations during the segmentation process [14] (a common issue in level set methods), and

$$\mathcal{E}_{\text{ext}}(\phi, \phi_{\text{DBN}}, \phi_{\text{PRIOR}}) = \lambda \mathcal{L}(\phi) + \alpha \mathcal{A}(\phi) + \beta \mathcal{S}(\phi, \phi_{\text{DBN}}) + \gamma \mathcal{S}(\phi, \phi_{\text{PRIOR}}), \quad (12.3)$$

with the length term $\mathcal{L}(\phi) = \int_{\Omega} g\delta(\phi(\mathbf{x}))|\nabla\phi(\mathbf{x})|d\mathbf{x}$ (with $\delta(\cdot)$ denoting the Dirac delta function and $g \triangleq \frac{1}{1+|\nabla G_g * I|}$ representing the edge indicator function), the area $\mathcal{A}(\phi) = \int_{\Omega} gH(-\phi(\mathbf{x}))d\mathbf{x}$ (with $H(\cdot)$ denoting the Heaviside step function), and $\mathcal{S}(\phi, \phi_{\kappa}) = \int_{\Omega} (\phi(\mathbf{x}) - \phi_{\kappa}(\mathbf{x}))^2 d\mathbf{x}$ (with $\kappa \in \{\text{DBN}, \text{PRIOR}\}$) representing the shape term [33] that drives ϕ either towards the shape ϕ_{DBN} , which is the distance function inferred from the deep belief network (DBN) structured inference described below, or the shape prior ϕ_{PRIOR} , estimated from the training set and also described in more detail below. The gradient flow of the energy $\mathcal{E}(\phi)$ is then defined as follows:

$$\begin{aligned} \frac{\partial\phi}{\partial t} = & \mu \operatorname{div}(d_p(|\nabla\phi|)\nabla\phi) + \lambda\delta(\phi)\operatorname{div}(g\frac{\nabla\phi}{|\nabla\phi|}) + \alpha g\delta(\phi) + \\ & 2\beta(\phi(\mathbf{x}) - \phi_{\text{DBN}}(\mathbf{x})) + 2\gamma(\phi(\mathbf{x}) - \phi_{\text{PRIOR}}(\mathbf{x})), \end{aligned} \quad (12.4)$$

where $\operatorname{div}(\cdot)$ denotes the divergence operator, and $d_p(\cdot)$ denotes the derivative of the function $p(\cdot)$ defined in (12.1).

The segmentation is obtained from the minimisation of the energy functional in (12.1) from the steady solution of the gradient flow equation [14] $\frac{\partial\phi}{\partial t} = -\frac{\partial\mathcal{E}}{\partial\phi}$, where $\partial\mathcal{E}/\partial\phi$ is the Gâteaux derivative of the functional $\mathcal{E}(\phi)$ and $\frac{\partial\phi}{\partial t}$ is defined in (12.4). The main idea of the DRLS [14] is then to iteratively follow the steepest descent direction (12.4) until convergence, resulting in the final steady solution.

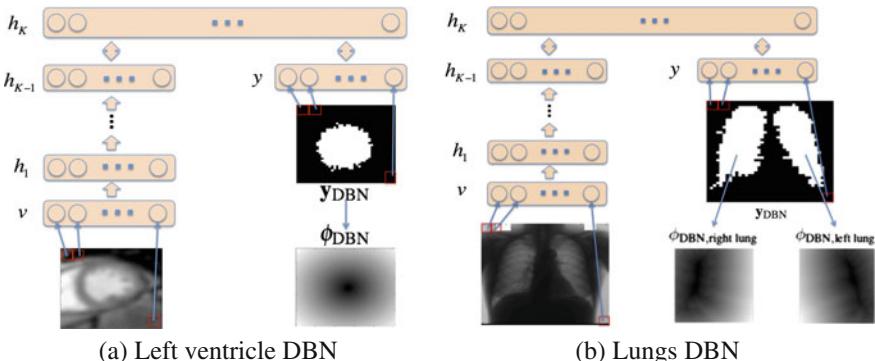


Fig. 12.3 Deep belief network that produces the segmentation maps \mathbf{y}_{DBN} and respective signed distance function ϕ_{DBN} for the left ventricle structures (epicardium and endocardium) in (a) and left and right lungs in (b)

The structured inference from the DBN (Fig. 12.3) produces the following segmentation map:

$$\mathbf{y}_{\text{DBN}} = \arg \max_{\mathbf{y}} \sum_{\mathbf{h}_1} \dots \sum_{\mathbf{h}_K} P(\mathbf{v}, \mathbf{h}_1, \dots, \mathbf{h}_K, \mathbf{y}; \Theta), \quad (12.5)$$

where \mathbf{v} represents the input image, $\mathbf{h}_k \in \{0, 1\}^{|\mathbf{h}_k|}$ represents the $|\mathbf{h}_k|$ hidden nodes of layer $k \in \{1, \dots, K\}$ of the deep belief network, and Θ denotes the DBN parameters (weights and biases). The probability term in (12.5) is computed as

$$P(\mathbf{v}, \mathbf{h}_1, \dots, \mathbf{h}_K, \mathbf{y}) = P(\mathbf{h}_K, \mathbf{h}_{K-1}, \mathbf{y}) \left(\prod_{k=1}^{K-2} P(\mathbf{h}_{k+1} | \mathbf{h}_k) \right) P(\mathbf{h}_1 | \mathbf{v}), \quad (12.6)$$

where $-\log P(\mathbf{h}_K, \mathbf{h}_{K-1}, \mathbf{y}) \propto \mathcal{E}_{\text{RBM}}(\mathbf{h}_K, \mathbf{h}_{K-1}, \mathbf{y})$ with

$$\mathcal{E}_{\text{RBM}}(\mathbf{h}_K, \mathbf{h}_{K-1}, \mathbf{y}) = -\mathbf{b}_K^\top \mathbf{h}_K - \mathbf{a}_{K-1}^\top \mathbf{h}_{K-1} - \mathbf{a}_y^\top \mathbf{y} - (\mathbf{h}_K)^\top \mathbf{W}_K \mathbf{h}_{K-1} - (\mathbf{h}_K)^\top \mathbf{W}_y \mathbf{y} \quad (12.7)$$

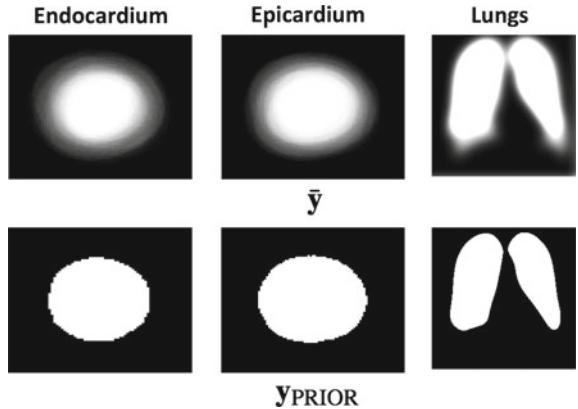
representing the energy function of a restricted Boltzmann machine (RBM) [15], where $\mathbf{b}_K, \mathbf{a}_{K-1}, \mathbf{a}_y$ denote the bias vectors and $\mathbf{W}_K, \mathbf{W}_y$ are the weight matrices. Also in (12.6), we have

$$P(\mathbf{h}_{k+1} | \mathbf{h}_k) = \prod_j P(\mathbf{h}_{k+1}(j) = 1 | \mathbf{h}_k), \quad (12.8)$$

with $P(\mathbf{h}_{k+1}(j) = 1 | \mathbf{h}_k) = \sigma(\mathbf{b}_{k+1}(j) + \mathbf{h}_k^\top \mathbf{W}_{k+1}(:, j))$, $P(\mathbf{h}_1(j) = 1 | \mathbf{v}_{\mathbf{m}_\phi}) = \sigma(\mathbf{b}_1(j) + \frac{\mathbf{v}_{\mathbf{m}_\phi}^\top \mathbf{W}_1(:, j)}{\sigma^2})$ (we assume zero-mean Gaussian visible units for the DBN), where $\sigma(x) = \frac{1}{1+e^{-x}}$, the operator (j) returns the j^{th} vector value, and $(:, j)$ returns the j^{th} matrix column. The signed distance function ϕ_{DBN} is then computed with (12.2). The DBN in (12.5) is trained in two stages. The first stage is based on the unsupervised bottom-up training of each pair of layers, where the weights and biases of the network are learned to build an auto-encoder for the values at the bottom layer, and the second stage is based on a supervised training that uses the segmentation map \mathbf{y} as the training label [15]. The structured inference process consists of taking the input image and performing bottom-up inferences, until reaching the top two layers, which form an RBM, and then initialise the layer $\mathbf{y} = \mathbf{0}$ and perform Gibbs sampling on the layers $\mathbf{h}_K, \mathbf{h}_{K-1}$ and \mathbf{y} until convergence [15]. The signed distance function ϕ_{DBN} is then computed with (12.2) from \mathbf{y}_{DBN} .

The shape prior (Fig. 12.4) is computed with the mean of the manual annotations $\{\mathbf{y}_i\}_{i \in \mathcal{T}}$, where $\mathcal{T} \subset \mathcal{D}$ denotes the training set, as follows: $\bar{\mathbf{y}}(\mathbf{x}) = \frac{1}{|\mathcal{T}|} \sum_{i=1}^{|\mathcal{T}|} \mathbf{y}_i(\mathbf{x})$, where $\mathbf{x} \in \Omega$. Assuming that each element of the mean map $\bar{\mathbf{y}}$ is between 0 and 1, the shape prior is computed as

Fig. 12.4 Shape priors $\mathbf{y}_{\text{PRIOR}}$ (computed from $\bar{\mathbf{y}}$ using (12.9)) for endocardium, epicardium and lungs



$$\mathbf{y}_{\text{PRIOR}}(\mathbf{x}) = \begin{cases} 1, & \text{if } \bar{\mathbf{y}}(\mathbf{x}) > 0.5 \\ 0, & \text{if } \bar{\mathbf{y}}(\mathbf{x}) \leq 0.5 \end{cases}. \quad (12.9)$$

The signed distance function ϕ_{PRIOR} is then computed with (12.2) from $\mathbf{y}_{\text{PRIOR}}$.

The segmentation using the combination of DRLS, DBN and shape prior is explained in Algorithm 1, which iteratively runs DRLS until convergence using the segmentation results from the DBN and from the shape prior as two of its optimisation terms. Notice that the initial segmentation ϕ_0 can be manually provided, which results in a semi-automated segmentation, or automatically produced, generating a fully automated segmentation method.

Algorithm 1 Combined DRLS and DBN Segmentation

- 1: INPUT: test image \mathbf{v} , shape prior $\mathbf{y}_{\text{PRIOR}}$ and initial segmentation ϕ_0
 - 2: Compute signed distance function ϕ_{PRIOR} from map $\mathbf{y}_{\text{PRIOR}}$ with (12.2)
 - 3: Infer \mathbf{y}_{DBN} from \mathbf{v} using (12.5)
 - 4: Compute signed distance function ϕ_{DBN} from map \mathbf{y}_{DBN} with (12.2)
 - 5: **for** $t = 1:T$ **do**
 - 6: Run DRLS using $\phi_{t-1}, \phi_{\text{DBN}}, \phi_{\text{PRIOR}}$ to produce ϕ_t
 - 7: **end for**
 - 8: Segmentation is the zero level set $\mathcal{C} = \{\mathbf{x} \in \Omega | \phi_T(\mathbf{x}) = 0\}$
-

12.3.1 Left Ventricle Segmentation

In this section, we present our **fully automated left ventricle segmentation method**. A cardiac cine MR sequence consists of K volumes $\{\mathbf{V}_i\}_{i=1}^K$, each representing a particular cardiac phase, where each volume comprises a set of N images $\{\mathbf{v}_j\}_{j=1}^N$, also known as volume slices, obtained using the short axis view (Fig. 12.5). We assume to

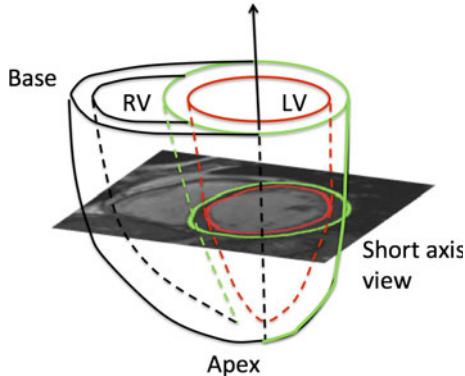


Fig. 12.5 Visualisation of an image on the short axis view, where RV and LV stand for right and left ventricles, respectively, and the *red* contour represents the endocardium contour and *green* denotes the epicardium

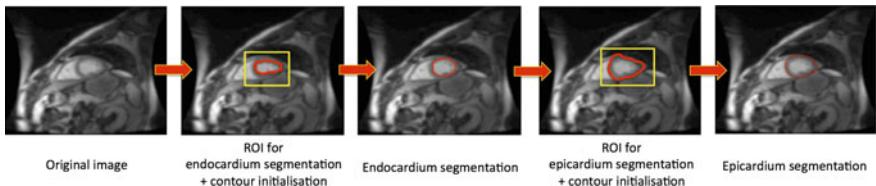
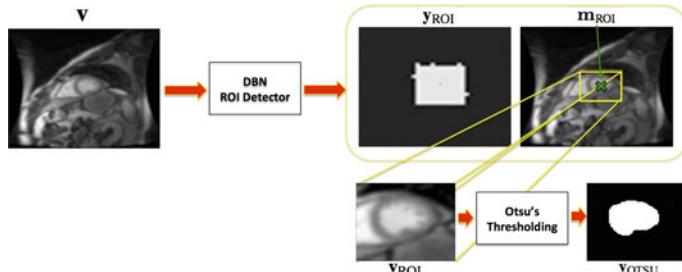


Fig. 12.6 All steps for the left ventricle segmentation—Fig. 12.7 depicts each step in more detail

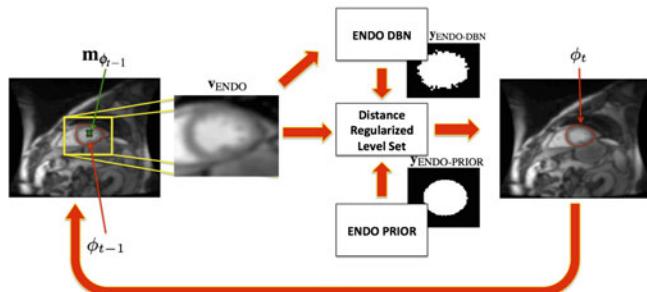
have annotation only at the end diastolic (ED) and end systolic (ES) cardiac phases (i.e., only two out of the K phases available) for all N images in these two volumes. In each of these annotated images, the explicit endocardial and epicardial contour representations are denoted by \mathbf{c}_{ENDO} and \mathbf{c}_{EPI} , respectively, and the segmentation maps are denoted by \mathbf{y}_{ENDO} and \mathbf{y}_{EPI} . The set of annotated sequences is represented by $\mathcal{D} = \{(\mathbf{v}, \mathbf{c}_{\text{ENDO}}, \mathbf{c}_{\text{EPI}}, \mathbf{y}_{\text{ENDO}}, \mathbf{y}_{\text{EPI}}, i, q)_s\}_{s \in \{1, \dots, S\}, i \in \{1, \dots, N_s\}, q \in \{\text{ED, ES}\}}$, where s denotes the sequence index (each sequence represents one patient), i denotes the index to an image within the sequence s , and q represents the cardiac phase (Fig. 12.2). Note that our methodology runs the segmentation process slice by slice in each of the ED and ES volumes, using the steps displayed in Fig. 12.6.

12.3.2 Endocardium Segmentation

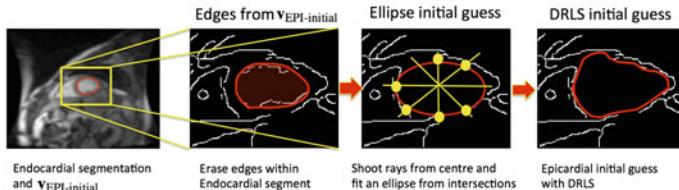
For segmenting the endocardium, it is first necessary to detect a region of interest (ROI) that fully contains the left ventricle. This ROI detection uses the structured inference computed from a DBN, which outputs an image region that is used in the estimation of the initial endocardium segmentation ϕ_0 (see Algorithm 1 and



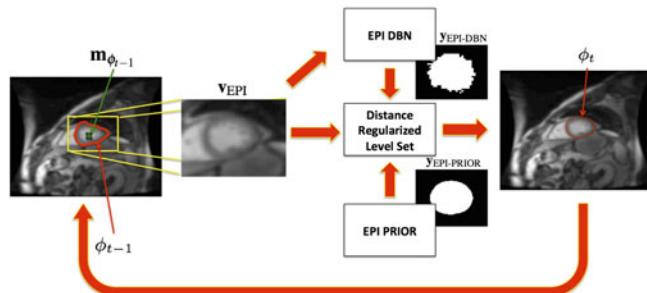
(a) ROI Detection and Initial Endocardium Segmentation



(b) Endocardium Segmentation



(c) Initial Epicardium Segmentation



(d) Epicardium Segmentation

Fig. 12.7 Models of the ROI detection and initial endocardium segmentation (a), final endocardium segmentation (b), initial epicardium segmentation (c) and final epicardium segmentation (d)

Fig. 12.7a). The endocardium segmentation follows Algorithm 1 and is represented in Fig. 12.7b. We explain the details of the endocardial segmentation below.

12.3.2.1 ROI DBN Detection and Initial Endocardium Segmentation

For the ROI detection, we use the DBN model introduced in (12.5), with parameters Θ_{ROI} , that produces the segmentation map $y_{ROI} : \Omega \rightarrow [0, 1]$. The training set comprises images v and their respective ROI segmentation maps that are automatically built from the manual endocardial border delineations c_{ENDO} by producing a segmentation map with 0's everywhere except at a square of 1's with size $M_{ROI} \times M_{ROI}$, centred at the centre of gravity of the annotation c_{ENDO} (see training samples in Fig. 12.8b).

After estimating the ROI segmentation map y_{ROI} , a rough endocardial border delineation is estimated by first applying the following function:

$$(v_{ROI}, m_{ROI}) = f_R(y_{ROI}, v, M_{ROI}), \quad (12.10)$$

where m_{ROI} is the centre of gravity of y_{ROI} computed as $m_{ROI} = \int_{\Omega} \mathbf{x} h(y_{ROI}) d\mathbf{x}$, with $h(y_{ROI}) = \frac{H(y_{ROI})}{\int_{\Omega} H(y_{ROI}) d\mathbf{x}}$ and $H(\cdot)$ denoting the Heaviside step function, and v_{ROI} is a sub-image of size $M_{ROI} \times M_{ROI}$ extracted with $v_{ROI} = v(m_{ROI} \pm M_{ROI}/2)$. Then, Otsu's thresholding [34] is run on the sub-image v_{ROI} , where the convex hull of the connected component linked to the centre $M_{ROI}/2$ is returned as the rough endocardial border delineation with $y_{OTSU} = f_O(v_{ROI})$, as displayed in Fig. 12.8a. Recall that Otsu's thresholding [34] is a segmentation method that binarises a grey-level image using a threshold estimated to minimise the intra-class variance of the grey values,

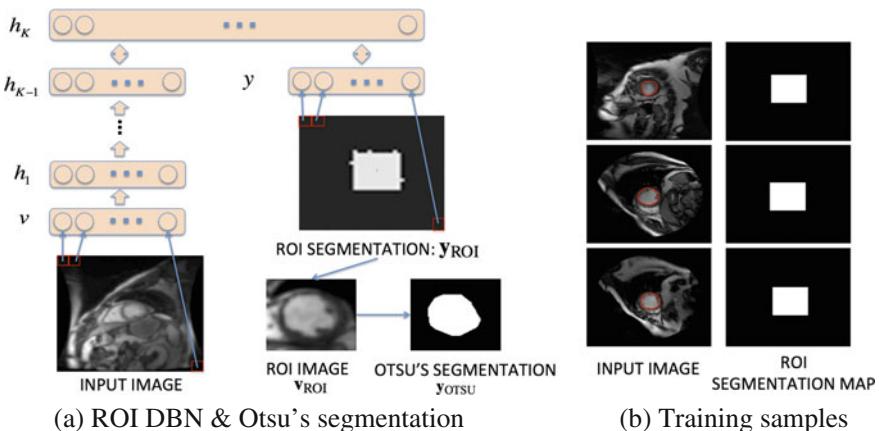


Fig. 12.8 ROI DBN Model and Otsu's segmentation (a) and training samples for the ROI DBN (b)

where the classes are defined by the pixel values above and below this threshold. This segmentation is used to form the initial signed distance function (Algorithm 1), as follows:

$$\phi_0 = f_\phi(\mathbf{y}_{\text{OTSU}}, \mathbf{m}_{\text{ROI}}, M_{\text{ROI}}, \mathbf{v}), \quad (12.11)$$

where we first create a temporary binary map $\hat{\mathbf{y}} : \Omega \rightarrow \{0, 1\}$ with a map of the size of \mathbf{v} containing only zeros, as in $\hat{\mathbf{y}} = \mathbf{0}_{\text{size}(\mathbf{v})}$ (the function $\text{size}(i)$ returns the size of the image), then we fill this map with the result from \mathbf{y}_{OTSU} centred at \mathbf{m}_{ROI} , with $\hat{\mathbf{y}}(\mathbf{m}_{\text{ROI}} \pm M_{\text{ROI}}/2) = \mathbf{y}_{\text{OTSU}}(M_{\text{ROI}}/2 \pm M_{\text{ROI}}/2)$. Finally, the signed distance function ϕ_0 is computed from $\hat{\mathbf{y}}$ with (12.2).

12.3.2.2 Endocardium Segmentation Combining DRLS and DBN

Given the initial segmentation ϕ_0 defined in (12.11), we run a slightly modified version of the segmentation method in Algorithm 1. The main difference is the introduction of an outer loop between lines 3 and 7, inclusive, which changes the sub-image of \mathbf{v} that will be used as the input for the ENDO DBN, where the change is related to the sub-image centre given by the centre of gravity of ϕ_{t-1} , computed with $\mathbf{m}_{\phi_{t-1}} = \int_{\Omega} \mathbf{x} h(\phi_{t-1}(\mathbf{x})) d\mathbf{x}$ with $h(\phi_{t-1}) = \frac{H(-\phi_{t-1})}{\int_{\Omega} H(-\phi_{t-1}) d\mathbf{x}}$ (see Fig. 12.7b). Also the segmentation in line 6 of Algorithm 1 has inputs $\phi_{t-1}, \phi^{\text{ENDO-DBN},q}$ and $\phi^{\text{ENDO-PRIOR},q}$ (below, we provide details on these last two functions), with $t \in \{1, 2, \dots, T\}$ and $q \in \{\text{ED,ES}\}$, where the shape terms, from (12.3), are denoted by $\mathcal{S}(\phi, \phi_{\kappa}) = \int_{\Omega} (\phi(\mathbf{x}) - \phi_{\kappa}(\mathbf{x} + \mathbf{m}_{\phi_{t-1}}))^2 d\mathbf{x}$ (with $\kappa \in \{\text{(ENDO-DBN}, q), \text{(ENDO-PRIOR}, q)\}$, and $q \in \{\text{ED,ES}\}$). This segmentation algorithm results in the signed distance function $\phi^{\text{ENDO},q}_*$, from which we can compute the estimated endocardial contour from its zero level set $\{\mathbf{x} \in \Omega | \phi^{\text{ENDO},q}_*(\mathbf{x}) = 0\}$ and endocardial binary segmentation map $\mathbf{y}^{\text{ENDO},q}_* = H(-\phi^{\text{ENDO},q}_*)$.

The ENDO DBN used at this stage is the same as the one depicted in Fig. 12.3a, where the input image is a sub-image of \mathbf{v} of size $M_{\text{ENDO}} \times M_{\text{ENDO}}$ centred at position $\mathbf{m}_{\phi_{t-1}}$, where this sub-image is represented by \mathbf{v}_{ENDO} . We have two distinct DBNs, one to segment images for $q = \text{ES}$ phase and another for $q = \text{ED}$ phase of the cardiac cycle, where the training set is formed by samples $\{(\mathbf{v}_{\text{ENDO}}, \mathbf{y}_{\text{ENDO}}, i, q)_s\}_{s \in \{1, \dots, S\}, i \in \{1, \dots, N_s\}, q \in \{\text{ED,ES}\}}$ extracted from the original training set with $f_R(\cdot)$, defined in (12.10). The segmentation from ENDO DBN produces $\mathbf{y}_{\text{ENDO-DBN},q}$ from input \mathbf{v}_{ENDO} using (12.5). The segmentation $\mathbf{y}_{\text{ENDO-DBN},q}$ can then be used to compute the signed distance function $\phi_{\text{ENDO-DBN},q}$ with (12.2). Finally, the ENDO shape prior, represented by $\mathbf{y}_{\text{ENDO-PRIOR},q}$, is computed as defined in (12.9) using the binary segmentation maps $\{(\mathbf{y}_{\text{ENDO}}, i, q)_s\}_{s \in \{1, \dots, S\}, i \in \{1, \dots, N_s\}, q \in \{\text{ED,ES}\}}$. Similarly, $\mathbf{y}_{\text{ENDO-PRIOR},q}$ is used to calculate the signed distance function $\phi_{\text{ENDO-PRIOR},q}$ with (12.2).

12.3.3 Epicardium Segmentation

The epicardium segmentation also follows two steps, comprising an initial segmentation, which produces a square region containing the epicardium and an initial estimation of its border, similarly to the approach in Sect. 12.3.2.1 (Fig. 12.7c). The second step involves an optimisation with DRLS [14], similar to the one presented above in Sect. 12.3.2.2 (Fig. 12.7d).

12.3.3.1 Initial Epicardium Segmentation

The epicardium segmentation process is initialised with a rough delineation based on the endocardium detection (see Fig. 12.7c). Specifically, after the endocardium segmentation is finalised, we estimate the borders of the epicardium segmentation by first running the Canny edge detector [35] that outputs the edges within the sub-image $\mathbf{v}_{\text{EPI-initial}}$ of size $M_{\text{EPI}} \times M_{\text{EPI}}$ centred at position $\mathbf{m}_{\text{EPI-initial},q} = \int_{\Omega} \mathbf{x} h(\phi_{\text{ENDO},q}^*(\mathbf{x})) d\mathbf{x}$ with $h(\phi_{\text{ENDO},q}^*) = \frac{H(-\phi_{\text{ENDO},q}^*)}{\int_{\Omega} H(-\phi_{\text{ENDO},q}^*) d\mathbf{x}}$. The edges lying in the region where $H(-\phi_{\text{ENDO},q}^*)$ equals to one (this region represents blood pool found by the endocardium segmentation) are erased and then, by “shooting” 20 rays (18 degrees apart from each other) from the centre $\mathbf{m}_{\text{EPI-initial},q}$ and recording the intersection position between each ray and the first edge it crosses, we form a set of points that are likely to belong to the endocardial border. At this stage, since it is expected that the endocardial border will be relatively close to the epicardial border, we only record the points that are within a limited range from the original endocardial border (specifically, we expect the epicardial border to be within 1.05 and 1.1 of the length of the ray from $\mathbf{m}_{\text{EPI-initial}}$ to the endocardial border; otherwise no point is recorded—these numbers are estimated from the 95% confidence interval of the distance between the endocardium and epicardium annotations from the training set). Finally, by fitting an ellipse to these points and running a small number of iterations of the original DRLS [14] (which is the model in (12.1)–(12.3) with $\beta = \gamma = 0$), we form the initial epicardium segmentation that is represented by a map $\mathbf{y}_{\text{EPI-initial}}$, which is then used to form the initial signed distance function $\phi_0 = f_{\phi}(\mathbf{y}_{\text{EPI-initial}}, \mathbf{m}_{\text{EPI-initial}}, M_{\text{EPI}}, \mathbf{v})$, as defined in (12.2).

12.3.3.2 Epicardium Segmentation Combining DRLS and DBN

Using the initial epicardium segmentation ϕ_0 from Sect. 12.3.3.1 above, we run the segmentation method in Algorithm 1 with the same modification explained in Sect. 12.3.2.2 (i.e., the outer loop between lines 3 and 7 that changes the sub-image of \mathbf{v} used in the input for the EPI DBN according to the centre of gravity $\mathbf{m}_{\phi_{t-1}}$ of ϕ_{t-1}). The segmentation in line 6 of Algorithm 1 has inputs $\phi_{t-1}, \phi_{\text{EPI-DBN},q}$ and $\phi_{\text{EPI-PRIOR},q}$ (please see details below on these last two functions), with $t \in \{1, 2, \dots, T\}$ and $q \in \{\text{ED,ES}\}$, where the shape terms, from (12.3), are denoted by $\mathcal{S}(\phi, \phi_k) =$

$\int_{\Omega} (\phi(\mathbf{x}) - \phi_{\kappa}(\mathbf{x} + \mathbf{m}_{\phi_{t-1}}))^2 d\mathbf{x}$ (with $\kappa \in \{\text{(EPI-DBN}, q), \text{(EPI-PRIOR}, q)\}$, and $q \in \{\text{ED,ES}\}$). This segmentation algorithm results in the signed distance function $\phi_{\text{EPI},q}^*$, from which we can compute the estimated epicardial contour from its zero level set $\{\mathbf{x} \in \Omega | \phi_{\text{EPI},q}^*(\mathbf{x}) = 0\}$ and epicardial binary segmentation map $\mathbf{y}_{\text{EPI},q}^* = H(-\phi_{\text{EPI},q}^*)$.

The EPI DBN is the same as the one displayed in Fig. 12.3a, where the input image is represented by \mathbf{v}_{EPI} , centred at $\mathbf{m}_{\phi_{t-1}}$ and of size $M_{\text{EPI}} \times M_{\text{EPI}}$. We can estimate the parameters of two DBNs for $q \in \{\text{ED,ES}\}$ with the following training set $\{(\mathbf{v}_{\text{EPI}}, \mathbf{y}_{\text{EPI}}, i, q)_s\}_{s \in \{1, \dots, S\}, i \in \{1, \dots, N_s\}, q \in \{\text{ED,ES}\}}$ extracted from the original training set with $f_R(\cdot)$, defined in (12.10). The inference process is the same as the one defined in (12.5), resulting in $\mathbf{y}_{\text{EPI-DBN},q}$, which is used to compute the signed distance function $\phi_{\text{EPI-DBN},q}$ with (12.2). Finally, the EPI shape prior, denoted by $\mathbf{y}_{\text{EPI-PRIOR},q}$, is computed from (12.9) using the binary segmentation maps $\{(\mathbf{y}_{\text{EPI}}, i, q)_s\}_{s \in \{1, \dots, S\}, i \in \{1, \dots, N_s\}, q \in \{\text{ED,ES}\}}$. Similarly, $\mathbf{y}_{\text{EPI-PRIOR},q}$ is used to calculate the signed distance function $\phi_{\text{EPI-PRIOR},q}$ with (12.2).

12.3.4 Lung Segmentation

In this section, we present our **semi-automated lung segmentation method**. The annotated chest radiograph database (Fig. 12.2) is represented by $\mathcal{D} = \{(\mathbf{v}, \mathbf{c}, \mathbf{y}, q)_i\}_{i=1}^{|\mathcal{D}|}$, where \mathbf{v} represents an image, \mathbf{c} denotes the explicit contour representation, \mathbf{y} the respective binary segmentation map, and $q \in \{\text{left lung, right lung}\}$.

The segmentation Algorithm 1 takes a manually provided initial segmentation ϕ_0 and, in each iteration, uses the functions ϕ_{t-1} , $\phi_{\text{DBN},q}$ and $\phi_{\text{PRIOR},q}$, with $t \in \{1, 2, \dots, T\}$ and $q \in \{\text{left lung, right lung}\}$, and the final steady solution of this optimisation is represented by ϕ_q^* , from which we can compute the estimated contour from the zero level set $\{\mathbf{x} \in \Omega | \phi_q^*(\mathbf{x}) = 0\}$ and the binary segmentation map $\mathbf{y}_q^* = H(-\phi_q^*)$. The DBN is the one shown in Fig. 12.3b, where the resulting segmentation \mathbf{y}_{DBN} of both lungs is divided into two separate signed distance functions: $\phi_{\text{DBN, right lung}}$ for the right lung and $\phi_{\text{DBN, left lung}}$ for the left lung, where this separation is done via connected component analysis.

12.4 Experiments

12.4.1 Data Sets and Evaluation Measures

The proposed **endocardium and epicardium segmentation** method is assessed with the dataset and the evaluation introduced in the MICCAI 2009 LV segmentation challenge [4]. This dataset contains 45 cardiac short axis (SAX) cine MR, which are divided into three sets (online, testing and training sets) of 15 sequences,

with each sequence containing four ischemic heart failures, four non-ischemic heart failures, four LV hypertrophies and three normal cases. Each of those sequences has been acquired during a 10–15 s breath-holds, with a temporal resolution of 20 cardiac phases over the heart cycle, starting from the ED cardiac phase, and containing six to 12 SAX images obtained from the atrioventricular ring to the apex (thickness = 8 mm, gap = 8 mm, FOV = 320 mm × 320 mm, matrix = 256 × 256). Expert annotations are provided for endocardial contours in all slices at ED and ES cardiac phases, and for epicardial contours only at ED cardiac phase. The evaluation used to assess the algorithms submitted to the MICCAI 2009 LV segmentation challenge is based on the following three measures: (1) percentage of “good” contours, (2) the average Dice metric (ADM) of the “good” contours, and (3) average perpendicular distance (APD) of the “good” contours. A segmentation is classified as good if $APD < 5$ mm. During the MICCAI 2009 LV Segmentation Challenge [4], the organisers first released the training and test sets, where the training set contained the manual annotation, but the test set did not include the manual annotation. The online dataset only became available a few days before the challenge day, so that the participants could submit their segmentation results for assessment. The challenge organisers reported all segmentation results for all datasets that were available from the participants. Currently all three data sets with their respective expert annotations are publicly available. Given that most of the results from the challenge participants are available for the training and test sets, we decided to use the training set to estimate all DBN parameters, the online set for validating some DBN parameters (e.g., number of layers and number of nodes per layer), and the test set exclusively for testing (since this is the set which has the majority of results from the participants).

The proposed **lung segmentation** method is assessed with the Japanese Society of Radiological Technology (JSRT) dataset [16], which contains 247 chest radiographs with manual segmentations of lung fields, heart and clavicles [25]. Out of these 247 chest radiographs, 154 contain lung nodules (100 malignant, 54 benign) and 93 have no nodules, and each sample is represented by 12-bit grey-scale image with size 2048 × 2048 pixels and 0.175 mm pixel resolution. This database is randomly split into three sets: training (84 images), validation (40 images) and test (123 images), and the assessment is based on following three measures: Jaccard Similarity Coefficient (Ω), Dice Coefficient (DSC) and Average Contour Distance (ACD) [5].

12.4.2 Experimental Setup

For the **endocardium and epicardium segmentation**, the training set is used to model the ROI DBN, ENDO DBN and EPI DBN network (weights and biases), the shape priors and for estimating the weights of the DRLS method (i.e., μ , λ , α , β , γ in (12.1) and (12.3)); while the online set is used for the model selection of the DBNs (i.e., the estimation of the number of hidden layers and the number of nodes per layer for the DBNs). For this model selection, we use the online set to estimate the number of hidden layers (from two to four hidden layers), and the number of nodes

per hidden layer (from 100 to 2000 nodes per layer in intervals of 100 nodes). For the ROI DBN, the estimated model is as follows: two hidden layers with 1300 nodes in the first layer and 1500 in the second, and the input and segmentation layers with 40×40 nodes (i.e., the image is resized from 256×256 to 40×40 using standard blurring and downsampling techniques). For the ENDO DBN trained for the ED cycle, we reach the following model: two hidden layers with 1000 nodes in the first layer and 1000 in the second, and the input and segmentation layers with size 40×40 nodes (again, image is resized from $M_{\text{ENDO}} \times M_{\text{ENDO}}$ to 40×40). The ENDO DBN for the ES cycle has the following configuration: two hidden layers with 700 nodes in the first layer and 1000 in the second, and the input and segmentation layers with size 40×40 . The EPI DBN for the ED cycle has the following configuration: two hidden layers with 1000 nodes in the first layer and 1000 in the second, and the input and segmentation layers with size 40×40 nodes (image resized from $M_{\text{EPI}} \times M_{\text{EPI}}$ to 40×40). For training all DBN models, we augment the training set, where we generate additional training images by translating the original training image (and its annotation) within a range of ± 10 pixels. More specifically, we have 105 ED images and 75 ES annotated training images (from the 15 training volumes), and in addition to the original training image, we generate 40 additional images with the translations mentioned above. Therefore, in total we have $105 \times 41 = 4305$ annotated images for training the ED endocardial DBN and epicardial DBN, and $75 \times 41 = 3075$ annotated images for training the ES endocardial DBN. The segmentation accuracy on training saturates with this augmented training data (i.e., adding more translated training images no longer improves the training results). The level set weights in (12.1) estimated with the training set for the endocardium segmentation are $\Delta t = 2$ (time step in the level set formulation), $\mu = \frac{0.24}{\Delta t} = 0.12$, $\lambda = 4$, $\alpha = -2$, $\beta = 0.02$, and $\gamma = 0.001$; and for the epicardium segmentation, we have $\Delta t = 2$, $\mu = \frac{0.24}{\Delta t} = 0.12$, $\lambda = 4$, $\alpha = -4$, $\beta = 0.015$, and $\gamma = 0.001$. The size of the sub-windows are set as M_{ROI} , M_{ENDO} , $M_{\text{EPI}} = 100$ (note that we found that the segmentation results are stable if M_{ROI} , M_{ENDO} , $M_{\text{EPI}} \in [80, 120]$).

For the **lung segmentation**, we use the training set for estimating the DBN and DRLS parameters and the validation set for the DBN model selection (similarly as for the ROI, ENDO and EPI DBN detailed above). This model selection estimated the following configuration for the DBN: two hidden layers, where each hidden layer has 1000 nodes and the input and segmentation layers have 1600 nodes. The initial guess ϕ_0 in Algorithm 1 is manually produced, so we show how the performance of our approach is affected by initial guesses of different accuracies, which are generated by random perturbations from the manual annotation. We denote the different initial guesses by the index $k \in \{1, 2, 3\}$, where $k = 1$ indicates the highest precision and $k = 3$ means the lowest precision initial guess. The estimation of the level set parameters is performed separately for each type of initial guess, and we achieve the following result: $\Delta t = 2$, $\mu = \frac{0.24}{\Delta t} = 0.12$, $\lambda = 2$, $\alpha = -3$, $\beta = 0$, $\gamma = 0.0005$ for $k = 1$; $\mu = 0.12$, $\lambda = 2$, $\alpha = -10$, $\beta = 0$, $\gamma = 0.003$ for $k = 2$; and $\mu = 0.12$, $\lambda = 2$, $\alpha = -15$, $\beta = 0$, $\gamma = 0.007$ for $k = 3$.

Note that for the level set weights in (12.1), we follow the recommendation by Li et al. [14] in defining the values for Δt , and μ (the recommendations are $\Delta t > 1$ and $\mu < \frac{0.25}{\Delta t}$), and for the inference procedure, the number of level set (DRLS) iterations is $T = 10$ (note that the segmentation results are stable if $T \in [5, 20]$).

12.4.3 Results of Each Stage of the Proposed Methodology

The role of each stage of the proposed **endocardium segmentation** is presented in Table 12.1. The “Initial endocardium segmentation” shows the result produced by the zero level set of ϕ_0 in (12.11) (i.e., the result from the ROI detection, followed by the initial endocardium segmentation). The “ENDO DBN alone” displays the accuracy results of the endocardium segmentation produced by the ENDO DBN (Sect. 12.3.2.2) alone. The “Model without DBN/shape prior” represents the energy functional in (12.3) with $\beta = \gamma = 0$, which effectively represents our model without the influence of the ENDO PRIOR and the ENDO DBN. Similarly the “Model without DBN” denotes the case where the functional in (12.3) has $\beta = 0$ (i.e., with no influence from ENDO DBN) and the “Model without shape prior” has $\gamma = 0$ (no influence from ENDO PRIOR). Finally, the “Proposed model” displays the result with all steps described in Sect. 12.3.2, and “Proposed model (semi)” represents our model using a manual initialisation instead of the automated initialisation described in Sect. 12.3.2.1. This manual initialisation consists of a circle, where the centre is the manual annotation centre of gravity and the radius is the minimum distance between the manual annotation and this centre. The proposed **epicardium segmentation** is assessed in Table 12.2, which shows the result of the “initial epicardium segmentation” explained in Sect. 12.3.3.1, and the result of the segmentation produced by the complete model described in Sect. 12.3.3.2 (labelled as “Proposed model”). We also show the result of the semi-automated epicardium segmentation with manual initialisation (defined in the same way as the manual initialisation above for the endocardium segmentation), labelled as “Proposed model (semi)”. Note that we do not show all steps in Table 12.2 because the results are similar to the initial epicardium segmentation.

Table 12.3 shows the results of our proposed methodology for **lung segmentation** with the different types of initial guesses. In this table, we also show the results when $\gamma = 0$, which is denoted by “Model without DBN” (this shows the influence of the DBN in the proposed methodology); and we also show the results for the initial guess, represented by “Initial guess only”.

12.4.4 Comparison with the State of the Art

Tables 12.4 and 12.5 show a comparison between our methodology (labelled “Proposed model”) and the state of the art for the **endocardium segmentation** problem,

Table 12.1 Quantitative experiments on the MICCAI 2009 challenge database [4] showing the **influence of each step** of the proposed methodology for the **endocardium segmentation**. Each cell is formatted as “mean (standard deviation) [min value-max value]”

Method	“Good” Percentage	Endocardium ADM	Endocardium APD
Test set (15 sequences)			
Proposed model (semi)	100(0)[100 – 100]	0.91(0.03)[0.83 – 0.95]	1.79(0.36)[1.28 – 2.75]
Proposed model	95.91(5.28)[84.62 – 100]	0.88(0.03)[0.82 – 0.93]	2.34(0.46)[1.62 – 3.24]
Model without shape prior	95.71(6.96)[78.95 – 100]	0.88(0.03)[0.83 – 0.93]	2.34(0.45)[1.67 – 3.14]
Model without DBN	85.89(18.00)[36.84 – 100]	0.84(0.04)[0.77 – 0.92]	2.77(0.58)[1.73 – 3.74]
Model without DBN/shape prior	84.49(18.31)[36.84 – 100]	0.84(0.04)[0.78 – 0.92]	2.78(0.58)[1.72 – 3.81]
ENDO DBN alone	18.31(19.46)[0 – 100]	0.87(0.02)[0.84 – 0.89]	3.81(0.64)[2.97 – 4.88]
Initial endocardium segmentation	85.18(15.83)[47.37 – 100]	0.85(0.04)[0.79 – 0.92]	2.81(0.47)[2.07 – 3.58]
Training set (15 sequences)			
Proposed model (semi)	100(0)[100 – 100]	0.91(0.03)[0.85 – 0.95]	1.63(0.40)[1.29 – 2.70]
Proposed model	97.22(3.16)[91.67 – 100]	0.88(0.05)[0.76 – 0.95]	2.13(0.46)[1.27 – 2.73]
Model without shape prior	97.42(4.63)[83.33 – 100]	0.88(0.04)[0.76 – 0.95]	2.14(0.43)[1.28 – 2.63]
Model without DBN	89.42(11.83)[61.11 – 100]	0.85(0.06)[0.71 – 0.93]	2.61(0.66)[1.74 – 3.65]
Model without DBN/shape prior	88.11(13.84)[50.00 – 100]	0.84(0.06)[0.70 – 0.93]	2.57(0.62)[1.72 – 3.53]
ENDO DBN alone	48.09(38.42)[0 – 100]	0.86(0.05)[0.73 – 0.90]	3.23(0.44)[2.70 – 4.05]
Initial endocardium segmentation	89.61(11.57)[55.56 – 100]	0.85(0.06)[0.71 – 0.93]	2.71(0.57)[1.78 – 3.49]

Table 12.2 Quantitative experiments on the MICCAI 2009 challenge database [4] compared different versions of the proposed methodology for the **epicardium segmentation**. Each cell is formatted as “mean (standard deviation) [min value–max value]”

Method	“Good” Percentage	Epicardium ADM	Epicardium APD
Test set (15 sequences)			
Proposed model (semi)	100(0)[100 – 100]	0.94(0.01)[0.92 – 0.97]	1.73(0.28)[1.16 – 2.17]
Proposed model	94.65(6.18)[85.71 – 100]	0.93(0.02)[0.88 – 0.96]	2.08(0.60)[1.27 – 3.74]
Initial epicardium segmentation	94.65(6.18)[85.71 – 100]	0.93(0.02)[0.88 – 0.96]	2.19(0.58)[1.32 – 3.68]
Training set (15 sequences)			
Proposed model (semi)	100.00(0.00)[100 – 100]	0.94(0.01)[0.91 – 0.96]	1.64(0.34)[1.17 – 2.47]
Proposed model	98.52(5.74)[77.78 – 100]	0.93(0.02)[0.89 – 0.96]	1.99(0.46)[1.35 – 3.13]
Initial epicardium segmentation	96.83(6.92)[77.78 – 100]	0.93(0.02)[0.89 – 0.95]	1.99(0.40)[1.46 – 3.14]

while Tables 12.6 and 12.7 display a similar comparison for the **epicardium segmentation** problem for different subsets of the MICCAI 2009 challenge databases [4]. Most of the approaches on that table are based on active contour models [17, 18, 21, 22, 36, 37], machine learning models [19, 23], or a combination of both models [38]. Furthermore, Tables 12.4, 12.5, 12.6 and 12.7 also show a semi-automated version of our method (labelled “Proposed model (semi)”) using the same initial guess described above in Sect. 12.4.3. Figure 12.9 shows a few **endocardium and epicardium segmentation** results produced by our approach for challenging cases, such as with images from apical and basal slice images and presenting papillary muscles and trabeculations.

Table 12.8 compares the results of our proposed **lung segmentation** method with the ones produced by the current state of the art on the JSRT database. The most competitive methods in that table [5, 25] are based on hybrid methods based on MRF and appearance/shape active models. Finally, Fig. 12.10 shows a few lung segmentation results using initial guess $k = 2$ on images of the test set.

12.5 Discussion and Conclusions

Table 12.1 clearly shows the importance of each stage of our proposed methodology for the endocardium segmentation problem. In particular, the initial endocardium segmentation is similar to the result from DRLS method [14] when the ENDO PRIOR

Table 12.3 Quantitative experiments on the JSRT database [16] showing the performance of the proposed **lung segmentation** method as a function of the initial guess used, where each cell is formattted as “mean (standard deviation) [min value–max value]”

Initial guess	Method	Ω^2	DSC	ACD
$k = 1$	Proposed model	0.985(0.003)[0.972 – 0.991]	0.992(0.002)[0.986 – 0.996]	1.075(0.065)[0.825 – 1.267]
	Model without DBN	0.984(0.003)[0.969 – 0.990]	0.992(0.002)[0.984 – 0.995]	1.376(0.221)[1.234 – 6.184]
	Initial guess only	0.955(0.006)[0.919 – 0.968]	0.977(0.003)[0.958 – 0.984]	1.392(0.006)[1.372 – 1.404]
$k = 2$	Proposed model	0.973(0.007)[0.944 – 0.985]	0.986(0.004)[0.971 – 0.993]	1.120(0.165)[0.628 – 1.916]
	Model without DBN	0.946(0.007)[0.910 – 0.961]	0.972(0.004)[0.953 – 0.980]	2.408(0.232)[0.021 – 7.232]
	Initial guess only	0.912(0.013)[0.844 – 0.935]	0.954(0.007)[0.916 – 0.967]	2.519(0.041)[2.369 – 2.621]
$k = 3$	Proposed model	0.948(0.012)[0.893 – 0.970]	0.973(0.006)[0.943 – 0.985]	1.852(0.286)[1.120 – 3.708]
	Model without DBN	0.866(0.018)[0.790 – 0.900]	0.928(0.010)[0.883 – 0.947]	4.695(0.276)[3.792 – 9.112]
	Initial guess only	0.828(0.024)[0.712 – 0.873]	0.906(0.014)[0.832 – 0.932]	4.936(0.105)[4.391 – 5.200]

Table 12.4 Quantitative experiments on the **training and test sets** of the MICCAI 2009 challenge databases [4] comparing the performance of our proposed approach with the state of the art on the **endocardium segmentation problem**. Notice that the methods are classified into fully or semi-automated. The cell formatting is the same as in Table 12.1, but note that ‘?’ means that the result is not available in the literature. The top performance for each measure and dataset is highlighted

Method	“Good” Percentage	Endocardium ADM	Endocardium APD
Test set (15 sequences)			
Semi-automated			
Proposed model (semi)			
[31]	96.58(9.58)[63.15 – 100]	0.89(0.03)[0.83 – 0.93]	2.22(0.46)[1.69 – 3.30]
[18]	?	0.89(0.04)[?–?]	2.10(0.44)[?–?]
Fully automated			
Proposed model	95.91(5.28)[84.62 – 100]	0.88(0.03)[0.82 – 0.93]	2.34(0.46)[1.62 – 3.24]
[21]	94.33(9.93)[62.00 – 100]	0.88(0.03)[0.84 – 0.94]	2.44(0.62)[1.36 – 3.68]
[23]	86.47(11.00)[68.4 – 100]	0.89(0.03)[0.82 – 0.94]	2.29(0.57)[1.67 – 3.93]
[17]	72.45(19.52)[42.11 – 100]	0.89(0.03)[0.84 – 0.94]	2.07(0.61)[1.32 – 3.77]
[37]	?	0.86(0.04)[?–?]	?
[19]	?	0.81(?)[?–?]	?
Training set (15 sequences)			
Semi-automated			
Proposed model (semi)	100(0)[100 – 100]	0.91(0.03)[0.85 – 0.95]	1.63(0.40)[1.29 – 2.70]
[31]	98.45(3.11)[91.66 – 100]	0.90(0.03)[0.84 – 0.94]	1.96(0.35)[1.43 – 2.55]
[18]	?	0.90(0.04)[?–?]	2.03(0.34)[?–?]
Fully automated			
Proposed model	97.22(3.16)[91.67 – 100]	0.88(0.05)[0.76 – 0.95]	2.13(0.46)[1.27 – 2.73]
[21]	96.93(7.59)[72 – 100]	0.88(0.06)[0.75 – 0.95]	2.09(0.53)[1.35 – 3.23]

and ENDO DBN terms are not used (row “Model without DBN/shape prior”). The introduction of shape prior (see row “Model without DBN”) provides a slightly improvement to the initial segmentation, but it is not a significant change; therefore we could removed it from the framework in order to obtain small gains in terms of efficiency (if needed). The largest gain in terms of accuracy comes from the introduction of ENDO DBN (see row “Model without shape prior”), but note that ENDO DBN alone is not competitive, which implies that the results produced by ENDO DBN complements well the results from DRLS. The presence of all terms together, shows that our “Proposed model” produces better segmentation results than the DRLS and DBN methods. Also, notice the relative small differences between the training and testing segmentation results, which indicates good generalisation

Table 12.5 Quantitative experiments on the **online and full sets** of the MICCAI 2009 challenge databases [4] comparing the performance of our proposed approach with the state of the art on the **endocardium segmentation problem**. Notice that the methods are classified into fully or semi-automated. The cell formatting is the same as in Table 12.1, but note that ‘?’ means that the result is not available in the literature. The top performance for each measure and dataset is highlighted

Method	“Good” Percentage	Endocardium ADM	Endocardium APD
Online set (15 sequences)			
Semi-automated			
Proposed model (semi)	100(0)[100 – 100]	0.91(0.03)[0.85 – 0.96]	1.78(0.49)[1.17 – 3.15]
[31]	98.71(3.66)[86.66 – 100]	0.90(0.04)[0.83 – 0.95]	2.04(0.35)[1.53 – 2.67]
Fully automated			
Proposed model	90.54(14.40)[46.67 – 100]	0.89(0.03)[0.82 – 0.94]	2.17(0.46)[1.62 – 3.46]
Full set (45 sequences)			
Semi-automated			
Proposed model (semi)	100(0)[100 – 100]	0.91(0.03)[0.83 – 0.96]	1.73(0.31)[1.17 – 3.15]
[31]	97.91(6.18)[63.15 – 100]	0.90(0.03)[0.83 – 0.95]	2.08(0.40)[1.43 – 3.30]
[22]	91.00(8.00)[61 – 100]	0.89(0.04)[0.80 – 0.96]	1.94(0.42)[1.47 – 3.03]
Fully automated			
Proposed model	94.55(9.31)[46.67 – 100]	0.88(0.04)[0.76 – 0.95]	2.22(0.46)[01.27 – 3.46]
[22]	80.00(16.00)[29 – 100]	0.86(0.05)[0.72 – 0.94]	2.44(0.56)[1.31 – 4.20]
[38]	91.06(9.42)[?–?]	0.89(0.03)[?–?]	2.24(0.40)[?–?]
[36]	79.20(19.00)[?–?]	0.89(0.04)[?–?]	2.16(0.46)[?–?]

capabilities of our method (even with the relatively small training set of the MICCAI 2009 challenge database [4]). Finally, by using a manual initialisation, we obtain the best segmentation results in the field.

For the epicardium segmentation problem, Table 12.2 shows that the initial segmentation produces a result that is close to the final segmentation produced by our proposed model. This means that the EPI DBN provides a improvement that is not quite significant. Also note that the use of manual initialisation shows the best result in the field, similarly to the endocardium segmentation. Finally, one can question the need for two separate DBN models (i.e., ENDO and EPI DBNs) given their appearance similarities. The main reason for the use of these two models lies in the empirical evidence that they produce more accurate segmentation results, as shown in Tables 12.4 and 12.5, where the rows labelled by **Proposed model (semi)** show the results with the two separate DBNs, while the rows labelled by [31] display results using a single classifier.

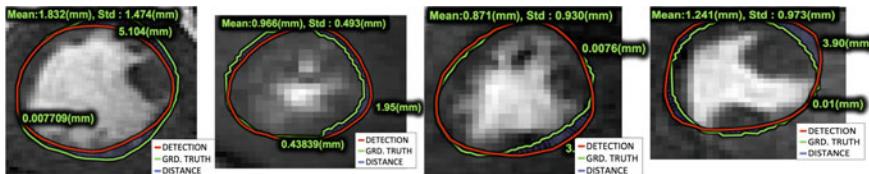
Table 12.6 Quantitative experiments on the **training and test sets** of the MICCAI 2009 challenge databases [4] comparing the performance of our proposed approach with the state of the art on the **epicardium segmentation problem**. Notice that the methods are classified into fully or semi-automated. The cell formatting is the same as in Table 12.1, but note that ‘?’ means that the result is not available in the literature. The top performance for each measure and dataset is highlighted

Method	“Good” Percentage	Epicardium ADM	Epicardium APD
Test set (15 sequences)			
Semi-automated			
Proposed model (semi)	100(0)[100 – 100]	0.94(0.01)[0.92 – 0.97]	1.73(0.28)[1.16 – 2.17]
[18]	?	0.94(0.01)[?–?]	1.95(0.34)[?–?]
Fully automated			
Proposed model	94.65(6.18)[85.71 – 100]	0.93(0.02)[0.88 – 0.96]	2.08(0.60)[1.27 – 3.74]
[21]	95.60(6.90)[80.00 – 100]	0.93(0.02)[0.90 – 0.96]	2.05(0.59)[1.28 – 3.29]
[23]	94.20(7.00)[80.00 – 100]	0.93(0.01)[0.90 – 0.96]	2.28(0.39)[1.57 – 2.98]
[17]	81.11(13.95)[57.14 – 100]	0.94(0.02)[0.90 – 0.97]	1.91(0.63)[1.06 – 3.26]
Training set (15 sequences)			
Semi-automated			
Proposed model (semi)	100.00(0.00)[100 – 100]	0.94(0.01)[0.91 – 0.96]	1.64(0.34)[1.17 – 2.47]
[18]	?	0.93(0.02)[?–?]	2.28(0.42)[?–?]
Fully automated			
Proposed model	98.52(5.74)[77.78 – 100]	0.93(0.02)[0.88 – 0.96]	1.99(0.46)[1.35 – 3.13]
[21]	99.07(3.61)[86.00 – 100]	0.93(0.01)[0.91 – 0.95]	1.88(0.40)[1.20 – 2.55]

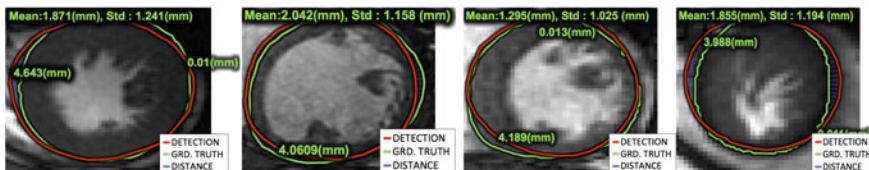
The comparison with the state of the art for the problem of endocardium segmentation (Tables 12.4 and 12.5) and the epicardium segmentation (Tables 12.6 and 12.7) shows that the proposed approach has the best results for the semi-automated segmentation problem. When considering the fully automated segmentation, the results from the proposed method are comparable to the ones by [21], which is regarded as the current state of the art by a recent review paper by Petitjean et al. [3]. In regards to the “Good” percentage measure, our approach shows better results than the other methods; whilst in terms of ADM and ADP, our approach shows comparable results. When considering the epicardium segmentation, the results of our method are comparable to the one by Jolly’s approach [21], but better than all others. It is important to note that although some approaches are more accurate in terms of APD or ADM [17], they also present low values for “Good” percentage, which means that these methods also produce a large number of segmentations with APD larger than 5 mm, but the few ones that survive the “Good” percentage test are reasonably accurate. We also note the relatively worse performance of the fully automated approach compared to semi-automated segmentation (not only for our proposed method, but other methods

Table 12.7 Quantitative experiments on the **online and full sets** of the MICCAI 2009 challenge databases [4] comparing the performance of our proposed approach with the state of the art on the **epicardium segmentation problem**. Notice that the methods are classified into fully or semi-automated. The cell formatting is the same as in Table 12.1, but note that ‘?’ means that the result is not available in the literature. The top performance for each measure and dataset is highlighted

Method	“Good” Percentage	Epicardium ADM	Epicardium APD
Online set (15 sequences)			
Semi-automated			
Proposed model (semi)	100.00(0.00)[100 – 100]	0.94(0.02)[0.88 – 0.96]	1.90(0.53)[1.22 – 3.16]
Fully automated			
Proposed model	84.32(23.45)[12.50 – 100]	0.93(0.03)[0.84 – 0.95]	2.05(0.61)[1.39 – 3.63]
Full set (45 sequences)			
Semi-automated			
Proposed model (semi)	100(0)[100 – 100]	0.94(0.02)[0.88 – 0.97]	1.76(0.40)[1.16 – 3.16]
[22]	91.00(10.00)[70 – 100]	0.92(0.02)[0.84 – 0.95]	2.38(0.57)[1.28 – 3.79]
Fully automated			
Proposed model	92.49(15.31)[12.50 – 100]	0.93(0.02)[0.84 – 0.96]	2.04(0.55)[1.27 – 3.70]
[22]	71.00(26.00)[0 – 100]	0.91(0.03)[0.81 – 0.96]	2.80(0.71)[1.37 – 4.88]
[38]	91.21(8.52)[?–?]	0.94(0.02)[?–?]	2.21(0.45)[?–?]
[36]	83.90(16.80)[?–?]	0.93(0.02)[?–?]	2.22(0.43)[?–?]



(a) Results of endocardium segmentation on the test set



(b) Results of epicardium segmentation on the test set

Fig. 12.9 Epicardium and endocardium segmentation results with challenging cases, such as images from apical and basal slice images and presenting papillary muscles and trabeculations. The red contour denotes the automated detection, and green shows the manual annotation. For more results, please see the supplementary material

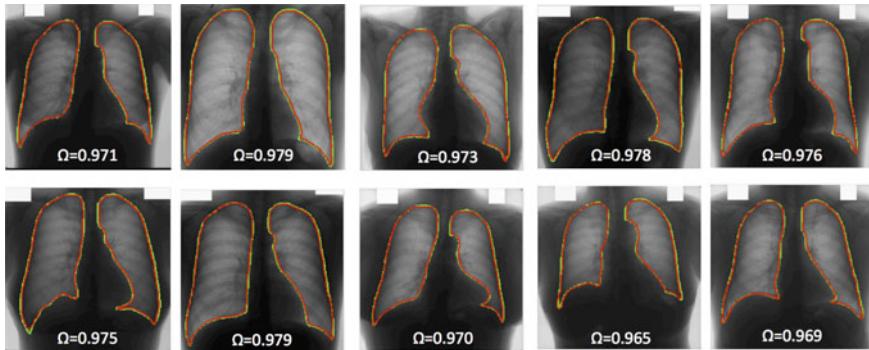


Fig. 12.10 Lung segmentation results with initial guess $k = 2$. The *green* contour shows expert annotation and the *red* illustrates the final result

in the literature), which implies that there is still an opportunity to improve further the accuracy of the initial endocardium and epicardium segmentations. In terms of running time, the system developed based on the proposed methodology runs on average in 175 ± 35 s for the endocardium segmentation and 119 ± 20 s for the epicardium segmentation using a *non-optimised MATLAB program running on a standard computer* (Intel(R) Core(TM) i5-2500k 3.30GHz CPU with 8GB RAM), which is slower or comparable to other approaches that run between one minute [21–23] and three minutes [17, 38].

For the lung segmentation problem, Table 12.3 shows that the proposed model always improve over the initial guess, but this improvement is more obvious with poorer initial guesses (see results of “Initial guess only” and “Proposed mode” for $k = 3$). Another important observation is that the DRLS always improve over the initial guess, and the introduction of the DBN model improves the initial DRLS result. An obvious question is the reason for the absence of the shape prior model, and the reason is that we did not notice any empirical improvement. The comparison with the state of the art in Table 12.8 shows that with the manual initial guesses $k \in \{1, 2\}$, our proposed approach produces the best results in the field. Additionally, using a similar Matlab code running on the same computer introduced above, our method runs on average in 20.68 seconds/image, which is comparable to the result by Candemir et al. [5], who report a running time of between 20 and 25 seconds/image using the same input resolution and similar computer configuration.

There are several points that can be explored in order to improve the results above. For the endocardium and epicardium segmentation, we can run the method over the whole volume and use a 3-D shape model to constrain the search process. We can also use a motion model to constrain the segmentation process. More complex DBN models can be trained when new training sets become available. Finally, we can decrease the running time of our approach by parallelising the segmentation processes since the segmentation of each slice is done independently of all others (roughly this means that we can in principle make our approach 10 times faster).

Table 12.8 Quantitative experiments on the JSRT database [16] comparing our results with the state of the art on the same database, sorted from best (top) to worst (bottom). The symbol ‘?’ indicates that the result is not available

Method	Ω	DSC	ACD
Proposed model, $k = 1$	0.985(0.003)[0.972 – 0.991]	0.992(0.002)[0.986 – 0.996]	1.075(0.065)[0.825 – 1.267]
Proposed model, $k = 2$	0.973(0.007)[0.944 – 0.985]	0.986(0.004)[0.971 – 0.993]	1.120(0.165)[0.628 – 1.916]
[5]	0.954(0.015)[?–?]	0.967(0.008)[?–?]	1.321(0.316)[?–?]
[25]	0.949(0.020)[0.818 – 0.978]	?(?)[?–?]	1.62(0.66)[0.95 – 7.72]
Proposed model, $k = 3$	0.948(0.012)[0.893 – 0.970]	0.973(0.006)[0.943 – 0.985]	1.852(0.286)[1.120 – 3.708]
[25]	0.945(0.022)[0.823 – 0.972]	?(?)[?–?]	1.61(0.80)[0.83 – 8.34]
[39]	0.940(0.053)[?–?]	?(?)[?–?]	2.46(2.06)[?–?]
[25]	0.938(0.027)[0.823 – 0.968]	?(?)[?–?]	3.25(2.65)[0.93 – 15.59]
[25]	0.934(0.037)[0.706 – 0.968]	?(?)[?–?]	2.08(1.40)[0.91 – 11.57]
[40]	0.930(?)[?–?]	?(?)[?–?]	?(?)[?–?]
[25]	0.922(0.029)[0.718 – 0.961]	?(?)[?–?]	2.39(1.07)[1.15 – 12.09]
[41]	0.907(0.033)[?–?]	?(?)[?–?]	?(?)[?–?]

For the lung segmentation, we plan to introduce an automated initial guess with a method similar to the one proposed by Candemir et al. [5]. Furthermore, we plan to extend this method to other segmentation problems.

In this chapter, we have presented a methodology that combines level set method and structured output deep belief network models. We show the functionality of the proposed approach in two different problems: the segmentation of endocardium and epicardium from cine MR and the segmentation of lungs from chest radiographs. In both problems, we show extensive experiments that show the functionality of our approach, and they also show that our approach produces the current state-of-the-art segmentation results.

Acknowledgements This work was partially supported by the Australian Research Council’s Discovery Projects funding scheme (project DP140102794). Tuan Anh Ngo acknowledges the support of the 322 Program - Vietnam International Education Development, Ministry of Education and Training (VIED-MOET).

References

1. Ngo T, Carneiro G (2014) Fully automated non-rigid segmentation with distance regularized level set evolution initialized and constrained by deep-structured inference. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3118–3125
2. Ngo TA, Carneiro G (2015) Lung segmentation in chest radiographs using distance regularized level set and deep-structured learning and inference. In: 2015 IEEE international conference on image processing (ICIP). IEEE, pp 2140–2143
3. Petitjean C, Dacher J-N (2011) A review of segmentation methods in short axis cardiac mr images. *Med Image Anal* 15(2):169–184
4. Radau P, Lu Y, Connelly K, Paul G, Dick A, Wright G (2009) Evaluation framework for algorithms segmenting short axis cardiac mri. In: MIDAS J. cardiac MR left ventricle segmentation challenge
5. Candemir S, Jaeger S, Musco J, Xue Z, Karargyris A, Antani S, Thoma G, Palaniappan K (2014) Lung segmentation in chest radiographs using anatomical atlases with non-rigid registration
6. Carrascal FM, Carreira JM, Souto M, Tahoces PG, Gómez L, Vidal JJ (1998) Automatic calculation of total lung capacity from automatically traced lung boundaries in postero-anterior and lateral digital chest radiographs. *Med Phys* 25(7):1118–1131
7. Kass M, Witkin A, Terzopoulos D (1988) Snakes: active contour models. *Int J Comput Vision* 1(4):321–331
8. Osher S, Sethian JA (1988) Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *J Comput Phys* 79(1):12–49
9. Cootes TF, Taylor CJ, Cooper DH, Graham J (1995) Active shape models-their training and application. *Comput Vis Image Underst* 61(1):38–59
10. Georgescu B, Zhou XS, Comaniciu D, Gupta A (2005) Databased-guided segmentation of anatomical structures with complex appearance. In: CVPR
11. Cobzas D, Schmidt M (2009) Increased discrimination in level set methods with embedded conditional random fields. In: IEEE conference on Computer vision and pattern recognition, 2009. CVPR 2009. IEEE, pp 328–335
12. Huang R, Pavlovic V, Metaxas DN (2004) A graphical model framework for coupling mrf's and deformable models. In: Proceedings of the 2004 IEEE computer society conference on computer vision and pattern recognition, 2004. CVPR 2004, vol. 2. IEEE, pp II–739
13. Tsechpenakis G, Metaxas DN (2007) Crf-driven implicit deformable model. In: IEEE conference on computer vision and pattern recognition, 2007. CVPR'07. IEEE, pp 1–8
14. Li C, Xu C, Gui C, Fox MD (2010) Distance regularized level set evolution and its application to image segmentation. *IEEE Trans Image Process* 19(12):3243–3254
15. Hinton G, Salakhutdinov R (2006) Reducing the dimensionality of data with neural networks. *Science* 313(5786):504–507
16. Shiraiishi J, Katsuragawa S, Ikezoe J, Matsumoto T, Kobayashi T, Komatsu K-I, Matsui M, Fujita H, Kodera Y, Doi K (2000) Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologists' detection of pulmonary nodules. *Am J Roentgenol* 174(1):71–74
17. Lu Y, Radau P, Connelly K, Dick A, Wright G (2009) Automatic image-driven segmentation of left ventricle in cardiac cine mri. In: The MIDAS journal, vol 49
18. Huang S, Liu J, Lee L, Venkatesh S, Teo L, Au C, Nowinski W (2009) Segmentation of the left ventricle from cine mr images using a comprehensive approach. In: The MIDAS journal, vol. 49
19. O'Brien S, Ghita O, Whelan P (2009) Segmenting the left ventricle in 3d using a coupled asm and a learned non-rigid spatial model. In: The MIDAS journal, vol 49
20. Schaefer J, Casta C, Pousin J, Clarysse P (2010) A dynamic elastic model for segmentation and tracking of the heart in mr image sequences. *Med Image Anal* 14(6):738–749
21. Jolly M (2009) Fully automatic left ventricle segmentation in cardiac cine mr images using registration and minimum surfaces. In: The MIDAS journal, vol 49

22. Constantinides C, Roullot E, Lefort M, Frouin F (2012) Fully automated segmentation of the left ventricle applied to cine mr images: description and results on a database of 45 subjects. In: Engineering in medicine and biology society (EMBC) (2012) annual international conference of the IEEE. IEEE, pp 3207–3210
23. Wijnhout J, Hendriksen D, Assen H, der Geest R (2009) Lv challenge lkeb contribution: fully automated myocardial contour detection. In: The MIDAS journal, vol 43
24. Van Ginneken B, Frangi AF, Staal JJ, ter Haar Romeny BM, Viergever MA (2002) Active shape model segmentation with optimal features. *IEEE Trans Med Imaging* 21(8), 924–933
25. Van Ginneken B, Stegmann MB, Loog M (2006) Segmentation of anatomical structures in chest radiographs using supervised methods: a comparative study on a public database. *Med Image Anal* 10(1):19–40
26. BakIr G (2007) Predicting structured data. MIT press, Cambridge
27. Tsochantaridis I, Joachims T, Hofmann T, Altun Y, Singer Y (2005) Large margin methods for structured and interdependent output variables. *J Mach Learn Res* 6(9): 1453–1484
28. Collins M (2002) Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In: Proceedings of the ACL-02 conference on empirical methods in natural language processing-volume 10. Association for Computational Linguistics, pp 1–8
29. Fasel I, Berry J (2010) Deep belief networks for real-time extraction of tongue contours from ultrasound during speech. In: 2010 20th international conference on pattern recognition (ICPR). IEEE, pp 1493–1496
30. Farabet C, Couprie C, Najman L, LeCun Y (2012) Scene parsing with multiscale feature learning, purity trees, and optimal covers. [arXiv:1202.2160](https://arxiv.org/abs/1202.2160)
31. Ngo TA, Carneiro G (2013) Left ventricle segmentation from cardiac mri combining level set methods with deep belief networks. In: 2013 20th IEEE international conference on image processing (ICIP). IEEE, pp 695–699
32. Ngo TA, Carneiro G (2014) Fully automated non-rigid segmentation with distance regularized level set evolution initialized and constrained by deep-structured inference. In: 2013 IEEE conference on computer vision and pattern recognition (CVPR). IEEE
33. Cremers D, Osher SJ, Soatto S (2006) Kernel density estimation and intrinsic alignment for shape priors in level set segmentation. *Int J Comput Vision* 69(3):335–351
34. Otsu N (1975) A threshold selection method from gray-level histograms. *Automatica* 11(285–296):23–27
35. Canny J (1986) A computational approach to edge detection. *IEEE Trans Pattern Anal Mach Intell* 6:679–698
36. Huang S, Liu J, Lee LC, Venkatesh SK, San Teo LL, Au C, Nowinski WL (2011) An image-based comprehensive approach for automatic segmentation of left ventricle from cardiac short axis cine mr images. *J Digit Imaging* 24(4):598–608
37. Marak L, Cousty J, Najman L, Talbot H et al (2009) 4d morphological segmentation and the miccai lv-segmentation grand challenge. In: MICCAI 2009 workshop on cardiac MR left ventricle segmentation challenge, no 1, pp 1–8
38. Hu H, Liu H, Gao Z, Huang L (2012) Hybrid segmentation of left ventricle in cardiac mri using gaussian-mixture model and region restricted dynamic programming. In: Magnetic resonance imaging
39. Dawoud A (2011) Lung segmentation in chest radiographs by fusing shape information in iterative thresholding. *IET Comput Vision* 5(3):185–190
40. Seghers D, Loeckx D, Maes F, Vandermeulen D, Suetens P (2007) Minimal shape and intensity cost path segmentation. *IEEE Trans Med Imaging* 26(8):1115–1129
41. Yu T, Luo J, Ahuja N (2005) Shape regularized active contour using iterative global search and local optimization. In: IEEE computer society conference on computer vision and pattern recognition, 2005. CVPR 2005, vol 2. IEEE, pp 655–662

Chapter 13

Combining Deep Learning and Structured Prediction for Segmenting Masses in Mammograms

Neeraj Dhungel, Gustavo Carneiro and Andrew P. Bradley

Abstract The segmentation of masses from mammogram is a challenging problem because of their variability in terms of shape, appearance and size, and the low signal-to-noise ratio of their appearance. We address this problem with structured output prediction models that use potential functions based on deep convolution neural network (CNN) and deep belief network (DBN). The two types of structured output prediction models that we study in this work are the conditional random field (CRF) and structured support vector machines (SSVM). The label inference for CRF is based on tree re-weighted belief propagation (TRW) and training is achieved with the truncated fitting algorithm; whilst for the SSVM model, inference is based upon graph cuts and training depends on a max-margin optimization. We compare the results produced by our proposed models using the publicly available mammogram datasets DDSM-BCRP and INbreast, where the main conclusion is that both models produce results of similar accuracy, but the CRF model shows faster training and inference. Finally, when compared to the current state of the art in both datasets, the proposed CRF and SSVM models show superior segmentation accuracy.

13.1 Introduction

Statistical findings published by World Health Organization (WHO) [1] reveal that 23% of all diagnosed cancers and 14% of all cancer related deaths among women are due to breast cancer. These numbers show that breast cancer is one of the major

N. Dhungel (✉) · G. Carneiro
Australian Centre for Visual Technologies, The University of Adelaide,
Adelaide, SA, Australia
e-mail: neeraj.dhungel@adelaide.edu.au

G. Carneiro
e-mail: gustavo.carneiro@adelaide.edu.au

A.P. Bradley
School of Information Technology and Electrical Engineering,
The University of Queensland, St Lucia, QLD, Australia
e-mail: a.bradley@itee.uq.edu.au

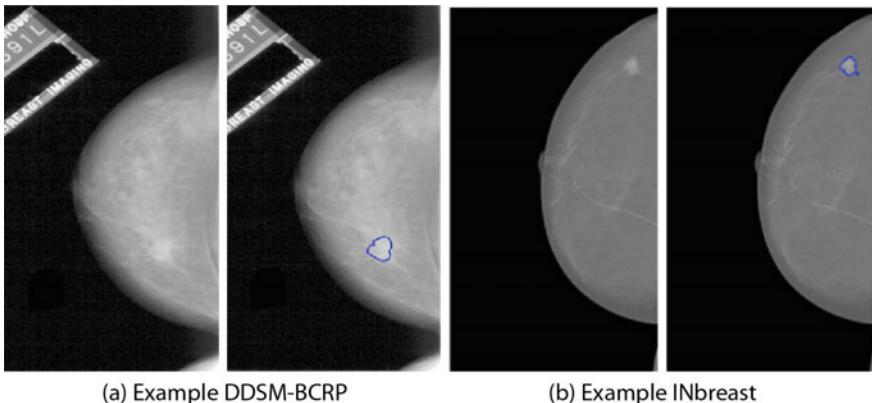


Fig. 13.1 Examples from INbreast [6] and DDSM-BCRP [7] databases with *blue* contour denoting the mass lesion with the *blue* contour

diseases affecting the lives of many women across the globe. One of the keys to reduce these numbers is the early detection of this disease, which is task that is mostly based on mammography screening. An important activity involved in this screening process is the detection and classification of breast masses, which is difficult because of the variable size, shape and appearance of masses [2] and their low signal-to-noise ratio (see Fig. 13.1). In this work, we focus on the problem of accurate mass segmentation because we assume that such precise segmentation is important for the subsequent mass classification task [3, 4]. In clinical practice, the task of detecting and segmenting masses from mammograms typically consists of a manual process performed by radiologists. This process can introduce variability depending on the radiologist's expertise and the number of mammograms to be analysed at one sitting, which can reduce the efficacy of the screening process. In a recent study [5], it has been shown that there is a clear trade-off between sensitivity (Se) and specificity (Sp) in manual interpretation, with a median Se of 84% and Sp of 91%.

Regardless of the development of numerous breast mass segmentation techniques, computer-aided diagnosis (CAD) systems, which depend on accurate breast mass segmentation methods, are not widely used in clinical practice. In fact, it has been observed that the use of CAD systems can reduce screening accuracy by increasing the rate of biopsies without improving the detection of invasive breast cancer [8]. We believe that one of the reasons is the lack of an easily reproducible and reliable assessment mechanism that provides a clear comparison between competing methodologies, which can lead to a better informed decision process related to the selection of appropriate algorithms for CAD systems. We have addressed this issue in previous versions of this work [9, 10], where we propose quantitatively comparison mechanisms on the publicly available databases DDSM-BCRP [7] and INbreast dataset [6]. Another reason for the relatively poor performance of most of the currently available breast mass segmentation methods lies in their reliance on more

traditional image processing and segmentation techniques, such as active contours, which typically produce sub-optimal results due to their non-convex cost functions. Differently from these methods, our approach is based on a machine learning technique that estimates optimal models directly from annotated data, and for this reason our approach has the potential to deliver improved segmentation accuracy, a result previously demonstrated in other medical image analysis problems [11].

In this work, we propose a new approach for segmenting breast masses from mammograms using two types of structured output prediction models: (1) conditional random field (CRF) [10, 12] and (2) structural support vector machine (SSVM) [9, 13]. Our main contribution is related to the introduction of powerful deep learning networks into the CRF and SSVM models above, based on the deep convolutional neural network (CNN) [14, 15] and the deep neural network (DBN) [16]. These deep learning architectures are able to extract image features in a fully automated manner, instead of being hand-crafted. In addition, these CNNs and DBNs have produced state-of-the-art results in several computer vision problems [14, 17], and we believe that these methodologies have the potential to produce competitive results in mass segmentation from mammography. The CRF model uses tree re-weighted belief propagation [18] for inference and truncated fitting for training [12], whilst SSVM performs label inference with graph cuts [19] and the parameters learning with the cutting plane algorithm [13, 20]. Given that these training algorithms learn all parameters for the structured output prediction models using the manually annotated training data and that we do not make any assumptions about the shape and appearance of masses, we believe that our proposed approach is capable of modelling in a robust manner the shape and appearance variations of masses encountered in the training data if enough annotated training data is available. We test our proposed methodologies on the publicly available datasets INbreast [6] and DDSM-BCRP [7], and our methodologies produce state-of-the-art results in terms of accuracy and running time. Moreover, comparing the CRF and SSVM models, we note that they produce comparable results in terms of segmentation accuracy, but the CRF model is more efficient in terms of training and testing.

13.2 Literature Review

Currently, the majority of the methodologies developed for the problem of segmenting masses from mammograms are based on statistical thresholding, dynamic programming models, morphological operators and active contour models. A statistical thresholding method that distinguishes pixels inside the mass area from those outside has been developed by Catarious et al. [21]. Although relatively successful, the main drawback of this type of approach is that it is not robust to low contrast images [3]. Song et al. [22] have extended this model with a statistical classifier based on edge gradient, pixel intensity and shape characteristics, where the segmentation is found by estimating the minimum cut of a graph representation of the image using dynamic programming. Similar dynamic programming models have also been applied by

Timp et al. [23], Dominguez et al. [24] and Yu et al. [25]. These approaches are similar to our proposed structured output prediction models, with the exception that they do not use structured learning to estimate the weights of the potential functions, which generally leads to sub-optimal performance. Morphological operators, such as the watershed method [26] or region growing [4], have also been explored for the mass segmentation problem, but these operators have been shown to be rather limited in providing sufficiently accurate results mainly because they only explore semi-local grey-level distributions without considering higher level information (e.g., shape model).

Active contour models are probably the most explored methodology for breast mass segmentation. The most accurate model reported in the field is the one proposed by Rahmati et al. [3], which is a level set method based on the maximum likelihood segmentation without edges that is particularly well adapted to noisy images with weak boundaries. Several other papers also propose mass segmentation methods based on standard active contour models [27–31]. The major drawback of active contour models lies in their need of a good initialization for the inference process due to the usual non-convexity of the energy function. Moreover, the weights of the terms forming the energy function of the active contour models are usually arbitrarily defined, or estimated via a cross-validation process that generally does not produce an optimal estimation of these weights.

13.3 Methodology

We start this section with an explanation of the learning process of our structured output prediction model [32]. Assume that we have an annotated dataset \mathcal{D} containing images of the region of interest (ROI) of the mass, represented by $\mathbf{x} : \Omega \rightarrow \mathbf{R}$ ($\Omega \in \mathbf{R}^2$), and the respective manually provided segmentation mask $\mathbf{y} : \Omega \rightarrow \{-1, +1\}$, where $\mathcal{D} = (\mathbf{x}, \mathbf{y})_{i=1}^{|\mathcal{D}|}$. Also assume that the parameter of our structured output prediction model is denoted by θ and the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ links the image \mathbf{x} and labels \mathbf{y} , where \mathcal{V} represents the set of graph nodes and \mathcal{E} , the set of edges. The process of learning the parameter of our structured prediction model is done via the minimization of the following empirical loss function [32]:

$$\theta^* = \arg \min_{\theta} \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} \ell(\mathbf{x}_i, \mathbf{y}_i, \theta), \quad (13.1)$$

where $\ell(\mathbf{x}, \mathbf{y}, \theta)$ is a continuous and convex loss function being minimized that defines the structured model. We use CRF and SSVM formulations for solving (13.1), which are explained in detail in Sects. 13.3.1 and 13.3.2, respectively, and we explain the potential functions used for both models in Sect. 13.3.3. In particular, the CRF formulation uses the loss

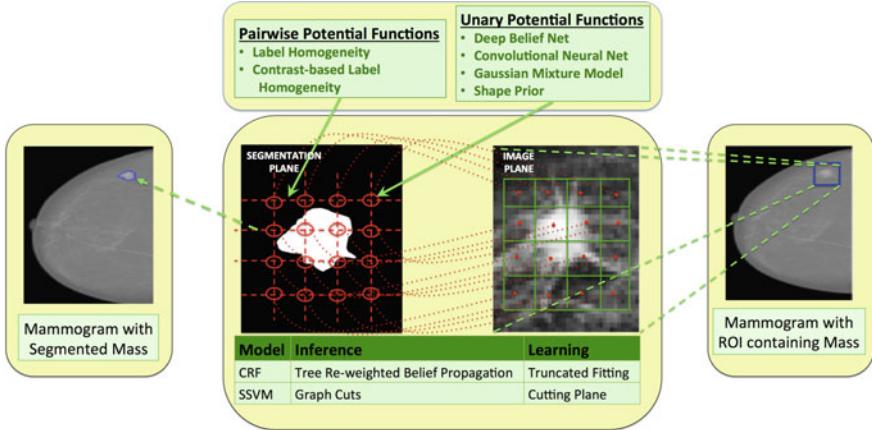


Fig. 13.2 The proposed structured output prediction models with a list of unary and pairwise potential functions for mass segmentation in mammograms, including the deep learning networks

$$\ell(\mathbf{x}_i, \mathbf{y}_i, \theta) = A(\mathbf{x}_i, \theta) - E(\mathbf{y}_i, \mathbf{x}_i; \theta), \quad (13.2)$$

where $A(\mathbf{x}; \theta) = \log \sum_{\mathbf{y} \in \{-1, +1\}^{|\Omega| \times |\Omega|}} \exp \{E(\mathbf{y}, \mathbf{x}; \theta)\}$ is the log-partition function that ensures normalization, and

$$E(\mathbf{y}, \mathbf{x}; \theta) = \sum_{k=1}^K \sum_{i \in \mathcal{V}} \theta_{1,k} \psi^{(1,k)}(\mathbf{y}(i), \mathbf{x}) + \sum_{l=1}^L \sum_{i,j \in \mathcal{E}} \theta_{2,l} \psi^{(2,l)}(\mathbf{y}(i), \mathbf{y}(j), \mathbf{x}), \quad (13.3)$$

In (13.3), $\psi^{(1,k)}(\cdot, \cdot)$ denotes one of the K potential functions between label (segmentation plane in Fig. 13.2) and pixel (image plane in Fig. 13.2) nodes, $\psi^{(2,l)}(\cdot, \cdot, \cdot)$ denoting one of the L potential functions on the edges between label nodes, $\theta = [\theta_{1,1}, \dots, \theta_{1,K}, \theta_{2,1}, \dots, \theta_{2,L}]^\top \in \mathbf{R}^{K+L}$, and $\mathbf{y}(i)$ being the i^{th} component of vector \mathbf{y} . Similarly, the SSVM uses the following loss function

$$\ell(\mathbf{x}_i, \mathbf{y}_i, \theta) = \max_{\mathbf{y} \in \mathcal{Y}} (\Delta(\mathbf{y}_i, \mathbf{y}) + E(\mathbf{y}, \mathbf{x}_i; \theta) - E(\mathbf{y}_i, \mathbf{x}_i; \theta)), \quad (13.4)$$

where $\Delta(\mathbf{y}_i, \mathbf{y})$ represents the dissimilarity between \mathbf{y}_i and \mathbf{y} , which satisfies the conditions $\Delta(\mathbf{y}_i, \mathbf{y}) \geq 0$ for $\mathbf{y}_i \neq \mathbf{y}$ and $\Delta(\mathbf{y}_i, \mathbf{y}_i) = 0$.

13.3.1 Conditional Random Field (CRF)

The solution of (13.1) using the CRF loss function in (13.2) involves the computation of the log-partition function $A(\mathbf{x}; \theta)$. The tree re-weighted belief propagation

algorithm provides the following upper bound to this log-partition function [18]:

$$A(\mathbf{x}; \theta) = \max_{\boldsymbol{\mu} \in \mathcal{M}} \boldsymbol{\theta}^T \boldsymbol{\mu} + H(\boldsymbol{\mu}), \quad (13.5)$$

where $\mathcal{M} = \{\boldsymbol{\mu}' : \exists \theta, \boldsymbol{\mu}' = \boldsymbol{\mu}\}$ denotes the marginal polytope, $\boldsymbol{\mu} = \sum_{\mathbf{y} \in \{-1, +1\}^{|\mathcal{Q}| \times |\mathcal{Q}|}} P(\mathbf{y}|\mathbf{x}, \theta) f(\mathbf{y})$, with $f(\mathbf{y})$ denoting the set of indicator functions of possible configurations of each clique and variable in the graph [33] (as denoted in (13.3)), $P(\mathbf{y}|\mathbf{x}, \theta) = \exp\{E(\mathbf{y}, \mathbf{x}; \theta) - A(\mathbf{x}; \theta)\}$ indicating the conditional probability of the annotation \mathbf{y} given the image \mathbf{x} and parameters θ (where we assume that this conditional probability function belongs to the exponential family), and $H(\boldsymbol{\mu}) = -\sum_{\mathbf{y} \in \{-1, +1\}^{|\mathcal{Q}| \times |\mathcal{Q}|}} P(\mathbf{y}|\mathbf{x}; \theta) \log P(\mathbf{y}|\mathbf{x}, \theta)$ is the entropy. Note that for general graphs with cycles (such as the case in this paper), the marginal polytope \mathcal{M} is difficult to characterize and the entropy $H(\boldsymbol{\mu})$ is not tractable [12]. Tree re-weighted belief propagation (TRW) solves these issues by first replacing the marginal polytope with a superset $\mathcal{L} \supset \mathcal{M}$ that only accounts for the local constraints of the marginals, and then approximating the entropy calculation with an upper bound. Specifically,

$$\mathcal{L} = \{\boldsymbol{\mu} : \sum_{\mathbf{y}(c) \setminus \mathbf{y}(i)} \boldsymbol{\mu}(\mathbf{y}(c)) = \boldsymbol{\mu}(\mathbf{y}(i)), \sum_{\mathbf{y}(i)} \boldsymbol{\mu}(\mathbf{y}(i)) = 1\} \quad (13.6)$$

replaces \mathcal{M} in (13.5) and represents the local polytope (with $\boldsymbol{\mu}(\mathbf{y}(i)) = \sum_{\mathbf{y}'} P(\mathbf{y}'|\mathbf{x}, \theta) \delta(\mathbf{y}'(i) - \mathbf{y}(i))$ and $\delta(\cdot)$ denoting the Dirac delta function), c indexes a graph clique, and the entropy approximation (that replaces $H(\boldsymbol{\mu})$ in (13.5)) is defined by

$$\tilde{H}(\boldsymbol{\mu}) = \sum_{\mathbf{y}(i)} H(\boldsymbol{\mu}(\mathbf{y}(i))) - \sum_{\mathbf{y}(c)} \rho_c I(\boldsymbol{\mu}(\mathbf{y}(c))), \quad (13.7)$$

where $H(\boldsymbol{\mu}(\mathbf{y}(i))) = -\sum_{s(i)} \boldsymbol{\mu}(\mathbf{y}(i)) \log \boldsymbol{\mu}(\mathbf{y}(i))$ is the univariate entropy of variable $\mathbf{y}(i)$, $I(\boldsymbol{\mu}(\mathbf{y}(c))) = \sum_{\mathbf{y}(c)} \boldsymbol{\mu}(\mathbf{y}(c)) \log \frac{\boldsymbol{\mu}(\mathbf{y}(c))}{\prod_{i \in c} \boldsymbol{\mu}(\mathbf{y}(i))}$ is the mutual information of the cliques in our model, and ρ_c is a free parameter providing the upper bound on the entropy. Therefore, the estimation of $A(\mathbf{x}; \theta)$ and associated marginals in (13.5) is based on the following message-passing updates [12]:

$$\begin{aligned} m_c(\mathbf{y}(i)) &\propto \sum_{\mathbf{y}(c) \setminus \mathbf{y}(i)} \exp \left\{ \frac{1}{\rho_c} \psi_c(\mathbf{y}(i), \mathbf{y}(j); \theta) \right\} \\ &\quad \prod_{j \in c \setminus i} \exp \left\{ \frac{1}{\rho_c} \psi_i(\mathbf{y}(i), \mathbf{x}; \theta) \right\} \frac{\prod_{d: j \in d} m_d(s(j))^{\rho_d}}{m_c(s(j))}, \end{aligned} \quad (13.8)$$

where $\phi_i(\mathbf{y}(i), \mathbf{x}; \theta) = \sum_{k=1}^K w_{1,k} \psi^{(1,k)}(\mathbf{y}(i), \mathbf{x})$ and $\psi_c(\mathbf{y}(i), \mathbf{y}(j); \theta) = \sum_{l=1}^L w_{2,l} \phi^{(2,l)}(\mathbf{y}(i), \mathbf{y}(j), \mathbf{x})$ (see (13.3)). Once the message-passing algorithm converges [18], the beliefs for the associated marginals are written as:

$$\begin{aligned}\mu_c(\mathbf{y}(c)) &\propto \frac{1}{\rho_c} \psi_c(\mathbf{y}(i), \mathbf{y}(j)) \prod_{i \in c} \psi_i(\mathbf{y}(i), \mathbf{x}; \theta) \frac{\prod_{d:j \in d} m_d(\mathbf{y}(j))^{\rho_d}}{m_c(\mathbf{y}(i))} \\ \mu_i(\mathbf{y}_i) &\propto \exp(\psi_i(\mathbf{y}(i), \mathbf{x}; \theta)) \prod_{d:i \in d} m_d(\mathbf{y}(i))^{\rho_d}.\end{aligned}\quad (13.9)$$

The learning process involved in the estimation of θ is typically based on gradient descent that minimizes the loss in (13.2) and should run until convergence, which is defined by the change rate of θ between successive gradient descent iterations. However, as noted by Domke [12], there are problems with this approach, where large thresholds in this change rate can lead to bad sub-optimal estimations, and tight thresholds result in slow convergence. These issues are circumvented by the truncated fitting algorithm [12], which uses a fixed number of iterations (i.e., no threshold is used in this training algorithm). We refer the reader to [12] for more details on this training algorithm.

13.3.2 Structured Support Vector Machine (SSVM)

The SSVM optimization to estimate θ consists of a regularized loss minimization problem formulated as $\theta^* = \min_{\theta} \|\theta\|^2 + \lambda \sum_i \ell(\mathbf{x}_i, \mathbf{y}_i, \theta)$, with $\ell(\cdot)$ defined in (13.4). The introduction of slack variable leads to the following optimization problem [13, 20]:

$$\begin{aligned}& \text{minimize}_{\theta} \frac{1}{2} \|\theta\|^2 + \frac{C}{|\mathcal{D}|} \sum_i \xi_i \\ & \text{subject to } E(\mathbf{y}_i, \mathbf{x}_i; \theta) - E(\hat{\mathbf{y}}_i, \mathbf{x}_i; \theta) \geq \Delta(\mathbf{y}_i, \hat{\mathbf{y}}_i) - \xi_i, \forall \hat{\mathbf{y}}_i \neq \mathbf{y}_i \\ & \quad \xi_i \geq 0.\end{aligned}\quad (13.10)$$

This optimization is a quadratic programming problem involving an intractably large number of constraints. In order to keep the number of constraints manageable, we use the cutting plane method that keeps a relatively small subset of the constraints by solving the maximization problem:

$$\hat{\mathbf{y}}_i = \arg \max_{\mathbf{y}} \Delta(\mathbf{y}_i, \mathbf{y}) + E(\mathbf{y}, \mathbf{x}_i; \theta) - E(\mathbf{y}_i, \mathbf{x}_i; \theta) - \xi_i, \quad (13.11)$$

which finds the most violated constraint for the i^{th} training sample given the parameter θ . Then if the right-hand side is strictly larger than zero, the most violated constraint is included in the constraint set and (13.10) is resolved. This iterative process runs until no more violated inequalities are found. Note that if we remove the constants from (13.11), the optimization problem is simply: $\hat{\mathbf{y}}_i = \arg \max_{\mathbf{y}} \Delta(\mathbf{y}_i, \mathbf{y}) + E(\mathbf{y}, \mathbf{x}_i; \theta)$, which can be efficiently solved using graph cuts [19] if the function $\Delta(\cdot, \cdot)$ can be properly decomposed in the label space. A simple example that works with graph cuts is $\Delta(\mathbf{y}, \mathbf{y}_i) = \sum_i 1 - \delta(\mathbf{y}(i) - \mathbf{y}_i(i))$,

which represents the Hamming distance and can be decomposed in the label space. Therefore, we use it in our methodology.

The label inference for a test mammogram \mathbf{x} , given the learned parameters θ from (13.10), is based on the following inference:

$$\mathbf{y}^* = \arg \max_{\mathbf{y}} E(\mathbf{y}, \mathbf{x}; \theta), \quad (13.12)$$

which can be efficiently and optimally solved for binary problems with graph cuts [19].

13.3.3 Potential Functions

It is worth noticing that the model in (13.3) can incorporate a large number of different types of potential functions. We propose the use of the deep convolutional neural networks (CNN) and deep belief networks (DBN), in addition to the more common Gaussian mixture model (GMM) and shape prior between the nodes of image and segmentation planes (see Fig. 13.2). Furthermore, we also propose the use of common pairwise potential functions.

13.3.3.1 CNN Potential Function

The CNN potential function is defined by [15] (Fig. 13.3):

$$\psi^{(1,1)}(\mathbf{y}(i), \mathbf{x}) = -\log P_{\text{CNN}}(\mathbf{y}(i)|\mathbf{x}, \theta_{\text{CNN}}), \quad (13.13)$$

where $P_{\text{CNN}}(\mathbf{y}(i)|\mathbf{x}, \theta_{\text{CNN}})$ denotes the probability of labelling the pixel $i \in |\Omega| \times |\Omega|$ with mass or background (given the whole input image \mathbf{x} for the ROI of the mass), and θ_{CNN} denotes the CNN parameters. A CNN model consists of multiple processing stages, with each stage comprising two layers (the convolutional layer,

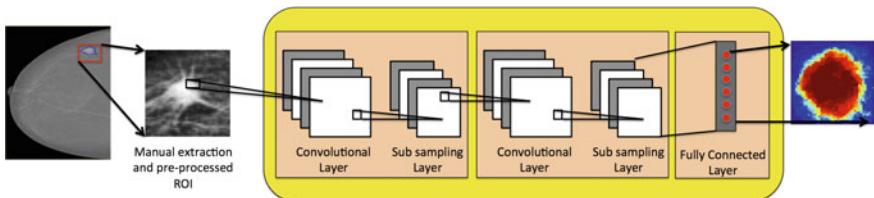


Fig. 13.3 CNN Model with the input \mathbf{x} (mass ROI from the mammogram) and the segmentation of the whole input with $\mathbf{y}(i) \in \{-1, +1\}$, denoting the absence (blue) or presence (red) of mass, respectively, and $i \in |\Omega| \times |\Omega|$

where the learned filters are applied to the image, and the nonlinear sub-sampling layer, that reduces the input image size for the next stage—see Fig. 13.3), and a final stage consisting of a fully connected layer. Essentially, the convolution stages compute the output at location j from input at i using the learned filter (at q^{th} stage) \mathbf{k}^q and bias b^q using $\mathbf{x}(j)^q = \sigma(\sum_{i \in M_j} \mathbf{x}(i)^{q-1} * \mathbf{k}_i^q + b_j^q)$, where $\sigma(.)$ is the logistic function and M_j is the input region addresses; while the nonlinear sub-sampling layers calculate sub-sampled data with $\mathbf{x}(j)^q = \downarrow(\mathbf{x}_j^{q-1})$, where $\downarrow(.)$ denotes a sub-sampling function that pools (using either the mean or max functions) the values from a region from the input data. The final stage consists of the convolution equation above using a separate filter for each output location, using the whole input from the previous layer. Inference is simply the application of this process in a feedforward manner, and training is carried out with stochastic gradient descent to minimize the segmentation error over the training set (via back propagation) [15].

13.3.3.2 DBN Potential Function

The DBN potential function is defined as [16]:

$$\psi^{(1,2)}(\mathbf{y}(i), \mathbf{x}) = -\log P_{\text{DBN}}(\mathbf{y}(i)|\mathbf{x}_S(i), \theta_{\text{DBN}}), \quad (13.14)$$

where $\mathbf{x}_S(i)$ is a patch extracted around image lattice position i of size $|\Omega| \times |\Omega|$ pixels, θ_{DBN} represents the DBN parameters (below, we drop the dependence on θ_{DBN} for notation simplicity), and

$$P_{\text{DBN}}(\mathbf{y}(i)|\mathbf{x}_S(i)) \propto \sum_{\mathbf{h}_1} \dots \sum_{\mathbf{h}_Q} P(\mathbf{x}_S(i), \mathbf{y}(i), \mathbf{h}_1, \dots, \mathbf{h}_Q), \quad (13.15)$$

with the DBN model consisting of a network with Q layers denoted by:

$$P(\mathbf{x}_S(i), \mathbf{y}(i), \mathbf{h}_1, \dots, \mathbf{h}_Q) = P(\mathbf{h}_Q, \mathbf{h}_{Q-1}, \mathbf{y}(i)) \left(\prod_{q=1}^{Q-2} P(\mathbf{h}_{q+1}|\mathbf{h}_q) \right) P(\mathbf{h}_1|\mathbf{x}_S(i)), \quad (13.16)$$

where $\mathbf{h}_q \in \mathbf{R}^{|q|}$ represents the hidden variables at layer q containing $|q|$ nodes. The first term in (13.16) is defined by:

$$-\log(P(\mathbf{h}_Q, \mathbf{h}_{Q-1}, \mathbf{y}(i))) \propto -\mathbf{b}_Q^\top \mathbf{h}_Q - \mathbf{a}_{Q-1}^\top \mathbf{h}_{Q-1} - \mathbf{a}_s^\top \mathbf{y}(i) - \mathbf{h}_Q^\top \mathbf{W} \mathbf{h}_{Q-1} - \mathbf{h}_Q^\top \mathbf{W}_s \mathbf{y}(i), \quad (13.17)$$

where \mathbf{a} , \mathbf{b} , \mathbf{W} are the network parameters, and the conditional probabilities are factorized as $P(\mathbf{h}_{q+1}|\mathbf{h}_q) = \prod_{i=1}^{|q+1|} P(\mathbf{h}_{q+1}(i)|\mathbf{h}_q)$ because the nodes in layer $q+1$ are independent from each other given \mathbf{h}_q , which is a consequence of the DBN structure ($P(\mathbf{h}_1|\mathbf{x}_S(i))$ is similarly defined). Furthermore, each node is activated by a sigmoid function $\sigma(.)$, which means that $P(\mathbf{h}_{q+1}(i)|\mathbf{h}_q) = \sigma(\mathbf{b}_{q+1}(i) + \mathbf{W}_i \mathbf{h}_q)$. The

inference is based on the mean field approximation of the values in layers \mathbf{h}_1 to \mathbf{h}_{Q-1} followed by the computation of free energy on the top layer [16]. The learning of the DBN parameters θ_{DBN} in (13.18) is achieved with an iterative layer by layer training of auto-encoders using contrastive divergence [16].

13.3.3.3 GMM Potential Function

The GMM potential function is defined by:

$$\psi^{(1,3)}(\mathbf{y}(i), \mathbf{x}) = -\log P_{\text{GMM}}(\mathbf{y}(i)|\mathbf{x}(i), \theta_{\text{GMM}}), \quad (13.18)$$

where $P_{\text{GMM}}(\mathbf{y}(i)|\mathbf{x}(i), \theta_{\text{GMM}}) = (1/Z) \sum_{m=1}^G \pi_m \mathcal{N}(\mathbf{x}(i); \mathbf{y}(i), \mu_m, \sigma_m) P(\mathbf{y}(i))$ with $\theta_{\text{GMM}} = [\pi_m, \mu_m, \sigma_m]_{m=1}^G$, $\mathcal{N}(\cdot)$ is the Gaussian function, Z is the normalizer, $\mathbf{x}(i)$ represents the pixel value at image lattice position i , and $P(\mathbf{y}(i) = 1) = 0.5$. The parameter vector θ_{GMM} in (13.14) is learned with the expectation–maximization (EM) algorithm [34] using the annotated training set.

13.3.3.4 Shape Prior Potential Function

The shape prior potential function is computed from the average annotation (estimated from the training set) at each image lattice position $i \in \mathcal{Q}$, as follows:

$$\psi^{(1,4)}(\mathbf{y}(i), \mathbf{x}) = -\log P_{\text{prior}}(\mathbf{y}(i)|\theta_{\text{prior}}), \quad (13.19)$$

where $P(\mathbf{y}(i)|\theta_{\text{prior}}) = \lambda(1/N) \sum_n \delta(\mathbf{y}_n(i) - 1) + (1 - \lambda)$, where $\lambda \in [0, 1]$.

13.3.3.5 Pairwise Potential Functions

The pairwise potential functions between label nodes in (13.3) encode label and contrast dependent labelling homogeneity. In particular, the label homogeneity is defined by:

$$\psi^{(2,1)}(\mathbf{y}(i), \mathbf{y}(j), \mathbf{x}) = 1 - \delta(\mathbf{y}(i) - \mathbf{y}(j)), \quad (13.20)$$

and the contrast dependent labelling homogeneity that we use is as follows [20]:

$$\psi^{(2,2)}(\mathbf{y}(i), \mathbf{y}(j), \mathbf{x}) = (1 - \delta(\mathbf{y}(i) - \mathbf{y}(j))) C(\mathbf{x}(i) - \mathbf{x}(j)), \quad (13.21)$$

where $C(\mathbf{x}(i), \mathbf{x}(j)) = e^{-(\mathbf{x}(i) - \mathbf{x}(j))^2}$.

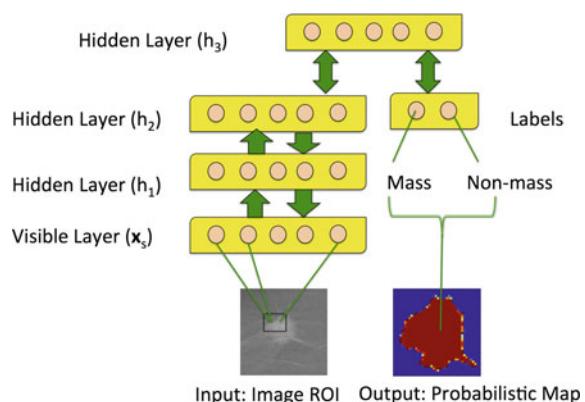
13.4 Experiments

In this section, we first introduce the datasets used, followed by an explanation of the experimental set-up and the results achieved.

13.4.1 Materials and Methods

We assess performance of our methodology on two publicly available datasets: INbreast [6] and DDSM-BCRP [7]. The INbreast [6] dataset consists of set of 56 cases containing 116 accurately annotated masses. We divide this dataset into mutually exclusive training and testing sets, each containing 28 cases (58 annotated images each). The DDSM-BCRP [7] dataset consists of 39 cases (77 annotated images) for training and 40 cases (81 annotated images) for testing. Segmentation accuracy is assessed with Dice index (DI) = $\frac{2TP}{FP+FN+2TP}$, where TP denotes the number of mass pixels correctly segmented, FP the background pixels falsely segmented as mass, and FN the mass pixels not identified. The ROI to be segmented is obtained by extracting a rectangular bounding box from around the centre of the manual annotation, where the size for each dimension of the rectangle is produced by the size of the annotation plus two pixels [35]. We use the preprocessing method by Ball and Bruce [27] in order to increase the contrast of the input image. This ROI is then resized to 40×40 pixels using bicubic interpolation. The model selection process for the structure of the CNN and DBN is performed via cross-validation on the training set, and for the CNN, the net structure is the one in Fig. 13.3, where the first stage has 6 filters of size 5×5 and the second stage has 12 filters of size 5×5 , and the sub-sampling method after each of these stages uses max pooling that reduces the input to half of its initial size in both stages. The final stage of the CNN has a fully connected layer with 588 nodes and an output layer with 1600 nodes which is reshaped to 40×40

Fig. 13.4 DBN model with variables x (mass ROI from the mammogram) and classification $y \in \{-1, +1\}$, denoting the absence or presence of mass, respectively



nodes (i.e., same size of the input layer). For the DBN, the model is the one shown in Fig. 13.4 with \mathbf{h}_1 , \mathbf{h}_2 and \mathbf{h}_3 each containing 50 nodes, with input patches of sizes 3×3 and 5×5 . We assessed the efficiency of our segmentation methodology with the mean execution time per image on a computer with the following configuration: Intel(R) Core(TM) i5-2500k 3.30GHz CPU with 8GB RAM.

13.4.2 Results

The experimental results presented in Fig. 13.5 assess the importance of adding each potential function to the energy model defined in (13.3). This figure shows the mean Dice index results on the testing set of INbreast using the CRF and SSVM models. In particular, we show these results using several subsets of the potential functions “CNN”, “DBN 3×3 ”, “DBN 5×5 ”, “GMM”, “Pairwise” and “Prior” presented in Sect. 13.3.3 (i.e., the potentials $\phi^{(1,k)}$ for $k = \{1, 2, 3, 4\}$ with 3×3 and 5×5 denoting the image patch size used by the DBN). It is important to mention that the Dice index of our methodology using all potential functions on the training set of INbreast is 0.93 using CRF and 0.95 using SSVM. It is also worth mentioning that the results on the INbreast test set, when we do not use preprocessing [27], fall to 0.85 using all potential functions for both models.

The comparison between the results from our methodology and other state-of-the-art results is shown in Table 13.1. This comparison is performed on the testing sets of DDSM-BCRP and INbreast, with the Dice index, average training time (for the whole training set) and testing time (per image), where our CRF and SSVM models have all potential functions: CNN+DBN 3×3 + DBN 5×5 + GMM + Prior + Pairwise. Notice that in this table, we only list the results available for the methods that use these publicly available databases because the great majority of

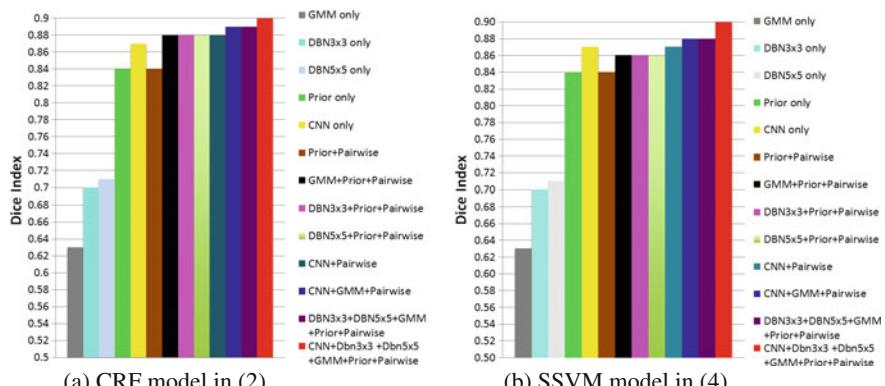
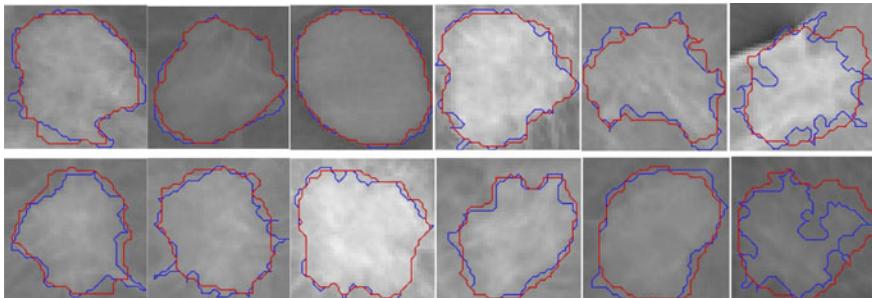


Fig. 13.5 Dice index on the test set of INbreast dataset for our CRF (a) and SSVM (b) models, using various subsets of the unary and pairwise potential functions

Table 13.1 Comparison between the proposed CRF and SSVM models and several state-of-the-art methods

Method	#Images	Dataset	Dice index	Test run. time	Train run. time
Proposed CRF model	116	INbreast	0.90	0.1 s	360 s
Proposed SSVM model	116	INbreast	0.90	0.8 s	1800 s
Cardoso et al. [35]	116	INbreast	0.88	?	?
Dhungel et al. [9]	116	INbreast	0.88	0.8 s	?
Dhungel et al. [10]	116	INbreast	0.89	0.1 s	?
Proposed CRF model	158	DDSM-BCRP	0.90	0.1 s	383 s
Proposed SSVM model	158	DDSM-BCRP	0.90	0.8 s	2140 s
Dhungel et al. [9]	158	DDSM-BCRP	0.87	0.8 s	?
Dhungel et al. [10]	158	DDSM-BCRP	0.89	0.1 s	?
Beller et al. [4]	158	DDSM-BCRP	0.70	?	?

**Fig. 13.6** Mass segmentation results produced by the CRF model on INbreast test images, where the *blue curve* denotes the manual annotation and *red curve* represents the automatic segmentation

papers published in this area have used subsets of the DDSM dataset and manual annotations that are not publicly available, which makes a direct comparison with these methods impossible. Finally, Fig. 13.6 shows examples of segmentation results produced by our CRF model on the test set of INbreast.

13.5 Discussion and Conclusions

The results from Fig. 13.5 explain the importance of each potential function used in the CRF and SSVM models, where it is clear that the CNN potential function provides the largest boost in performance. The addition of GMM and shape prior to deep learning models provides considerable improvements for both CRF and SSVM models. Another interesting observation is the fact that image preprocessing [27] appears to be important since it shows a substantial gain in terms of segmentation accuracy. The comparison with other methods in Table 13.1 shows that our methodology currently produces the best results for both databases, and the CRF and SSVM models hold comparable results in terms of segmentation accuracy. However, the comparison in terms of training and testing running times shows a significant advantage to the CRF model.

There are other important conclusions to make about the training and testing processes that are not displayed in these results: (1) we tried other types of CNN structures, such as with different filter sizes, and we also tried to use more than one CNN model as additional potential functions, but the use of only one CNN with the structure detailed in Sect. 13.4.1 produced the best result in cross-validation (the main issue affecting the CNN models is overfitting); (2) for the DBN models, we have also tried different input sizes (e.g., 7×7 patches), but the combinations of the ones detailed in Sect. 13.4.1 provided the best cross-validation results; and (3) the training for both the CRF and SSVM models estimates a much larger weight to the CNN potential function compared to other potential functions in Sect. 13.3.3, indicating that this is the most important potential function, but the CNN model alone (without CRF or SSVM) overfits the training data (with a Dice of 0.87 on test and 0.95 on training), so the structural prediction models serve as a regularizer to the CNN model. Finally, from the visual results in Fig. 13.6, we can see that our proposed CRF model produces quite accurate segmentation results when the mass does not show very sharp corners and cusps. We believe that the main issue affecting our method in these challenging cases is the limited size of the training sets in the DDSM-BCRP and INbreast datasets, which do not contain enough examples of such segmentations in order to allow an effective learning of a model that can deal with such complicated segmentation problems.

Acknowledgements This work was partially supported by the Australian Research Council's Discovery Projects funding scheme (project DP140102794). Prof. Bradley is the recipient of an Australian Research Council Future Fellowship(FT110100623).

References

1. Jemal A, Siegel R, Ward E, Hao Y, Xu J, Murray T, Thun MJ (2008) Cancer statistics, 2008. CA Cancer J Clin 58(2):71–96
2. Yuan Y, Giger ML, Li H, Suzuki K, Sennett C (2007) A dual-stage method for lesion segmentation on digital mammograms. Med Phys 34:4180
3. Rahmati P, Adler A, Hamarneh G (2012) Mammography segmentation with maximum likelihood active contours. Med Image Anal 16(6):1167–1186
4. Beller M, Stotzka R, Müller TO, Gemmeke H (2005) An example-based system to support the segmentation of stellate lesions. In: Bildverarbeitung für die Medizin 2005. Springer, Berlin, pp 475–479
5. Elmore JG, Jackson SL, Abraham L, Miglioretti DL, Carney PA, Geller BM, Yankaskas BC, Kerlikowske K, Onega T, Rosenberg RD et al (2009) Variability in interpretive performance at screening mammography and radiologists characteristics associated with accuracy1. Radiology 253(3):641–651
6. Moreira IC, Amaral I, Domingues I, Cardoso A, Cardoso MJ, Cardoso JS (2012) Inbreast: toward a full-field digital mammographic database. Acad Radiol 19(2):236–248
7. Heath M, Bowyer K, Kopans D, Moore R, Kegelmeyer P (2000) The digital database for screening mammography. In: Proceedings of the 5th international workshop on digital mammography. pp 212–218
8. Fenton JJ, Taplin SH, Carney PA, Abraham L, Sickles EA, D'Orsi C, Berns EA, Cutter G, Hendrick RE, Barlow WE et al (2007) Influence of computer-aided detection on performance of screening mammography. N Engl J Med 356(14):1399–1409
9. Dhungel N, Carneiro G, Bradley AP (2015) Deep structured learning for mass segmentation from mammograms. In: 2015 IEEE international conference on image processing (ICIP). pp 2950–2954
10. Dhungel N, Carneiro G, Bradley AP (2015) Tree re-weighted belief propagation using deep learning potentials for mass segmentation from mammograms. In: 2015 IEEE 12th international symposium on biomedical imaging (ISBI). pp 760–763
11. Carneiro G, Georgescu B, Good S, Comaniciu D (2008) Detection and measurement of fetal anatomies from ultrasound images using a constrained probabilistic boosting tree. IEEE Trans Med Imaging 27(9):1342–1355
12. Domke J (2013) Learning graphical model parameters with approximate marginal inference. arXiv preprint [arXiv:1301.3193](https://arxiv.org/abs/1301.3193)
13. Tschantaridis I, Joachims T, Hofmann T, Altun Y (2005) Large margin methods for structured and interdependent output variables. J Mach Learn Res 1453–1484
14. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: NIPS, vol 1. p 4
15. LeCun Y, Bengio Y (1995) Convolutional networks for images, speech, and time series. In: The handbook of brain theory and neural networks, vol 3361
16. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
17. Wang C, Komodakis N, Paragios N (2013) Markov random field modeling, inference and learning in computer vision and image understanding: A survey. Comput Vis Image Underst 117(11):1610–1627
18. Wainwright MJ, Jaakkola TS, Willsky AS (2003) Tree-reweighted belief propagation algorithms and approximate ml estimation by pseudo-moment matching. In: Workshop on artificial intelligence and statistics. Society for artificial intelligence and statistics Np, vol 21. p. 97
19. Boykov Y, Veksler O, Zabih R (2001) Fast approximate energy minimization via graph cuts. IEEE Trans Pattern Anal Mach Intell 23(11):1222–1239
20. Szummer M, Kohli P, Hoiem D (2008) Learning CRFS using graph cuts. In: Computer vision–ECCV 2008. Springer, Berlin, pp 582–595
21. Catarious DM Jr, Baydush AH, Floyd CE Jr (2004) Incorporation of an iterative, linear segmentation routine into a mammographic mass cad system. Med Phys 31(6):1512–1520

22. Song E, Jiang L, Jin R, Zhang L, Yuan Y, Li Q (2009) Breast mass segmentation in mammography using plane fitting and dynamic programming. *Acad Radiol* 16(7):826–835
23. Timp S, Karssemeijer N (2004) A new 2d segmentation method based on dynamic programming applied to computer aided detection in mammography. *Med Phys* 31(5):958–971
24. Domínguez AR, Nandi AK (2009) Toward breast cancer diagnosis based on automated segmentation of masses in mammograms. *Pattern Recognit* 42(6):1138–1148
25. Yu M, Huang Q, Jin R, Song E, Liu H, Hung C-C (2012) A novel segmentation method for convex lesions based on dynamic programming with local intra-class variance. In: Proceedings of the 27th annual ACM symposium on applied computing. ACM, New York, pp 39–44
26. Xu S, Liu H, Song E (2011) Marker-controlled watershed for lesion segmentation in mammograms. *J Digital Imaging* 24(5):754–763
27. Ball J, Bruce L (2007) Digital mammographic computer aided diagnosis (CAD) using adaptive level set segmentation. In: 29th annual international conference of the IEEE engineering in medicine and biology society, 2007. EMBS 2007. IEEE, New York, pp 4973–4978
28. te Brake GM, Karssemeijer N, Hendriks JH (2000) An automatic method to discriminate malignant masses from normal tissue in digital mammograms. *Phys Med Biol* 45(10):2843
29. Sahiner B, Chan H-P, Petrick N, Helvie MA, Hadjiiski LM (2001) Improvement of mammographic mass characterization using spiculation measures and morphological features. *Med Phys* 28(7):1455–1465
30. Sethian JA (1999) Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science, vol 3. Cambridge University Press, Cambridge
31. Shi J, Sahiner B, Chan H-P, Ge J, Hadjiiski L, Helvie MA, Nees A, Wu Y-T, Wei J, Zhou C et al (2007) Characterization of mammographic masses based on level set segmentation with new image features and patient information. *Med Phys* 35(1):280–290
32. Nowozin S, Lampert CH (2011) Structured learning and prediction in computer vision. *Found Trends® Comput Graph Vis* 6(3–4):185–365
33. Meltzer T, Globerson A, Weiss Y (2009) Convergent message passing algorithms: a unifying view. In: Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence, AUAI Press, pp 393–401
34. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. *J R Stat Soc Ser B (Methodol)*, 1–38
35. Cardoso JS, Domingues I, Oliveira HP (2014) Closed shortest path in the original coordinates with an application to breast cancer. *Int J Pattern Recognit Artif Intell*

Chapter 14

Deep Learning Based Automatic Segmentation of Pathological Kidney in CT: Local Versus Global Image Context

**Yefeng Zheng, David Liu, Bogdan Georgescu, Daguang Xu
and Dorin Comaniciu**

Abstract Chronic kidney disease affects one of every ten adults in USA (over 20 million). Computed tomography (CT) is a widely used imaging modality for kidney disease diagnosis and quantification. However, automatic pathological kidney segmentation is still a challenging task due to large variations in contrast phase, scanning range, pathology, and position in the abdomen, etc. Methods based on global image context (e.g., atlas- or regression-based approaches) do not work well. In this work, we propose to combine deep learning and marginal space learning (MSL), both using local context, for robust kidney detection and segmentation. Here, deep learning is exploited to roughly estimate the kidney center. Instead of performing a whole axial slice classification (i.e., whether it contains a kidney), we detect local image patches containing a kidney. The detected patches are aggregated to generate an estimate of the kidney center. Afterwards, we apply MSL to further refine the pose estimate by constraining the position search to a neighborhood around the initial center. The kidney is then segmented using a discriminative active shape model. The proposed method has been trained on 370 CT scans and tested on 78 unseen cases. It achieves a mean segmentation error of 2.6 and 1.7 mm for the left and right kidney, respectively. Furthermore, it eliminates all gross failures (i.e., segmentation is totally off) in a direct application of MSL.

14.1 Introduction

There are two bean-shaped kidneys in a normal person. Their main function is to extract waste from blood and release it from the body as urine. Chronic kidney disease (CKD) is the condition that a kidney does not function properly longer than a certain period of time (usually three months). In the most severe stage, the kidney completely stops working and the patient needs dialysis or a kidney transplant to survive. The incidence of CKD increases dramatically with age, especially for people

Y. Zheng (✉) · D. Liu · B. Georgescu · D. Xu · D. Comaniciu
Medical Imaging Technologies, Siemens Healthcare, Princeton, NJ, USA
e-mail: yefeng.zheng@siemens.com

older than 65 years. According to an estimate from the Center of Disease Control and Prevention, one in every ten American adults (over 20 million) has some level of CKD [1]. Computed tomography is a widely used imaging modality for kidney disease diagnosis and quantification. Different contrast phases are often used to diagnose different kidney diseases, including a native scan (no contrast at all to detect kidney stone), corticomedullary phase (much of contrast material still resides within the vascular system), nephrographic phase (contrast enters the collecting ducts), and excretory phase (contrast is excreted into the calices) [2].

Various methods have been proposed to detect and segment an anatomical structure and many can be applied to kidney segmentation. Atlas-based methods segment a kidney by transferring the label from an atlas to input data after volume registration [3]. However, volume registration is time consuming and several volume registrations are required in a multi-atlas approach to improve segmentation accuracy, but with increased computation time (taking several minutes to a few hours). Recently, regression-based approaches [4–6] were proposed to efficiently estimate a rough position of an anatomical structure. An image patch cropped from anywhere inside a human body can be used to predict the center of a target organ by assuming a relatively stable positioning of organs. Regression is much more efficient than atlas registration and can estimate the rough position of an organ in a fraction of a second. Both atlas-based and regression-based approaches use global context to localize an organ. Alternatively, an organ can be detected via local classification in which we train a classifier that tells us if an image patch contains the target organ or not. Marginal space learning (MSL) [7, 8] is such an example, which efficiently prunes the pose parameter space to estimate the nine pose parameters (translation, orientation, and size) in less than a second. Recently, deep learning has been applied to kidney segmentation using the fully convolutional network (FCN) architecture [9]. However, it has only been validated on contrasted kidneys; therefore, its performance on more challenging dataset like ours is not clear.

Though existing methods may work well on normal kidneys, pathological kidney segmentation is still a challenging task. First, the relative position of a pathological kidney to surrounding organs varies. Normally, the right kidney lies slightly below the liver, while the left kidney lies below the spleen (as shown by the third and fourth patients in Fig. 14.1). However, in a patient with severe pathologies, a kidney may be pushed off by neighboring organs (e.g., liver, spleen, stomach, and colon) due to the excessive growth of tumor or previous surgery. For example, in our dataset, the relative position of the right kidney to the liver varies quite a lot, as shown in Fig. 14.1. The first patient has the right kidney lying at the bottom of the abdomen; while the kidney of the last patient resides at the top of the abdomen. Previous methods relying on global image context [3–6] cannot handle such large variation. Second, kidneys with severe pathologies exhibit extra variations in size, shape, and appearance. Last but not least, we want to develop a generic algorithm, which can handle all possible contrast phases in a renal computed tomography (CT) scan (as shown in Fig. 14.5). The native scan is especially difficult to segment due to the weak contrast between a kidney and the surrounding tissues.

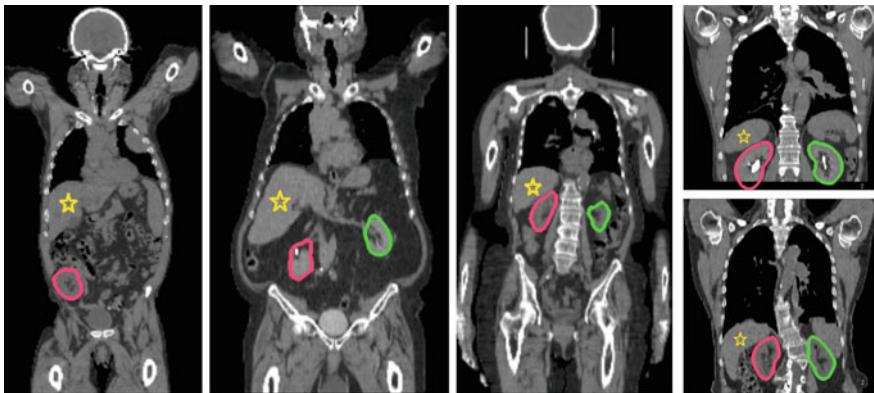


Fig. 14.1 Segmentation results of the *left* (green) and *right* (red) kidney. The relative position of the *right* kidney to the liver (yellow star) varies a lot as well as the scanning range. Note, the first patient has the *left* kidney surgically removed

Due to the large “floating” range of a pathological kidney inside the abdomen, a local classification-based approach is more robust than an approach exploiting global context (e.g., global image registration [3] or using the liver or other organs to predict the kidney position [4–6]). For example, even though working well on detecting other large organs (e.g., liver and lungs), a regression-based approach performs worse than MSL on kidney detection in a 3D magnetic resonance imaging (MRI) dataset [6]. Due to the challenges posed by our dataset, a direct application of MSL to CT kidney segmentation achieves a mixed result. It successfully detects and segments a kidney in 90–95% of cases. For the remaining cases, the segmentation may be completely off due to the failure in position detection, especially for patients with severe pathologies. We suspect that the limited success of MSL on pathological kidney detection is due to its use of hand-crafted features, which lack discriminative power to handle such large variations in our dataset.

In this work, we propose to exploit deep learning for rough localization of the kidney. Deep learning can automatically build a hierarchical image feature representation, which has been shown in a lot of computer vision problems to outperform hand-crafted features. Recently, deep learning has been applied in many medical image analysis problems, including body region recognition [10], landmark detection [11], cell detection [12], lymph node detection [13], organ detection/segmentation [14, 15], cross-modality registration [16], and 2D/3D registration [17]. On all these applications, deep learning outperforms the state of the art. In this work we apply deep learning to determine the abdomen range in a whole-body scan and then roughly localize the kidney inside the abdomen. Deep learning is especially data hungry, compared to other machine learning algorithms, to achieve good generalization capability. To mitigate the overfitting issue, we synthesize a lot of training data with realistic nonrigid deformations. After kidney localization, we apply MSL, but constrain the position search to a small range around the already detected kidney center. MSL

also estimates the orientation and size of the kidney, thereby providing a quite good initial segmentation after aligning a pre-learned mean shape to the estimated pose. The segmentation is further refined using a discriminative active shape model (ASM) [7]. Please note, in this work, we treat the left and right kidney as different organs and train separate models to detect/segment them.

The remainder of the chapter is organized as follows. In Sect. 14.2, we present an approach to synthesize more training data with nonrigid deformation of the kidney. Abdomen range detection is presented in Sect. 14.3, which is used to constrain the search of the kidney. Kidney localization is described with detail in Sect. 14.4, followed by segmentation in Sect. 14.5. Quantitative experiments in Sect. 14.6 demonstrate the robustness of the proposed method in segmenting pathological kidneys. This chapter concludes with Sect. 14.7.

14.2 Training Data Synthesis

To achieve good generalization on unseen data, deep learning needs a lot of training data; therefore, data augmentation is widely used to generate more training samples. Conventional data augmentation adds random translation, rotation, scaling, and intensity transformation, etc. In addition, we also add nonrigid deformation to cover variation in the kidney shape using an approach proposed in [18, 19]. Given two training volumes I_s and I_t with annotated kidney mesh S_s and S_t , respectively, we estimate the deformation field that warps S_s to S_t . The estimated deformation field is then used to warp all voxels in I_s to create a synthesized volume I_s^t . Here, we use the thin-plate spline (TPS) model [20] to represent the nonrigid deformation between the source and target volumes. The TPS interpolant $f(x, y, z)$ minimizes the bending energy of a thin plate

$$I_f = \int \int \int_{\mathcal{R}^3} \left(\frac{\partial^2 f}{\partial x^2} \right)^2 + \left(\frac{\partial^2 f}{\partial y^2} \right)^2 + \left(\frac{\partial^2 f}{\partial z^2} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial x \partial z} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial y \partial z} \right)^2 dx dy dz. \quad (14.1)$$

The interpolant $f(x, y, z)$ can be estimated analytically [20].

The kidneys in source and target volumes may be captured in different coordinate systems with different field of views. To avoid unnecessary coordinate changes, before estimating the TPS deformation field, we translate S_t so that, after translation, it has the same mass center as S_s .

The above TPS anchor points are concentrated on the kidney surface. To make the deformation field of background tissues smooth, we add the eight corners of the field of view of source volume I_s as additional TPS anchor points. Suppose the size of I_s is W , H , and D along three different dimensions, respectively. The following eight points are added as additional anchor points: $(0, 0, 0)$, $(W, 0, 0)$, $(0, H, 0)$, $(W, H, 0)$, $(0, 0, D)$, $(W, 0, D)$, $(0, H, D)$, and (W, H, D) . These corner points will not change after deformation; therefore, the deformation field is strong around the kidney and gradually fades out towards the volume border.

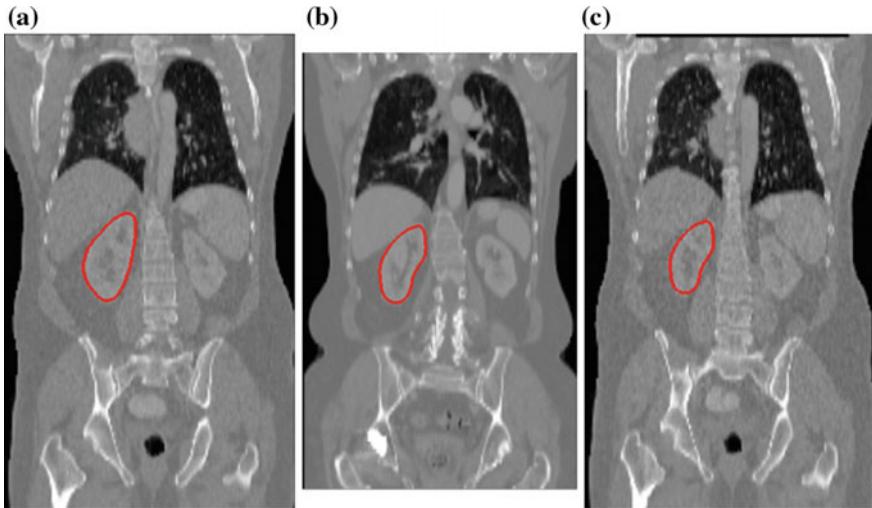


Fig. 14.2 Synthesis of training images with nonrigid deformation of the *right* kidney. **a** Source volume with *right* kidney mesh overlaid. **b** Target volume. **c** Synthesized volume with intensity pattern from the source volume but the *right* kidney shape from the target volume

Conceptually, the TPS deformation maps the source volume to the target volume. However, if we directly estimate this forward TPS warping and apply it to all voxels of the source volume, the resulting volume may have holes (i.e., voxels without any source voxels mapping to) unless we densely upsample the source volume. Dense upsampling the source volume increases the computation time when we perform forward TPS warping. In our implementation, we estimate the backward TPS warping, which warps the target volume to the source volume. For each voxel in the target volume, we use the backward TPS warping to find the corresponding position in the source volume. Normally, the corresponding source position is not on the imaging grid; so, linear interpolation is used to calculate the corresponding intensity in the source volume.

Figure 14.2 shows an example of data synthesis for the right kidney. The source and target volumes are shown in Fig. 14.2a, b, respectively. The synthesized volume has an intensity pattern from the source volume but the right kidney shape from the target volume, as shown in Fig. 14.2c. The synthesized data is so visually realistic that it is difficult (if not impossible) to tell which one is real and which one is synthesized, by comparing Fig. 14.2a, c.

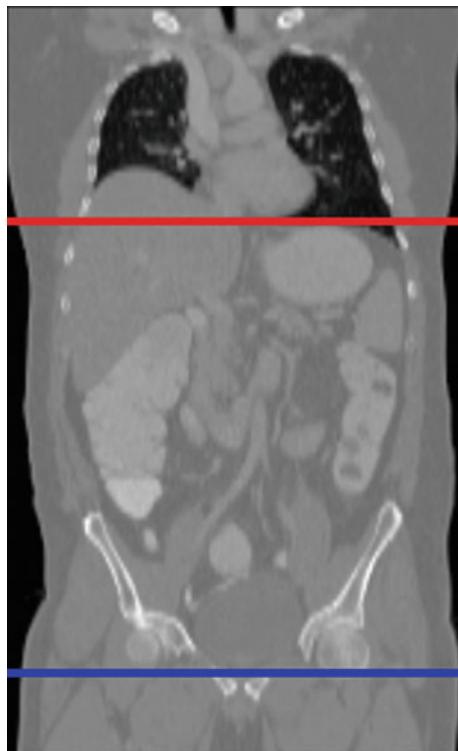
We have 370 training volumes. Since we can synthesize a new volume by taking an ordered pair of training data, we may synthesize up to $370 \times 369 = 136,530$ new training volumes. To reduce the training burden, we synthesize 2,000 new volumes by randomly picking training data pairs.

14.3 Abdomen Localization Using Deep Learning

Besides dedicated kidney scans, our dataset contains many whole-body CT scans, which are often acquired to diagnose cancer at multiple organs (e.g., to rule out metastasis). Depending on the exam indication, the scan range along the z-axis (pointing from patient's toe to head) may be quite large and often varies a lot (as shown in Fig. 14.1). If we can constrain position detection to a limited range along the z-axis, most detection failures can be eliminated. A kidney is bounded by the abdomen, though the position of a pathological kidney inside the abdomen varies as shown in Fig. 14.1. In this work we use a two-step approach to localize a kidney. We first determine the abdomen range and then detect a kidney inside the abdomen.

The abdomen has quite different image characteristics to other body regions (e.g., head, thorax, and legs); therefore, it can be detected reliably with an efficient classification scheme. We perform slice-wise classification by assigning a slice to one of three classes: above abdomen (head or thorax), abdomen, and legs. In our application, the lower limit of the abdomen stops at the top of the pubic symphysis (indicated by the blue line in Fig. 14.3), which joins the left and right pubic bones. With bony structures clearly visible in a CT volume, this landmark is easy to identify by a human

Fig. 14.3 Definition of abdomen range in a whole-body CT volume. The *upper* limit of the abdomen stops at the *bottom* of the heart; while, the *lower* limit of the abdomen stops at the *top* of the pubic symphysis



being and, hopefully, also easy to detect automatically. The thorax and abdomen are separated by the diaphragm, which is a cursive structure as shown in Fig. 14.3. In this work we are not interested in the exact boundary. Instead, we use one axial slice to determine the upper limit of the abdomen. Here, we pick a slice at the bottom of the heart as the upper limit of the abdomen (the red line in Fig. 14.3).

A convolutional neural network (ConvNet) is trained to perform the slice-wise classification. To be specific, we use Caffe [21] to train a ConvNet with five layers of convolution and two fully connected layers (the “*bvlc_reference_caffenet*” model). A straightforward approach is to take a whole axial image as input to a classifier. However, the classifier may have difficulty in handling the variation of patient’s position inside a slice. Here, we first extract the body region (the white boxes in Fig. 14.4) by excluding the black margin. The input image is then resized to 227×227 pixels before feeding into the ConvNet.

Once the ConvNet is trained, we apply it to all slices in an input volume. For each slice we get a three-dimensional vector representing the classification confidence of each class (head-thorax, abdomen, and legs). To find the optimal range of the abdomen, we aggregate the classification scores as follows. Suppose we want to determine the bottom range of the abdomen (the boundary between the abdomen and legs); the input volume has n slices; and, the classification scores for the abdomen and leg classes are $A[1, \dots, n]$ and $L[1, \dots, n]$, respectively. We search for the optimal slice index Ab_L such that

$$Ab_L = \operatorname{argmax}_j \sum_{i=1}^j (L[i] - A[i]) + \sum_{i=j+1}^n (A[i] - L[i]). \quad (14.2)$$

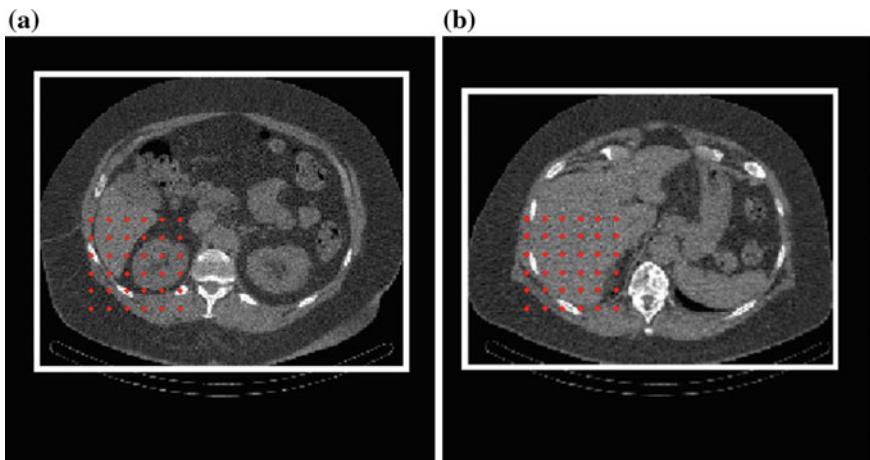


Fig. 14.4 Patch centers (red dots) on a positive axial slice (a) and a negative slice (b) for right kidney localization. White boxes show the body region after excluding black margin

Below the abdomen/leg boundary, the leg class should have a higher score than the abdomen class. So, each individual item of the first term should be positive. Above the abdomen/leg boundary, the abdomen class should have a higher score; therefore, the second term should be positive. Equation (14.2) searches for an optimal slice index maximizing the separation between abdomen and legs. The upper range of the abdomen Ab_U is determined in a similar way.

$$Ab_U = \operatorname{argmax}_j \sum_{i=1}^j (A[i] - T[i]) + \sum_{i=j+1}^n (T[i] - A[i]). \quad (14.3)$$

Here, $T[1, \dots, n]$ is the classification score of the thorax-head class. Aggregating the classification score of all slices, our approach is robust against noise in classification result on some slices.

Our approach is robust: the upper/lower range of the abdomen can be determined within an error of 1–2 slices without any gross failure. Using abdomen detection, we can quickly exclude up to 75% of slices from the following more through kidney localization procedure (which is far more time consuming). It accelerates the detection speed and, at the same time, reduces the kidney detection failures.

Previously, slice-based classification is also used by Seifert et al. [22] to determine the body region. Our approach has several advantages compared to [22]. First, Seifert et al. formulated the task as a two-class classification problem, where the slice separating different body regions is taken as a positive sample and all other slices are treated as negative samples. Each training volume contributes only one positive training sample (maybe, a few after adding perturbation) and there are many more negative samples, which are often downsampled to get a balanced training set. So, only a small number of slices are used for training. In our approach, we formulate the task as a multi-class classification problem (i.e., thorax-head, abdomen, and legs). The distribution of different classes is more balanced; therefore, much more slices can be used for training. Second, using a two-class classification scheme, ideally, only the target slice should generate a high score and all other slices give a low score. If there is classification error on the target slice, the detection fails. In our approach, we aggregate the classification score of all slices to determine the boundary between body regions. Therefore, our approach potentially is more robust than [22]. Third, to separate the body into multiple regions, Seifert et al. train a binary classifier for each separating slice (in our case, two slices with one for the upper and the other for the lower range of the abdomen). During detection, all these classifiers need to be applied to all slices. Using a multi-class classification scheme, we apply a single classifier to each slice only once, which is more computationally efficient. Last but not least, [22] uses hand-crafted Haar wavelet-like features for classification; while, we leverage the recent progress on deep learning, which can automatically learn more powerful hierarchical image features.

14.4 Kidney Localization Using Deep Learning

Similar to abdomen detection, a classifier (e.g., ConvNet) can be trained to tell us if an axial image contains a kidney or not. However, this naive global context approach generates a few gross failures. Assuming a kidney is next to the liver/spleen, a deep learning algorithm may use features from the liver/spleen to predict presence of a kidney in an axial image, which has a large input field of view covering kidney and surrounding organs. However, as shown in Fig. 14.1, the relative position of a pathological kidney to its surrounding organs is not stable. In this work, we propose to crop a small image patch enclosing the kidney as input to a ConvNet. Since we do not know the exact position of the kidney, we need to test multiple patches. The red dots in Fig. 14.4 show the centers of cropped patches inside a predicted region of interest (ROI). During the training phase, we calculate the shift of the kidney center to the body region box center. The distribution of the shift helps us to define the ROI. As shown in Fig. 14.4, we crop $6 \times 6 = 36$ patches. Around each patch center, we crop an image of $85 \times 85 \text{ mm}^2$, which is just enough to cover the largest kidney in our training set. For each positive slice, the patch with the smallest distance to the true kidney center is picked as a positive training sample. Afterwards, we randomly pick the same number of negative patches from slices without a kidney. Figure 14.5 shows a few positive and negative training patches. Some negative patches are quite similar to positive patches (e.g., the last negative patch versus the first two positive patches).

Similar to abdomen localization, we use Caffe [21] to train a ConvNet using the “bvlc_reference_caffenet” model. The standard input image size to this ConvNet model is 227×227 pixels. For patch-based classification, we need to perform multiple classifications. To speed up the computation, we tried different input sizes and found that we could reduce the input to 65×65 pixels without deteriorating the accuracy. With a smaller input image size, we reduce the filter size of the first convolution layer from 11×11 to 7×7 and the stride from 4 to 2. All the other network parameters are kept the same.

After classification of all axial slices, the ultimate goal is to have a single estimate of the kidney center. For whole slice or body region based classification, we only get one classification score for each slice. For a patch-based approach, we have multiple patches and each has a classification score (a real positive value). We take the summation of scores of all patches that are classified positive as the final score of that slice. Negative patches do not contribute. A slice with more positive patches tends to have a higher score. Independent classification of each slice often generates noisy output (as shown in Fig. 14.6). We perform Gaussian smoothing with a kernel of 100 mm (the rough size of a kidney along the z-axis). After smoothing, we pick the slice with the largest score as the kidney center along the z-axis (Z_o). We then take positive patches on all slices within $[Z_o - 50, Z_o + 50]$ mm. The weighted average of the positive patch centers provides an estimate of the kidney center in the x and y axes.

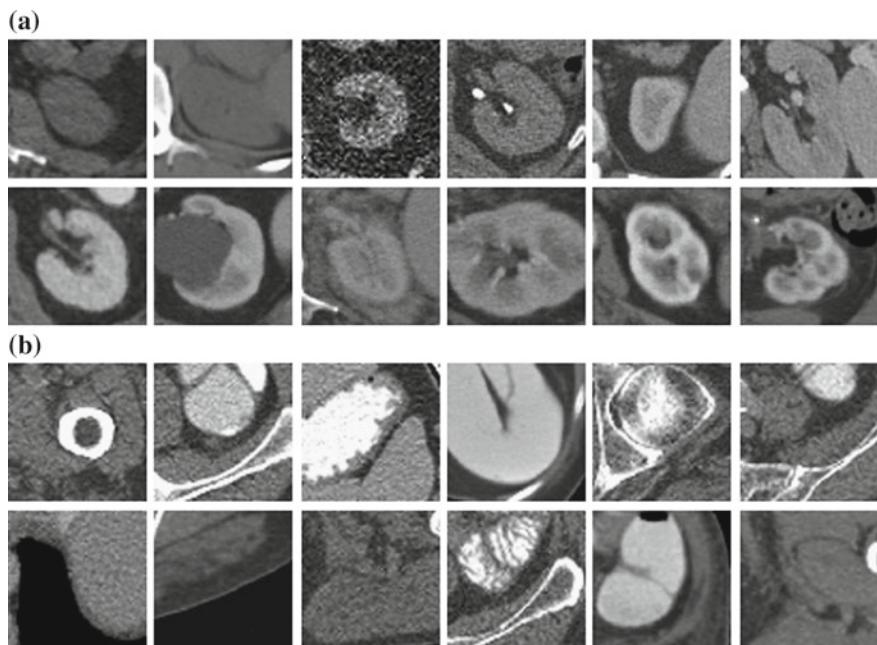


Fig. 14.5 A few **a** positive and **b** negative training patches of the *left* kidney scanned with different contrast phases. Some negative patches are quite similar to positive patches (e.g., the last negative patch versus the first two positive patches)

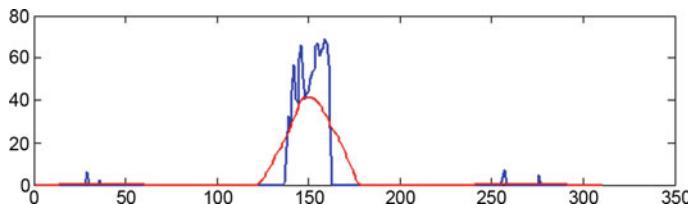


Fig. 14.6 Determining an axial slice containing the kidney center. The *blue curve* shows the aggregated classification score from multiple patches and the *red curve* shows the score after smoothing. The maximum peak on the *red curve* corresponds to the kidney center

14.5 Kidney Segmentation Based on MSL

After rough localization of a kidney, we use MSL to refine the position and further estimate its orientation and size. MSL is an efficient method for 3D anatomical structure detection and segmentation in various medical imaging modalities. The segmentation procedure is formulated as a two-stage learning problem: object pose estimation and boundary delineation. To accurately localize a 3D object, nine pose parameters need to be estimated (three for translation, three for orientation, and three

for anisotropic scaling). The object pose estimation is split into three steps: position estimation, position–orientation estimation, and position–orientation–size estimation. After each step only a small number of promising pose hypotheses are kept; therefore, the pose parameter space is pruned significantly to increase the detection efficiency. Since the kidney center has already been roughly determined using a ConvNet (Sect. 14.4), we constrain the MSL position search to a neighborhood around the initial center.

After the MSL-based pose estimation, a mean shape is aligned to the estimated transformation to generate a rough estimate of the kidney shape. We then deform the shape to fit the kidney boundary using a machine learning based boundary detector within the ASM framework. Interested readers are referred to [7] for more details of the MSL-based object detection and segmentation.

14.6 Experiments

We treat the left and right kidney as different organs and train separate models to detect/segment them. The systems are trained on 370 patients and tested on 78 patients (each patient contributes one CT scan). Our dataset is very diverse, containing various contrast phases and scanning ranges. Many patients have tumors inside the kidney or neighboring organs and some patients have previous abdominal surgery. The axial slice size is 512×512 pixels and the in-slice resolution varies from 0.5 to 1.5 mm, with a median resolution of 0.8 mm. The number of axial slices varies from 30 to 1239. The distance between neighboring slices varies from 0.5 to 7.0 mm with a median of 5.0 mm. On the test set, one patient has the left kidney surgically removed and three patients have the right kidney removed. These patients are ignored when we report the detection/segmentation error of the left and right kidney, respectively.

First, we evaluate the robustness of kidney localization using a ConvNet. There are far more negative training samples than the positives. We randomly subsample the negatives to generate a balanced training set with around 10,000 images for each class. We compare kidney localization errors of three input sizes: a whole slice, a body region, and a patch (85×85 mm 2). Since we only sample one training patch from each slice, the number of training samples is the same for three scenarios, while the size of image context is different.

Tables 14.1 and 14.2 report the kidney center localization errors. The whole-slice-based approach results in the worst performance with mean errors of 86.8 mm (the left kidney) and 113.6 mm (the right kidney) in determining the z-axis position of kidney center. Using the body region as input, we can significantly reduce the mean z-axis localization errors to 14.5 mm (the left kidney) and 17.8 mm (the right kidney). The patch-wise classification achieves the best result with mean z-axis localization errors of 6.8 mm (the left kidney) and 7.9 mm (the right kidney). In addition, it can accurately estimate the x and y position of the kidney center, with a mean error ranging from 2.0 to 3.1 mm. The larger mean errors in the z-axis are due to its much coarser resolution (a median resolution of 5.0 mm in z vs. 0.8 mm in x/y). For the left

Table 14.1 Left kidney localization errors on 78 test cases with different input image context sizes

	X		Y		Z	
	Mean	Max	Mean	Max	Mean	Max
Whole slice	–	–	–	–	86.8	557.5
Body region	–	–	–	–	14.5	112.5
Body region (Multi)	–	–	–	–	12.0	45.7
Patch	2.2	11.6	2.0	14.5	6.8	31.1

Table 14.2 Right kidney localization errors on 78 test cases with different input image context sizes

	X		Y		Z	
	Mean	Max	Mean	Max	Mean	Max
Whole slice	–	–	–	–	113.6	631.5
Body region	–	–	–	–	17.8	138.7
Body region (Multi)	–	–	–	–	12.9	101.7
Patch	3.1	46.9	3.0	17.5	7.9	56.7

kidney localization, the maximum z-axis error is 31.1 mm. We checked this case and found that the estimated position was still inside the kidney. (Please note, a typical kidney has a height of 100 mm along the z-axis.) For the right kidney localization, there is one case that the estimated center is slightly outside the kidney. This error can be corrected later in the constrained position estimation by MSL.

For the patch-based approach, we perform classification on 36 patches for each slice. One may suspect that its better performance comes from the aggregation of multiple classifications. To have a fair comparison, we also perform multiple classifications for the body region by shifting its center on a 6×6 grid (the same size as the patch grid). The results are reported as “Body Region (Multi)” in Tables 14.1 and 14.2. Aggregating multiple classifications improves the localization accuracy, but it is still worse than the proposed patch-based approach. This experiment shows that local image context is more robust than global context in pathological kidney detection.

After rough localization of the kidney center using a ConvNet, we apply MSL to further estimate the nine pose parameters, followed by detailed boundary delineation using a discriminative ASM. Based on the error statistics in Tables 14.1 and 14.2, we constrain the MSL position search to a neighborhood of $[-50, 50] \times [-20, 20] \times [-50, 50]$ mm³ around the initial estimate. As shown in Table 14.3, we achieve a mean mesh segmentation error of 2.6 and 1.7 mm for the left and right kidney, respectively. The larger mean error of the left kidney is due to a case with a segmentation error of 24.7 mm. For comparison, without constraint, the mean segmentation errors of MSL

Table 14.3 Kidney mesh segmentation errors on 78 test cases using marginal space learning with/without constrained position search range. The mesh errors are measured in millimeters, the smaller the better

	Mean	Std	Median	Worst	Worst 10%
Left kidney: unconstrained	9.5	38.1	1.3	236.9	79.5
Left kidney: constrained	2.6	4.2	1.5	24.7	11.6
Right kidney: unconstrained	6.7	27.9	1.4	220.4	51.2
Right kidney: constrained	1.7	1.2	1.4	6.8	4.6

Table 14.4 Dice coefficient of kidney segmentation on 78 test cases using marginal space learning with/without constrained position search range. The Dice coefficient is in [0, 1], the larger the better

	Mean	Std	Median	Worst	Worst 10%
Left kidney: unconstrained	0.86	0.24	0.94	0.00	0.21
Left kidney: constrained	0.89	0.15	0.93	0.11	0.54
Right kidney: unconstrained	0.88	0.19	0.93	0.00	0.46
Right kidney: constrained	0.92	0.05	0.94	0.73	0.79

are much larger due to some gross detection failures. The difference in the mean error of the worst 10% cases is more prominent: 11.6 mm versus 79.5 mm for the left kidney and 4.6 mm versus 51.2 mm for the right kidney. In Table 14.4, we also report the Dice coefficient. Unconstrained MSL has six gross failures (the segmentation has no overlap with ground truth resulting in a Dice coefficient of 0). All the failures are corrected by the proposed method.

It is hard to compare our errors with those reported in the literature due to the lack of a common test set. Lay et al. [6] reported that MSL outperformed their regression-based approach on kidney detection in 3D MRI scans. Here, we achieve further improved robustness upon MSL. Cuingnet et al. [5] reported 6% of cases with Dice <0.65, while we have only three kidneys (2%) with Dice <0.65.

Our approach is fully automatic and takes about 3.3 s to detect a kidney: Kidney localization takes 2.8 s/volume on an NVIDIA GTX 980 GPU; The MSL detection/segmentation step takes 0.5 s on a computer with an Intel Xeon 6-core 2.6 GHz CPU and 32 GB memory (no use of GPU). Figure 14.1 shows segmentation results on a few cases and more examples are shown in Fig. 14.7.

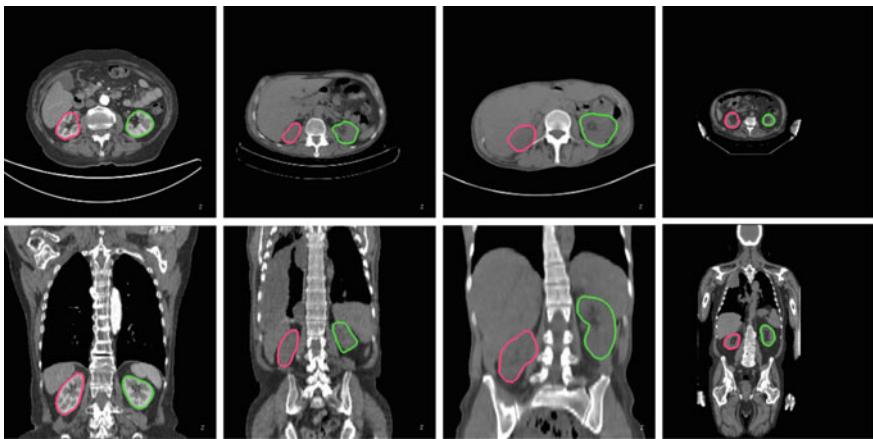


Fig. 14.7 A few examples of segmentation results of the *left* (green) and *right* (red) kidney. An axial view (*top*) and a coronal view (*bottom*) are shown for each example

14.7 Conclusions

In this paper, we proposed a robust fully automatic method for pathological kidney segmentation in CT scans. Deep learning is exploited to roughly estimate the kidney center, which is used to constrain the detection by MSL. We show that local image context (small patches) is more robust than global context (whole slice or body region) in kidney detection and the proposed approach significantly reduces the number of gross failures. Our method works for renal CT data with different contrast phases, scanning ranges, and pathologies.

References

1. Center for Disease Control and Prevention (2014) National chronic kidney disease fact sheet. http://www.cdc.gov/diabetes/pubs/pdf/kidney_factsheet.pdf
2. Yuh BI, Cohan RH (1999) Different phases of renal enhancement: role in detecting and characterizing renal masses during helical CT. Am J Roentgenol 173(3):747–755
3. Yang G, Gu J, Chen Y, Liu W, Tang L, Shu H, Toumoulin C (2014) Automatic kidney segmentation in CT images based on multi-atlas image registration. In: Proceedings of the international conference on IEEE engineering in medicine and biology society, pp 5538–5541
4. Criminisi A, Shotton J, Robertson D, Konukoglu E (2011) Regression forests for efficient anatomy detection and localization in CT studies. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 106–117
5. Cuignet R, Prevost R, Lesage D, Cohen LD, Mory B, Ardon R (2012) Automatic detection and segmentation of kidneys in 3D CT images using random forests. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 66–74

6. Lay N, Birkbeck N, Zhang J, Zhou SK (2013) Rapid multi-organ segmentation using context integration and discriminative models. In: Proceedings of the information processing in medical imaging, pp 450–462
7. Zheng Y, Barbu A, Georgescu B, Scheuering M, Comaniciu D (2008) Four-chamber heart modeling and automatic segmentation for 3D cardiac CT volumes using marginal space learning and steerable features. *IEEE Trans Med Imaging* 27(11):1668–1681
8. Zheng Y, Comaniciu D (2014) Marginal space learning for medical image analysis – efficient detection and segmentation of anatomical structures. Springer, Berlin
9. Thong W, Kadoury S, Piche N, Pal CJ (2015) Convolutional networks for kidney segmentation in contrast-enhanced CT scans. In: Proceedings of workshop on deep learning in medical image analysis, pp 1–8
10. Yan Z, Zhan Y, Peng Z, Liao S, Shinagawa Y, Metaxas DN, Zhou XS (2015) Bodypart recognition using multi-stage deep learning. In: Proceedings of the information processing in medical imaging, pp 449–461
11. Zheng Y, Liu D, Georgescu B, Nguyen H, Comaniciu D (2015) 3D deep learning for efficient and robust landmark detection in volumetric data. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 565–572
12. Liu F, Yang L (2015) A novel cell detection method using deep convolutional neural network and maximum-weight independent set. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 349–357
13. Roth HR, Lu L, Seff A, Cherry KM, Hoffman J, Wang S, Liu J, Turkbey E, Summers RM (2014) A new 2.5D representation for lymph node detection using random sets of deep convolutional neural network observations. In: Proceedings of the international conference on medical image computing and computer assisted intervention, pp 520–527
14. Carneiro G, Nascimento JC, Freitas A (2012) The segmentation of the left ventricle of the heart from ultrasound data using deep learning architectures and derivative-based search methods. *IEEE Trans Image Process* 21(3):968–982
15. Ghesu FC, Krubasik E, Georgescu B, Singh V, Zheng Y, Hornegger J, Comaniciu D (2016) Marginal space deep learning: efficient architecture for volumetric image parsing. *IEEE Trans Med Imaging* 35:1217
16. Cheng X, Zhang L, Zheng Y (2016) Deep similarity learning for multimodal medical images. *Comput Methods Biomed Eng Imaging Vis*
17. Miao S, Wang ZJ, Zheng Y, Liao R (2016) Real-time 2D/3D registration via CNN regression. In: Proceedings of the IEEE international symposium on biomedical imaging, pp 1–4
18. Zheng Y, Doermann D (2005) Handwriting matching and its application to handwriting synthesis. In: International conference on document analysis and recognition, pp 1520–5263
19. Zheng Y (2015) Cross-modality medical image detection and segmentation by transfer learning of shape priors. In: Proceedings of the IEEE international symposium on biomedical imaging, pp 424–427
20. Bookstein F (1989) Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Trans Pattern Anal Mach Intell* 11(6):567–585
21. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: convolutional architecture for fast feature embedding. [arXiv:1408.5093](https://arxiv.org/abs/1408.5093)
22. Seifert S, Barbu A, Zhou K, Liu D, Feulner J, Huber M, Suehling M, Cavallaro A, Comaniciu D (2009) Hierarchical parsing and semantic navigation of full body CT data. In: Proceedings of SPIE medical imaging, pp 1–8

Chapter 15

Robust Cell Detection and Segmentation in Histopathological Images Using Sparse Reconstruction and Stacked Denoising Autoencoders

Hai Su, Fuyong Xing, Xiangfei Kong, Yuanpu Xie, Shaoting Zhang
and Lin Yang

Abstract Computer-aided diagnosis (CAD) is a promising tool for accurate and consistent diagnosis and prognosis. Cell detection and segmentation are essential steps for CAD. These tasks are challenging due to variations in cell shapes, touching cells, and cluttered background. In this paper, we present a cell detection and segmentation algorithm using the sparse reconstruction with trivial templates and a stacked denoising autoencoder (sDAE) trained with structured labels and discriminative losses. The sparse reconstruction handles the shape variations by representing a testing patch as a linear combination of bases in the learned dictionary. Trivial templates are used to model the touching parts. The sDAE, trained on the original data with their structured labels and discriminative losses, is used for cell segmentation. To the best of our knowledge, this is the first study to apply sparse reconstruction and sDAE with both structured labels and discriminative losses to cell detection and segmentation. It is observed that structured learning can effectively handle weak or misleading edges, and discriminative training encourages the model to learn groups of filters that activate simultaneously for different input images to ensure better segmentation. The proposed method is extensively tested on four data sets containing more than 6000 cells obtained from brain tumor, lung cancer, and breast cancer and neuroendocrine tumor (NET) images. Our algorithm achieves the best performance compared with other state of the arts.

H. Su · X. Kong · Y. Xie · L. Yang (✉)
J. Crayton Pruitt Family Department of Biomedical Engineering,
University of Florida, Gainesville, FL 32611, USA
e-mail: Lin.Yang@bme.ufl.edu

F. Xing · L. Yang
Department of Electrical and Computer Engineering, University of Florida,
Gainesville, FL 32611, USA

S. Zhang
Department of Computer Science, University of North Carolina at Charlotte,
Charlotte, NC 28223, USA

15.1 Introduction

Reproducible and accurate analysis of digitized histopathological specimens plays a critical role in successful diagnosis and prognosis, treatment outcome prediction, and therapy planning. Manual analysis of histopathological slides is not only laborious, but also subject to interobserver variability. Computer-aided diagnosis (CAD) is a promising solution. In CAD, cell detection and segmentation are often prerequisite steps for critical morphological analysis [1, 2]. Cell detection reveals the locations of individual cells and cell segmentation separates individual cells from their surrounding cells and the background. Cell segmentation is essential to the subsequent diagnostic analysis procedures since it identifies the geometric information of the cells (i.e., boundary, shape, and size of the cells) that can be used to compute the disease characterizing visual information for diagnosis [3].

Accurate cell detection and segmentation in digital pathology has been attracting a wide range of interests recently [4]. The major challenges in cell detection and segmentation are: (1) large variations of cell shapes, (2) touching cells, and (3) inhomogeneous intensity and weak/missing boundaries. In early studies, distance transform was used to detect seeds (cells) in clustered objects. However, it falls short in handling densely clustered cells. Later, geometric and intensity information are exploited to improve the distance transform methods [5]. Despite of the improvement, this method is subject to high false detection rate. In [6], mutual proximity information is used to exclude the false seeds. Another approach to handle touching/occlusion cells is marker-based watershed algorithms [7–9]. In [10], markers generated from H-minima transform of cell shape is proposed. The H-value is determined by the fitting residuals between the ellipses and cell boundaries. However, H-minima transform-based methods is not robust enough against the intracellular heterogeneous intensity. In [11], a supervised marker-controlled watershed algorithm is investigated. The redundant neighboring local minima are merged based on the features extracted from the valley lines between the cells. Although this supervised method provides some degree of robustness, in the presence of intracellular heterogeneity false valley lines could be detected inside the cells, thereby causing false detections. In [12], Lin et al. propose a gradient-weight-based watershed algorithm integrated with a hierarchical merging tree for splitting touching cells for 3D confocal microscopic images. However, in their work the intra-class shape variation is not considered. In [13], a subregion merging mechanism and a Laplacian-of-Gaussian (LOG) filter are exploited to improve the conventional watershed algorithm. A cell detection and segmentation system for RNAi fluorescent cell images are presented in [14]. The visual cues from different channels of the fluorescent image are utilized to split the overlapping cells.

Graph-based method have also been explored for cell detection and segmentation [15, 16]. In [17], based on a weighted graph, cell detection is formulated as a normalized cut-based graph partition problem. In [18], a graph-cut algorithm preceding by multiscale LOG filtering is investigated. In [19], a multi-reference graph-cut algorithm is used to split the foreground cells, and touching cells are split

through geometric reasoning. Despite the endeavors, graph-based algorithms might be not robust to weak cell boundaries. Recently, deep learning methods are observed to exhibit significant advantages in visual recognition tasks [20]. A deep convolutional neural network (DCNN) [21] achieves great performance in mitosis detection. However, this system does not consider touching cells problem that is common in pathology images.

Other methods for touching cell detection and segmentation is to exploit the symmetries in cell structure and shape. Based on the assumption that most cells exhibit round shape, radial voting methods are proposed to robustly separate touching cells [22, 23]. In [24], radial symmetry is integrated with other image cues (i.e., concave points) to separate touching cells. In [25], Veta et al. propose to detect the cells based on fast radial symmetry transform and segment the cells based on marker-controlled watershed algorithm. These methods achieve good performance on images containing mostly round shaped cells. However, in the presence of elongated shaped cells, voting methods are subject to high false positive rate. To handle the shape variation, ellipse fitting based on the concave points is studied in [26]. To enhance the performance, ellipse fitting followed by feature extraction, and classification is proposed to split the muscle nuclei [27]. In [28], a single-pass radial voting followed by mean-shift clustering is proposed to split the touching cells and a repulsive level set is developed for cell segmentation. One drawback of the level set is that it does not enforce the original topology of the object thus can generate spurious segmentation contours in the presence of heterogeneous intensity. In [29], an improved radial voting is proposed for cell detection with considering the variations in cell scales and a repulsive balloon snake deformable model is applied to cell segmentation. Recently, shape prior model is proposed to improve the performance in the presence of weak edges [30, 31].

Sparse representation has achieved encouraging performance in object detection and tracking [32–35]. Its applications in biomedical images can be found in [36, 37]. In [38], Kårsnäs et al. propose to learn a patch dictionary through a modified vector quantization algorithm. The learned dictionary is used to delineate the foreground. The touching cells in the foreground are separated by the marker-controlled watershed and a complement to the distance transform. In [31, 39], sparse shape modeling is cooperated with repulsive active contour models for robust cell segmentation. Sparse methods are also exploited to learn useful features for classifying histology images [40–42]. Despite the exiting efforts, the advantages of sparse learning has not been systematically explored for the cell detection task.

In this paper, we propose a novel cell detection and segmentation algorithm. First, sparse reconstruction using an adaptive dictionary and trivial templates is proposed to detect cells, which can handle the shape variations, inhomogeneous intensity, and cell overlapping. Thereafter, a stacked denoising autoencoder (sDAE) trained with structural labels is applied to cell segmentation based on the previous cell detection. Traditionally, denoising autoencoders are trained on corrupted samples to learn robust features for classification tasks [43–45]. They require “clean” images as a premise, but this is difficult to achieve in pathology images due to their noisy nature. In the proposed method, the noisy original images and their human annotated structural

labels are used as training samples. The sDAE model is trained to map the original image into a reconstructed boundary image. In our experiment, it is observed that the structural labels enforce the first layer of the model to learn a set of filters capturing the object components (e.g., edges and blob-like patterns). In testing, different subset of the filters respond to the objects with different shapes. The second layer of the model learns to capture the correlation between the filters in the first layer, and serves to correct the mistakenly responding filters. A problem in the sDAE is that the filters in the first layer is merely a coarse-grained decomposition of the edges and noise. The learned filters may be shared between objects of different shapes. This introduces ambiguity into the correlation between the filters and makes it hard for the second layer to learn such correlation. The resultant model shows limited robustness against to noise. Based on this observation, we propose to add a discriminative loss into the cost function. The obtained model is referred as stacked discriminative denoising autoencoder (sdDAE). The discriminative training encourages the model to learn a fine-grained decomposition of the object components with respect to the categories. Therefore, the first layer of the sdDAE contains a set of filters corresponds to image structures belong to different categories and noise. With these learned class-specific filters, noticeable improvement in group activation is achieved and more robust segmentation performance is obtained.

This article is an extension of our previously published work [46]. The previous work is significantly extended in three aspects: (i) A new discriminative term is added into the cost function of the sDAE; (ii) The working mechanism of the proposed segmentation algorithm is elaborated; (iii) Two more new data sets are used to validate the superior performance of the proposed method.

15.2 Methodology

An overview of the proposed method is shown in Fig. 15.1. During the training for cell detection, a compact cell dictionary (Fig. 15.1b) is learned by applying K-selection [47] to a cell patch repository containing single-centered cells. In the testing (Fig. 15.1a–e), a sample patch from the testing image is first used as a query to retrieve similar patches in the learned dictionary. Since the appearance variation within one particular image is small, any sample patch containing a centered cell can be used. Next, sparse reconstruction using *trivial templates* [34] is utilized to generate a probability map to indicate the potential locations of the cells. Finally, weight-guided mean-shift clustering [48] is used to compute the seed detection. Different from [34], our algorithm removes the sparsity constraints for the trivial templates. Therefore, the proposed method is more robust to the variations of the cell size and background. During the segmentation stage (Fig. 15.1f–i), the sDAE is trained using the gradient maps of the training patches and their corresponding human annotated edges (Fig. 15.1f). Our proposed segmentation algorithm is designed to handle

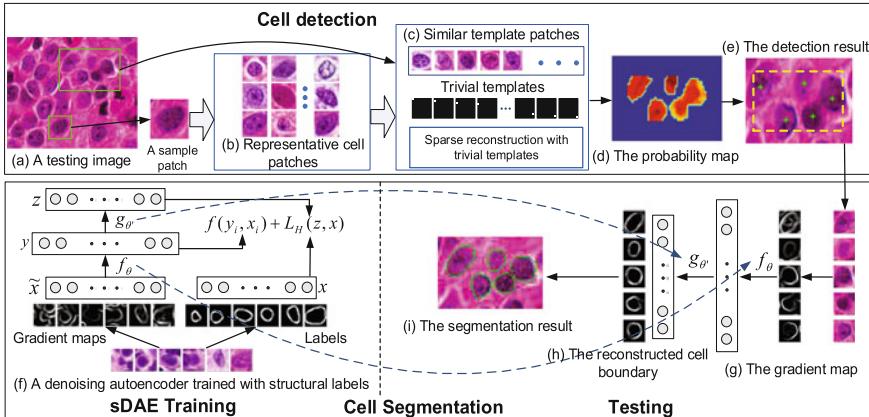


Fig. 15.1 An overview of the proposed algorithm

touching cells and inhomogeneous cell intensities. As shown in Fig. 15.1h, the false edges are removed, the broken edges are connected, and the weak edges are recovered.

15.2.1 Detection via Sparse Reconstruction with Trivial Templates

Adaptive Dictionary Learning: During cell dictionary learning, a set of relevant cell patches are first retrieved based on their similarities compared with the sample patch. Considering the fact that pathological images commonly exhibit staining variations, the similarities are measured by normalized *local steering kernel* (nLSK) feature and *cosine similarity*. nLSK is more robustness to contrast changes [49]. An image patch is represented by the densely computed nLSK features. Principal component analysis (PCA) is used for dimensionality reduction, as suggested by [49]. Cosine distance, $D_{cos} = (\mathbf{v}_i^T \mathbf{v}_j) / (\|\mathbf{v}_i\| \|\mathbf{v}_j\|)$, where \mathbf{v}_i denotes the nLSK feature of patch i , is proven to be the optimal similarity measurement under maximum likelihood decision rule [49]. Therefore, it is used to measure the similarity. The dictionary patches are selected by a nearest neighbor search.

Probability Map Generation via Sparse Reconstruction with Trivial Templates: Given a testing image, we propose to utilize sparse reconstruction to generate the probability map by comparing the reconstructed image to the original patch via a sliding window approach. Because the testing image patch may contain part of other neighboring cells, trivial templates are utilized to model these noise parts. When the testing patch is aligned to the center of a cell, it can be linearly represented by cell dictionary bases with a small reconstruction error. The touching part can be modeled

with trivial templates. Let $\mathbf{p}_{ij} \in \mathbb{R}^{\sqrt{m} \times \sqrt{m}}$ denote a testing patch located at (i, j) , and \mathbf{B} represent the learned cell dictionary, this patch can be sparsely reconstructed by: $\mathbf{p}_{ij} \approx \mathbf{B}\mathbf{c} + \mathbf{e} = [\mathbf{B} \ \mathbf{I}][\mathbf{c} \ \mathbf{e}]^T$, where \mathbf{e} is the term to model the touching part, and $\mathbf{I}_{m \times m}$ is an identity matrix containing the trivial templates. The optimal sparse reconstruction can be found by:

$$\min_{\tilde{\mathbf{c}}} \|\mathbf{p}_{ij} - \tilde{\mathbf{B}}\tilde{\mathbf{c}}\|^2 + \lambda \|\mathbf{d} \odot \mathbf{c}\|^2 + \gamma \|\mathbf{e}\|^2, \text{ s.t. } \mathbf{1}^T \mathbf{c} = 1, \quad (15.1)$$

where $\tilde{\mathbf{B}} = [\mathbf{B} \ \mathbf{I}]$, $\tilde{\mathbf{c}} = [\mathbf{c} \ \mathbf{e}]^T$, and \mathbf{d} represents the distance between the testing patch and the dictionary atoms, \odot denotes element-wise multiplication, λ controls the importance of the locality constraints, and γ controls the contribution of the trivial templates. The first term incorporates trivial templates to model the touching cells, and the second term enforces that only local neighbors in the dictionary are used for the sparse reconstruction. The locality constraint enforces sparsity [33]. In order to solve the locality-constrained sparse optimization, we first perform a KNN search in the dictionary excluding the trivial templates. The selected nearest neighbor bases together with the trivial templates form a smaller local coordinate system. Next, we solve the sparse reconstruction problem with least square minimization [33].

The reconstruction error is defined as $\epsilon_{rec} = \|(\mathbf{p}_{ij} - \tilde{\mathbf{B}}\tilde{\mathbf{c}}) \odot k(u, v)\|$, where $k(u, v)$ is a “bell-shape” spatial kernel that emphasizes the errors in the central region. A probability map is obtained by $P_{ij} = \frac{|\epsilon_{rec} - \max(E)|}{\max(E) - \min(E)}$, where P_{ij} denotes the probability at location (i, j) , and E represents the reconstruction error map. We demonstrate the reconstruction results of touching cells with and without trivial templates in Fig. 15.2a, b. The final cell detection is obtained by running a weight-guided mean-shift clustering [48] on the probability map.

15.2.2 Cell Segmentation via Stacked Denoising Autoencoders

In this section, we propose to train a stacked denoising autoencoder (sDAE) [45] with structural labels to remove the fake edges while preserving the true edges. An overview of the training and testing procedure is shown in Fig. 15.1f-i. Traditionally, denoising autoencoders (DAE) are trained with corrupted versions of the original samples, and it requires “clean image” as a premise. In our proposed method, we use the gradient images of original image patches as the noisy inputs and the human annotated boundaries (structured labels) as the clean images. The DAE is trained to map a noisy input to a clean (recovered) image patch that can be used for segmentation.

For better illustration, we describe a single layer DAE. Let $\tilde{\mathbf{X}} \in \mathbb{R}^m$ denote the noisy gradient magnitude map of the original image patch centered on a detected center of the cell (seed). The DAE learns a parametric encoder function $f_\theta(\tilde{\mathbf{x}}) = s(\mathbf{W}\tilde{\mathbf{x}} + \mathbf{b})$, where $s(\cdot)$ denotes the sigmoid function to transform the input from the original feature space into the hidden layer representation $\mathbf{y} \in \mathbb{R}^h$, where $\theta = \{\mathbf{W}, \mathbf{b}\}$

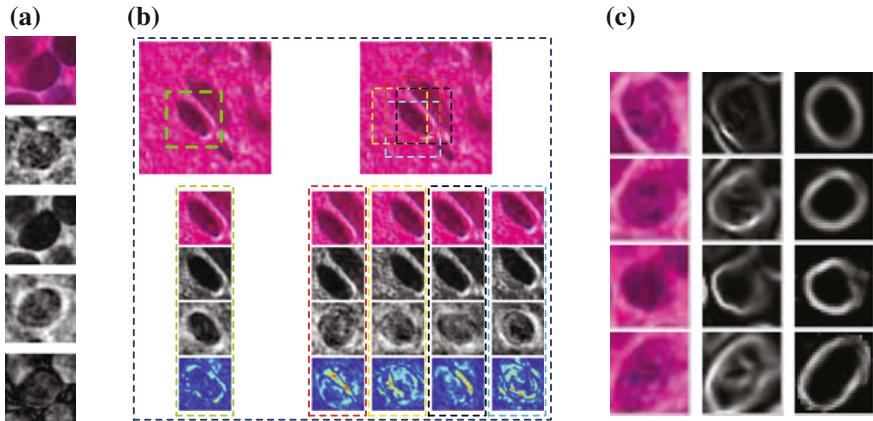


Fig. 15.2 **a** A demonstration of sparse reconstruction with/without trivial templates. From row 1 to 3: a testing patch, the sparse reconstruction without trivial templates, the sparse reconstruction with trivial templates. Row 4 and 5 are the first term and the second term in equation $\mathbf{p}_{ij} \approx \mathbf{B}\mathbf{c} + \mathbf{e}$, respectively. **b** A demonstration of reconstruction errors obtained from a testing patch aligned to the center of the cell and from those misaligned patches. Row 1 displays a small testing image. The green box shows a testing patch aligned to the cell. Boxes in other colors show misaligned testing patches. From row 2 to row 5: A testing image patch with occlusion from a neighboring cell, the reconstruction of the testing patch, the reconstructed patches with the occlusion part removed, and the visualization of the reconstruction errors. Note that the aligned testing patch has the smallest error. **c** From left to right the original testing patches, the gradient magnitude maps, and the recovered cell boundaries using sDAE

and $\mathbf{W} \in \mathbb{R}^{h \times m}$. A parametric decoder function $g_{\theta'}(\mathbf{y}) = s(\mathbf{W}'\mathbf{y} + \mathbf{b}')$, $\theta' = \{\mathbf{W}', \mathbf{b}'\}$ is learned to transform the hidden layer representation back to a reconstructed version $\mathbf{Z} \in \mathbb{R}^m$ of the input $\tilde{\mathbf{X}}$.

Since it is a reconstruction problem based on real-valued variables, a square error loss function of the reconstruction \mathbf{z} and a manually annotated structural label \mathbf{x} is chosen, and the sigmoid function in $g_{\theta'}$ is omitted. The parameters $\{\theta, \theta'\}$ are obtained by:

$$\min_{\mathbf{W}, \mathbf{b}, \mathbf{W}', \mathbf{b}'} \|\mathbf{x} - g_{\theta'} \circ f_{\theta}(\tilde{\mathbf{X}})\|^2. \quad (15.2)$$

We choose *tied weights* by setting $\mathbf{W}' = \mathbf{W}^T$ [45]. In order to restore a reliable edge image that enhances true edge responses and suppresses fake edge responses, we train a two-layer autoencoder in the experiment (see Fig. 15.2). The final segmentation results can be obtained by applying several iterations of an active contour model [50] to the convex hull computed from the reconstructed image.

15.2.3 The Learned Filters

We demonstrate the mechanism of a two-layer autoencoder for cell segmentation in Fig. 15.3. We trained a two-layer autoencoder with 400 units in the first layer and 200 units in the second layer using lung cancer data set. An example cell patch with weak edges and inhomogeneous intracellular intensity is shown in Fig. 15.3a.

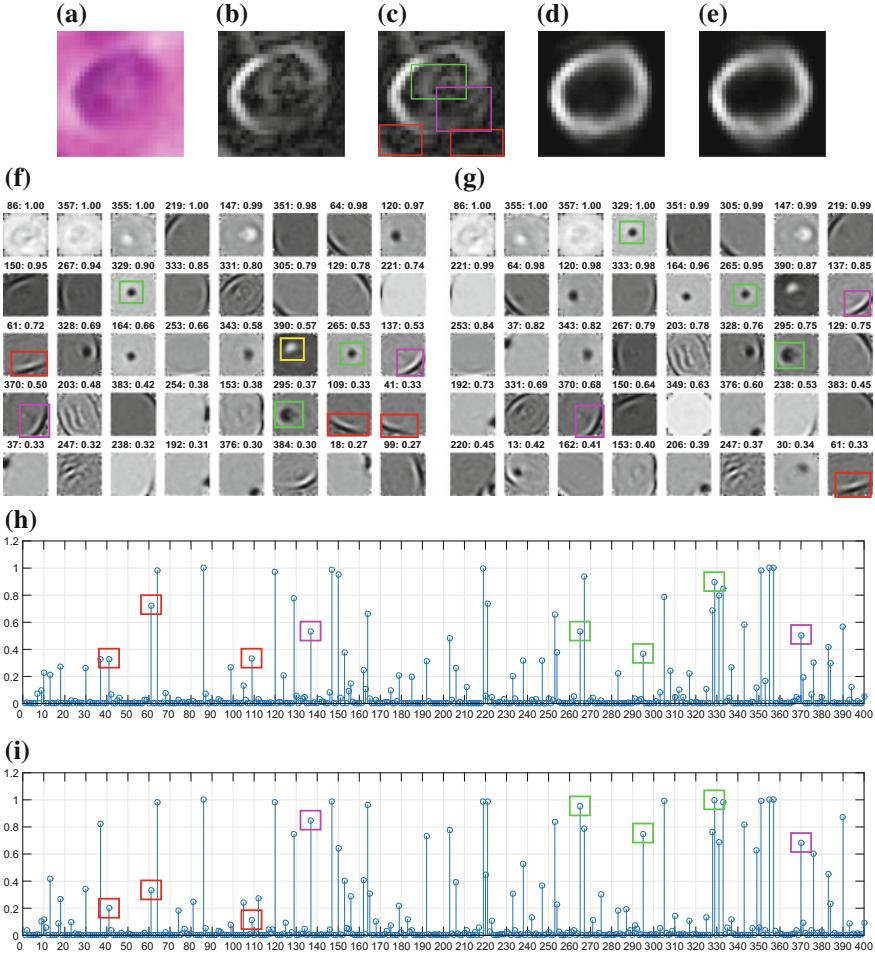


Fig. 15.3 **a** The original image patch. **b** The noisy input image patch (gradient magnitude image of a). **c** The noisy input image with highlighted regions containing weak edges and noise. The restored cell edge images in the **d** first and **e** second layers. **f** The filters with the strongest responses in the first layer. The integer above each filter indexes the filter. The real number shows the magnitude of the response. **g** The filter responses restored by the second layer. **h** The response of the first layer filters. **i** The first layer responses restored in the second layer

Its gradient magnitude image is shown in Fig. 15.3b. In this example, we show how the learned model suppresses the noise (the green box and the red boxes in Fig. 15.3c) and recovers the weak edges (pink box in Fig. 15.3c). The restored images by the first and second layers of the autoencoder are shown in Fig. 15.3d, e, respectively. The 40 filters with the strongest responses in the first layer are shown in Fig. 15.3f. As we can see, the learned filters are a decomposition of the patterns present in a patch containing a cell. Different filters are sensitive to different parts of the cell. The dark blobs in the central regions of the filters, (e.g., filters 120, 329, 328, 164, 343, 265, 295), respond to the relatively dark color inside the cell, and the edge filters, (e.g., 219, 64, 150, 267, 331, 305, 129, and etc.), respond to the cell boundaries. It is worth noting that there are several activated white blob filters (e.g., filter 390) due to the noise present inside the cell (green box in Fig. 15.3c). Meanwhile, there are several edge filters responding to the fake edges (e.g., filters 61, 109, and 41).

The second layer is learned to improve the restoration. As shown in Fig. 15.3g, i, the responses of all the dark blob filters are elevated. Specifically, the responses of the dark blob filters (green box in Fig. 15.3g) are enhanced to counteract the effect of the filter 390 (yellow box in Fig. 15.3f). More importantly, the responses of the edge filters (i.e., 137 and 370) are increased to make up the weak edge (pink boxes). Meanwhile, the responses of the filters corresponding to the fake edges (i.e., 61, 109, and 41) are decreased (red boxes). All the responses in the first and second layers are shown in Fig. 15.3h, i, respectively. The responses of the corresponding filters mentioned above are highlighted with the same colors in Fig. 15.3.

15.2.4 Training DAE with Discriminative Loss

Although the segmentation method described above is able to handle the touching cells, it tends to generate distorted segmentation when two non-touching cells locate close to each other. This problem is illustrated in Fig. 15.4, which shows three example image patches. The DAE model trained with Eq. 15.2 tends to cast enlarged contours in the presence of an adjacent cell. This is because the DAE described above is unable to differentiate valid and invalid combinations of the filter activations. Therefore, any strong activation of the filters in the first layer is treated as valid activation such that the undesired strong filter activations cannot be suppressed or removed. This leaves the model to be less robust to noise which might invoke strong filter activations. In order to train a DAE that is robust to neighboring strong noise, we propose to augment the original loss function in Eq. 15.2 by a discriminative term [51]:

$$\min_{\mathbf{W}, \mathbf{b}, \tilde{\mathbf{W}}, \tilde{\mathbf{b}}} \sum_{i=1}^N \|\mathbf{x}_i - g_{\theta'} \circ f_{\theta}(\tilde{\mathbf{x}}_i)\|^2 + \mathcal{L}(\theta_d = \{\mathbf{W}_d, \mathbf{b}_d\}, y_i, f_{\theta}(\tilde{\mathbf{x}}_i)), \quad (15.3)$$

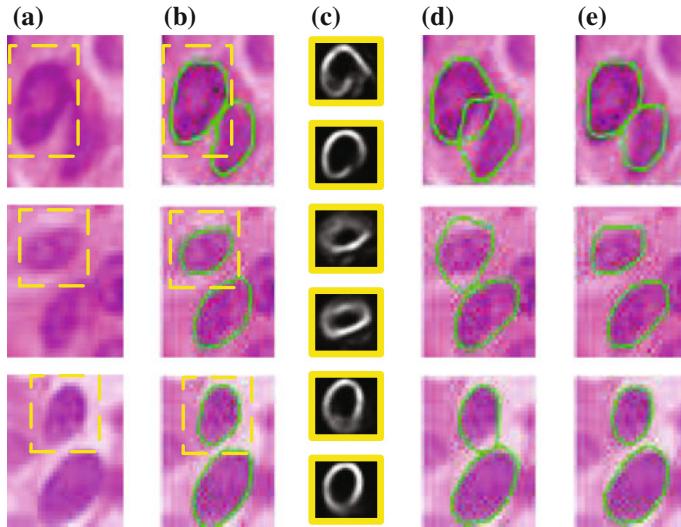


Fig. 15.4 **a** The original image patches. **b** The ground truth segmentation. **c** The restored cell boundaries by models trained without discriminative loss (upper patches) and with discriminative loss (bottom patches). **d** The segmentation results obtained by the model trained without discriminative loss. **e** The segmentation result generated by the model with discriminative term. As can be seen that the restored image patches obtained by the discriminative DAE is more accurate than those obtained by the original DAE. Therefore, the final segmentation contours obtained by the discriminative DAE **e** is more accurate

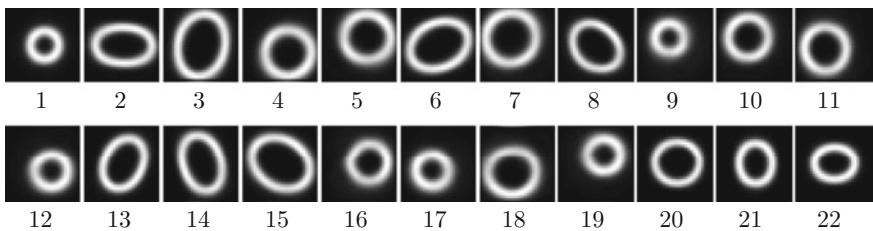


Fig. 15.5 The 22 cluster centers of the structural labels. As indicated by the cluster centers, the samples of different clusters occupy different pixels in the image patch. When training with the discriminative loss, the model can learn the class-specific filters and will be able to differentiate valid and invalid combinations of the filters

where the discriminative loss term is a multi-class logistic regression:

$$\mathcal{L}(\theta_d = \{\mathbf{W}_d, \mathbf{b}_d\}, y_i, f_\theta(\tilde{\mathbf{x}}_i)) = \sum_{i=1}^N \|y_i - (\mathbf{W}_d f_\theta(\tilde{\mathbf{x}}_i) + \mathbf{b}_d)\|^2, \quad (15.4)$$

where $\theta_d = \{\mathbf{W}_d, \mathbf{b}_d\}$ is the parameters of the discriminative loss function, y_i is the class label of data sample $\tilde{\mathbf{x}}_i$.

In our problem, the purpose of labeling the data is to enforce the class (group) information in the learning. That is to enforce the model to learn a set of filters that form the decomposition with respect to the labels. The labels serve as a definition of valid and invalid edge combinations. The label y_i is generated via K-means clustering, and the cluster centers are shown in Fig. 15.5. As we can see, the cluster means implicitly define a set of valid shapes. In the training, filters with class association can be learned from these shapes.

We demonstrate the effects of the discriminative loss in training the DAE in Fig. 15.6. An example image patch and its segmentation results are shown in Fig. 15.6a. The original image patch, noisy input image and the ground truth segmentation are shown in the left panel. The restored images by the first and second layers of a two-layer DAE as well as the segmentation result are depicted in the middle panel. The right panel displays the results obtained by the first and second layers of a two-layer discriminative DAE. As one can tell, the discriminative DAE generates better restoration and segmentation results. This is because the filters learned in the first layer decompose the cell edges and noises with respect to the image labels. This discriminative training grants the model more discriminative power. The second layer implicitly learns the association between the filters and promotes the simultaneous activation or silence of the grouped filters. We can observe the improvements by comparing the Fig. 15.6b, d and c, e. Figure 15.6b shows filters with the strongest activation, including some incorrectly activated filters by the noise. The effects of the second layer are shown in Fig. 15.6c, e. The highlighted filters in Fig. 15.6b, d are deactivated. Meanwhile, the responses of the correct cell boundary associated edge filters are enhanced (green boxes in Fig. 15.6c, e). It is worth noting that filters associated with particular classes emerge in the discriminative DAE. For this case, it is the filter 323 highlighted in green box in Fig. 15.6c.

Compared to the most activating filters of the sdDAE Fig. 15.6, the most activating filters of the DAE Fig. 15.7 do not capture the cell edges. This is because without discriminative losses, the learned filters do not decompose into groups corresponding to different classes (and noise). Some filters may be shared between different classes. These filters have less discriminative power and may not activate when there is too much noise. Due to the suboptimal decomposition, the filters are not classified themselves into different groups, and thus, it is difficult for the second layer to learn the association between the filters. As a result, the second layer in the DAE do not help effectively in correcting the reconstruction.

An analysis of the association of the filters learned in the first and second layers of the denoising autoencoders is depicted in Fig. 15.8. One DAE and one sdDAE are trained with the same structure (400 units in the first layer and 200 in the second layer). In total, 5000 data samples are passed through the two models, and the filter responses in the first and second layers are recorded. The pairwise response similarity of the filters are computed. In general, coactivation of the filters are observed, especially for the sdDAE, and the second layer serves to recover the association between the filters. This can be observed by comparing the response similarity matrix in Fig. 15.8a, b. The response similarity matrix of the restored filter responses contains more low-value entries indicating that the filters activate in groups, especially for the sdDAE.

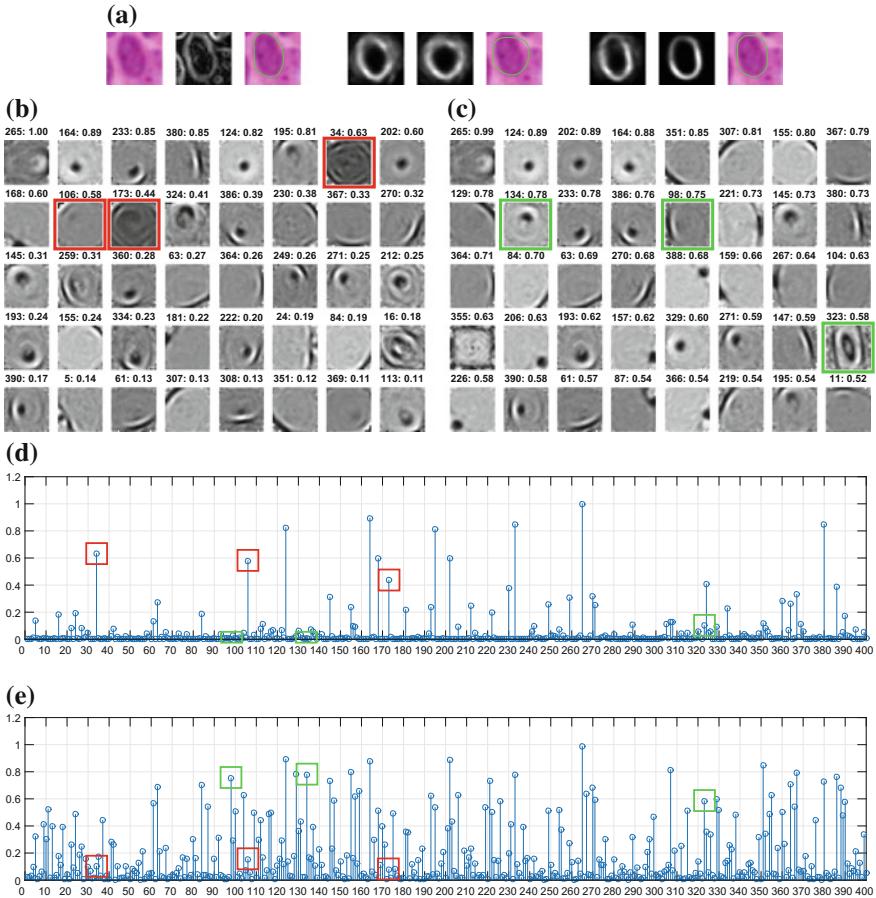


Fig. 15.6 **a** A demonstration of the segmentation results obtained from sDAE and sdDAE. The *left panel* shows the original image patch, the noisy gradient magnitude image, and the ground truth annotation. The *middle panel* displays the reconstruction and segmentation results of the DAE. From *left to right* are: the reconstruction by a one-layer DAE, the reconstruction by a two-layer DAE, and the segmentation result. The *right panel* shows the corresponding results obtained by a sdDAE. The responses of the filters learned in the first **(b)** and second **(c)** layers of the sdDAE. As one can tell, from **a** that the sdDAE generates a better restored edge map and segmentation results. A comparison between **(b, d)** and **(c, e)** reveals that the filters incorrectly activated (*red boxes* in **b** and **d**) by the noise are suppressed, i.e., their responses are decreased significantly (*red boxes* in **e**). On the contrary, the mistakenly silenced filters (*green boxes* in **d**) in the first layer are activated by the second layer of the sdDAE (*green boxes* in **c** and **e**)

This characteristic is resulted by the discriminative training, in which the filters learn a better decomposition of the edges belonging to the cells from difference classes and noise.

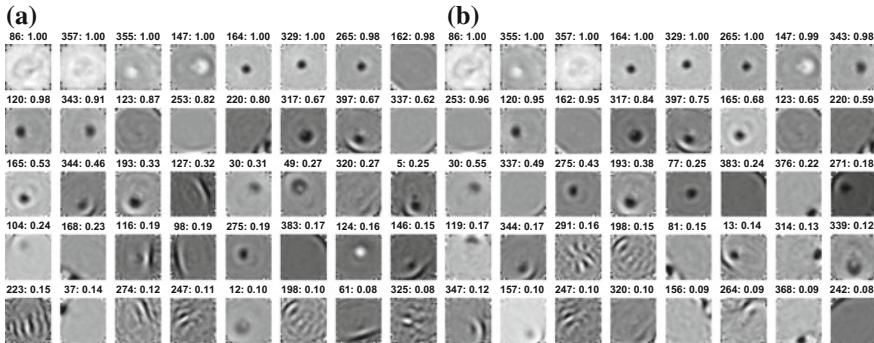


Fig. 15.7 The most activating filters in the first layer of a sDAE for the image patch shown in Fig. 15.6a. Compared to the most activating filters in the sdDAE, less edge filters are triggered. This is because without discriminative losses, the learned filters do not decompose into groups corresponding to different classes (and noise). Some filters may be shared between different classes. These filters have less discriminative power and may not activate when there is too much noise

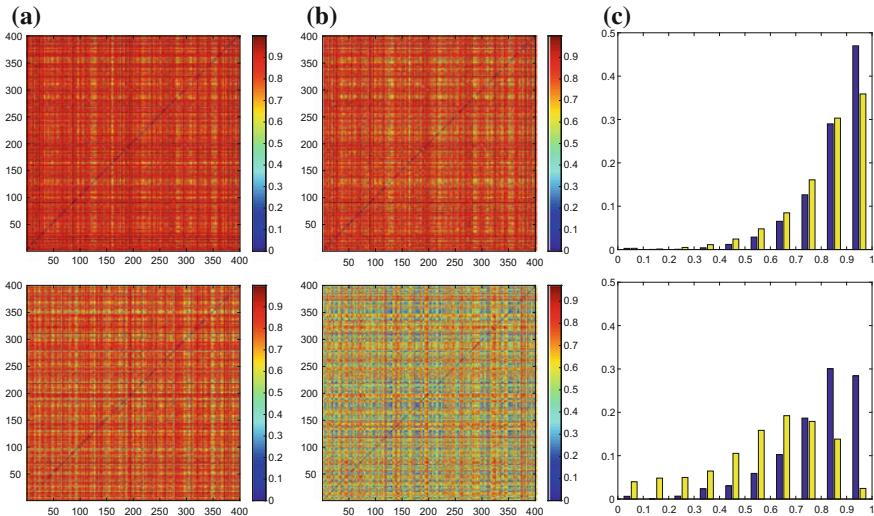


Fig. 15.8 An analysis of the group activation of the learned filters and the role of the second layer of the sdDAE. The upper row shows the results of the DAE and the bottom row depicts the results from the sdDAE. **a** The response similarity matrix computed based on the responses of the 400 first layer filters with respect to 5000 samples. **b** The response similarity matrix computed based on the filter responses restored by the second layer. **c** The histogram of the entry values in the response similarity matrix. The blue bars are computed from the matrix in **a** and the yellow bars are computed from that in **b**. It is obvious that the second layers of both the DAE and the sdDAE learn to capture the association between the filters. That is for the both denoising autoencoders, the restored filter responses show more group association. Specifically, more low-value entries present in the response similarity matrix. Compare to the DAE, the second layer of the sdDAE improves the association significantly, and it can be seen in the changes of the response similarity matrix and the histogram. This is because the filters learned with the discriminative loss form a better decomposition of the edges belonging to cells from different classes and the noise

15.3 Experimental Results

Data set: The proposed algorithm is extensively tested on four data sets including about 2000, 1500, 1500, and 1000 cells in lung cancer, brain tumor, breast cancer, and neuroendocrine tumor (NET) images, respectively. For the detection part, 2000 patches of size 31×31 with a centralized single cell are manually cropped from each data set. $K = 1400$ patches are selected by K-selection. The parameter γ in Eq. (15.1) is set to 10^{-4} . In the segmentation part, contours of more than 6900 cells are annotated. Training sample augmentation is conducted via rotation and random translation. In total more than 16×10^4 image patches are generated and each of them is resized to 28×28 . The samples are clustered into 22 clusters by K-means clustering. The numbers of training samples from each class are balanced. Therefore, in total 11×10^4 samples are used for training. A two-layer sDAE and a two-layer discriminative DAE with 1000 maps in the first layer and 1200 maps in the second layer are trained on each data set. An active contour model [50] is applied to obtain the final segmentation result. All the experiments are implemented with MATLAB and Python Theano Package on a workstation with Intel Xeon E5-1650 CPU and 128 GB memory.

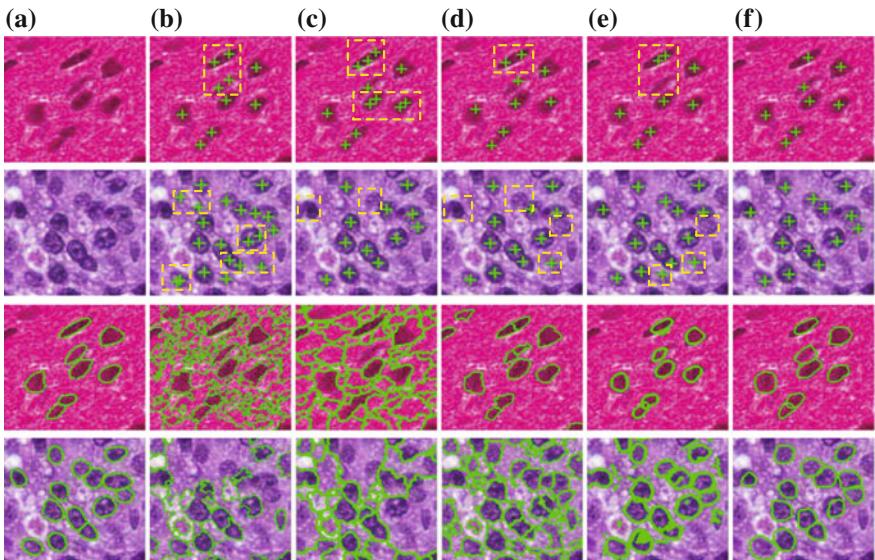


Fig. 15.9 Detection and segmentation results of a testing image. Row 1 shows the comparison of the detection results: **a** is the original image patch. **b–f** are the corresponding results obtained by LoG [18], IRV [23], ITCN [52], SPV [28], and the proposed method. Rows 2 shows the comparison of the segmentation results: **a** is the ground truth. **b–f** are the corresponding results obtained by MS, ISO [53], GCC [18], RLS [28], and the proposed method

Detection Performance Analysis: We evaluate the proposed detection method through both qualitative and quantitative comparison with four state of the arts, including Laplacian-of-Gaussian (LoG) [18], iterative radial voting (IRV) [23], and image-based tool for counting nuclei (ITCN) [52], and single-pass voting (SPV) [28]. The qualitative comparison of a randomly selected sample patch is shown in Fig. 15.9, which demonstrates the superior performance of our method.

To evaluate our algorithm quantitatively, we adopt a set of metrics defined in [29], including false negative rate (FN), false positive rate (FP), over-detection rate (OR), and effective rate (ER). Furthermore, precision (P), recall (R), and F_1 score are also computed. In our experiment, a true positive is defined as a detected seed that is within the circular neighborhood with 8-pixel distance to a ground truth and there is no other seeds within the 12-pixel distance neighborhood. The comparison results are shown in Tables 15.1, 15.2 15.3 and 15.4. It can be observed that the proposed method outperforms the other methods in terms of most of the metrics on the three data sets, including brain tumor, lung cancer, and breast cancer data sets. For the NET images, our detection method is outperformed slightly by SPV. We also observed that in solving Eq. (15.1), increasing the number of nearest neighbors can help the detection performance. This effect vanishes when more than 100 nearest neighbors are selected. Friedman test is performed on the F_1 scores obtained by the methods under comparison, and P -values <0.05 are observed. Therefore, the proposed approach is significantly better than the comparative methods.

Table 15.1 The comparison of the detection performance

Methods	Brain tumor data						
	FN	FP	OR	ER	P	R	F_1
LoG [18]	0.15	0.004	0.3	0.8	0.94	0.84	0.89
IRV [23]	0.15	0.04	0.07	0.76	0.95	0.83	0.88
ITCN [52]	0.22	0.0005	0.01	0.77	0.99	0.77	0.87
SPV [28]	0.1	0.02	0.06	0.86	0.98	0.89	0.93
Ours	0.07	0.0007	0.04	0.92	0.99	0.93	0.96

Table 15.2 The comparison of the detection performance

Methods	Lung cancer data						
	FN	FP	OR	ER	P	R	F_1
LoG [18]	0.19	0.003	0.13	0.78	0.96	0.80	0.88
IRV [23]	0.33	0.014	0.21	0.64	0.98	0.66	0.79
ITCN [52]	0.31	0.002	0.05	0.68	0.98	0.69	0.81
SPV [28]	0.18	0.008	0.006	0.79	0.98	0.81	0.89
Ours	0.15	0.01	0.06	0.81	0.96	0.85	0.90

Table 15.3 The comparison of the detection performance

Methods	Breast cancer data						
	FN	FP	OR	ER	P	R	F_1
LoG [18]	0.08	0.002	0.29	0.90	0.98	0.92	0.95
IRV [23]	0.21	0.11	0.11	0.74	0.95	0.75	0.84
ITCN [52]	0.19	0.006	0.18	0.78	0.98	0.80	0.88
SPV [28]	0.18	0.001	0.066	0.82	0.997	0.82	0.90
Ours	0.11	0.001	0.05	0.865	0.978	0.88	0.93

Table 15.4 The comparison of the detection performance

Methods	NET data						
	FN	FP	OR	ER	P	R	F_1
LoG [18]	0.12	0.002	0.12	0.85	0.97	0.87	0.92
IRV [23]	0.17	0.83	0.06	0.72	0.96	0.81	0.88
ITCN [52]	0.15	0.07	0.003	0.73	0.95	0.83	0.88
SPV [28]	0.04	0.012	0.05	0.90	0.95	0.95	0.95
Ours	0.04	0.024	0.06	0.85	0.90	0.95	0.93

Segmentation Performance Analysis: A qualitative comparison of performance between our approaches and the other four methods, including mean-shift (MS), isoperimetric graph partitioning (ISO) [53], graph-cut, and coloring (GCC) [18], and repulsive level set (RLS) [28], is shown in Fig. 15.9. It is clear that the proposed method learns to capture the structure of the cell boundaries. Therefore, the true boundaries can be recovered in the presence of inhomogeneous intensity, and a better segmentation performance is achieved. The detection and segmentation of thousands of cells are shown in Fig. 15.10. The quantitative comparison based on the mean and variance of precision (P), recall (R), and F_1 score is shown in Tables 15.5, 15.6, 15.7 and 15.8. In addition, Friedman test followed by Bonferroni–Dunn test is conducted on the F_1 scores. P -values are all significantly smaller than 0.05. The Bonferroni–Dunn test shows that there does exist significant difference between our methods and the other state of the arts.

We also explored the segmentation performance with respect to the number of training epoches. The result is shown in Fig. 15.11a. As one can tell, the performance increases as the number of training epoches increases, and it converges after 200 epoches. The number of training samples needed for a reasonable performance depends on the variation of the data. In our experiment setting, it is observed that around 5000 samples are sufficient. The performance with respect to the model complexity is shown in Fig. 15.11b, where the dimension of the second layer is fixed to 200.

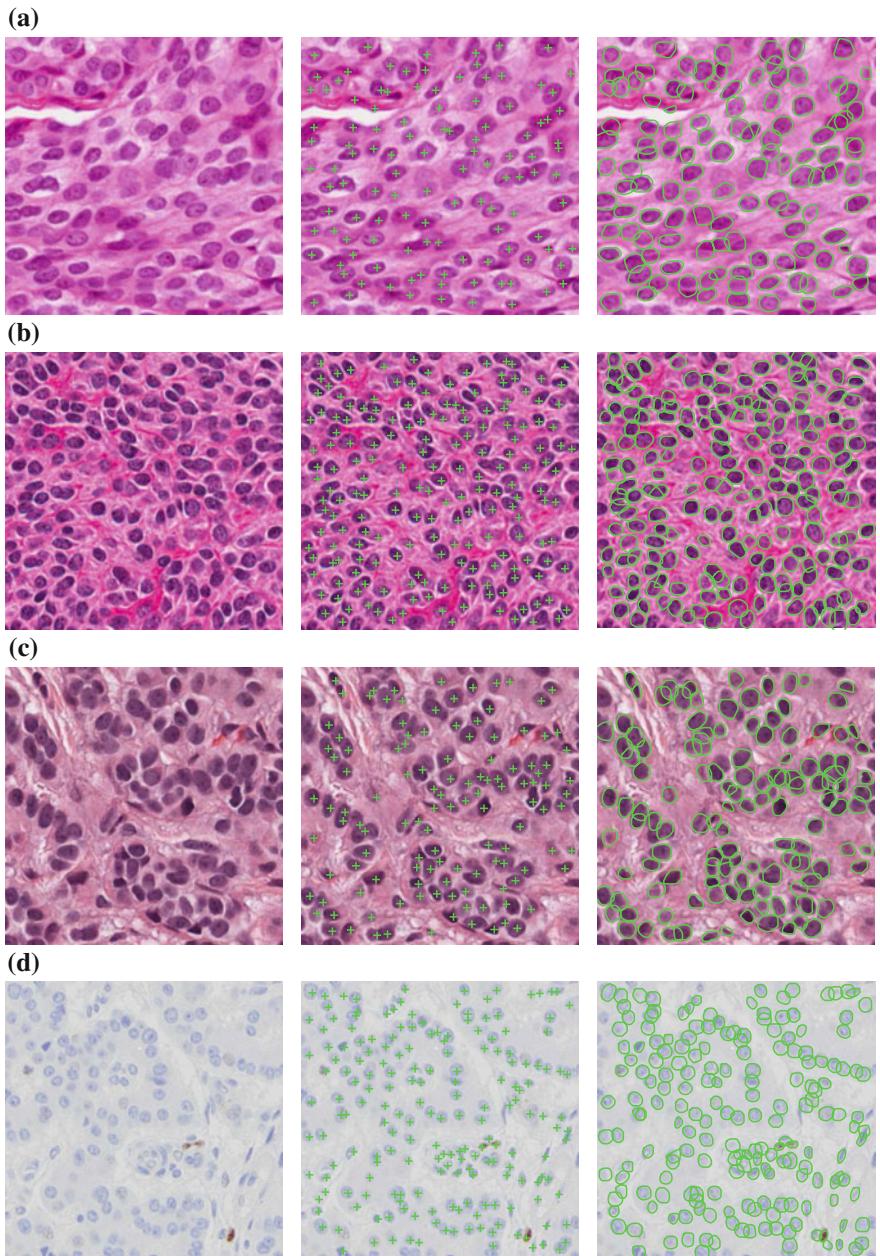


Fig. 15.10 The detection and segmentation results of one image from each of the four data sets

Table 15.5 The comparison of the segmentation performance

Methods	Brain tumor data					
	P.M.	P.V.	R.M.	R.V.	F_1 M.	F_1 . V.
MS	0.92	0.02	0.59	0.08	0.66	0.05
ISO [53]	0.71	0.04	0.81	0.03	0.71	0.03
GCC [18]	0.87	0.03	0.77	0.044	0.78	0.024
RLS [28]	0.84	0.01	0.75	0.09	0.74	0.05
sDAE	0.86	0.018	0.87	0.01	0.85	0.009
sdDAE	0.867	0.019	0.885	0.01	0.86	0.01

Table 15.6 The comparison of the segmentation performance

Methods	Lung cancer data					
	P.M.	P.V.	R.M.	R.V.	F_1 M.	F_1 . V.
MS	0.88	0.01	0.73	0.04	0.77	0.02
ISO [53]	0.75	0.03	0.82	0.025	0.75	0.02
GCC [18]	0.87	0.03	0.73	0.04	0.77	0.02
RLS [28]	0.85	0.013	0.82	0.04	0.81	0.02
sDAE	0.86	0.023	0.85	0.012	0.84	0.01
sdDAE	0.87	0.015	0.866	0.008	0.86	0.006

Table 15.7 The comparison of the segmentation performance

Methods	Breast cancer data					
	P.M.	P.V.	R.M.	R.V.	F_1 M.	F_1 . V.
MS	0.90	0.007	0.76	0.03	0.84	0.018
ISO [53]	0.70	0.05	0.74	0.06	0.67	0.03
GCC [18]	0.87	0.02	0.72	0.04	0.76	0.024
RLS [28]	0.85	0.02	0.85	0.024	0.84	0.018
sDAE	0.85	0.019	0.91	0.01	0.87	0.01
sdDAE	0.88	0.012	0.90	0.008	0.88	0.008

Table 15.8 The comparison of the segmentation performance

Methods	NET data					
	P.M.	P.V.	R.M.	R.V.	F_1 M.	F_1 . V.
MS	0.86	0.014	0.76	0.034	0.78	0.017
ISO [53]	0.73	0.048	0.67	0.05	0.66	0.032
GCC [18]	0.88	0.033	0.57	0.046	0.64	0.031
RLS [28]	0.83	0.034	0.79	0.032	0.80	0.025
sDAE	0.89	0.015	0.84	0.012	0.857	0.008
sdDAE	0.89	0.015	0.85	0.011	0.86	0.007

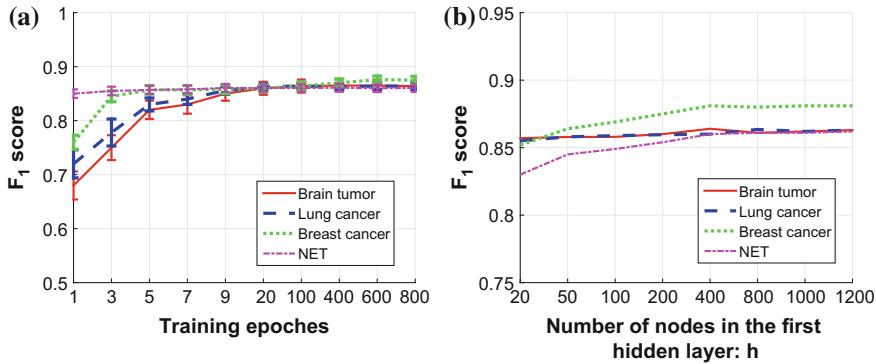


Fig. 15.11 **a** F₁ score as a function of the number of training epochs. **b** F₁ score as a function of the model complexity

15.3.1 Computational Complexity

The time complexity of the detection algorithm is composed of two parts: (1) the adaptive dictionary generation, and (2) the sliding window scanning and clustering. The step (1) needs to be done only once for the images from one patient. It is a constant depending on K , the number of patches selected by K-selection. The time complexity step (2) is dominated by the optimization of the LLC problem in Eq. (15.1). To compute one sliding window, the optimization procedure consists of a KNN search and solving an analytic solution to a least square problem. The time complexity is $\mathcal{O}(k + q + s^2)$, where k denotes the number of nearest neighbors ($k = 100$), q represents the size of the dictionary generated in step (1), and s^2 is the dimension of the trivial templates that equals to the number of pixels in a sliding window. For an image of size $v \times v$, the time complexity is $\mathcal{O}(v^2(k + q + s^2))$. The time complexity for weight-guided mean-shift clustering is $\mathcal{O}(TR^2)$, where T denotes the maximal number of iterations, and R is the number of data points. The proposed detection algorithm is based on MATLAB and is not yet optimized with respect to efficiency. It takes about 55 s to obtain the detection result of an image of size 300×300 .

The complexity of the segmentation is dependent on the number of cells detected. For each cell, the time complexity is a constant $\mathcal{O}(s^2(L_{dae}^{(1)} + L_{dae}^{(2)} + T_{def}))$, where $L_{dae}^{(1)}$ and $L_{dae}^{(2)}$ denote the number of hidden units in the first layer and the second layer of the denoising autoencoder, respectively, and T_{def} represents the maximum number of iterations for the active contour deformable model, and s^2 is the size of the input patch. In our experiment, it takes only 286 s to segment 2000 cells with $\{s^2 = 784, L_{dae}^{(1)} = 1000, L_{dae}^{(2)} = 1200, T_{def} = 5\}$.

15.4 Conclusion

In this paper, we have proposed an automatic cell detection and segmentation algorithm for pathological images. The detection step exploits sparse reconstruction with trivial templates to handle shape variations and touching cells. The segmentation step applies an sdDAE trained with structural labels and discriminative losses to remove the non-boundary edges. The proposed algorithm is tested on four data sets containing more than 6000 cells, and it exhibits superior performance over other methods. The proposed approach is a general approach that can be adapted to many other pathological applications.

References

1. Veta M, Pluim JP, van Diest PJ, Viergever MA (2014) Breast cancer histopathology image analysis: a review. *TBME* 61(5):1400–1411
2. Zhang X, Liu W, Dundar M, Badve S, Zhang S (2015) Towards large-scale histopathological image analysis: hashing-based image retrieval. *IEEE Trans Med Imaging* 34(2):496–506
3. Zhang X, Xing F, Su H, Yang L, Zhang S (2015) High-throughput histopathological image analysis via robust cell segmentation and hashing. *J Med Image Anal* 26(1):306–315
4. Xing F, Yang L (2016) Robust nucleus/cell detection and segmentation in digital pathology and microscopy images: a comprehensive review. *IEEE Rev Biomed Eng* (99):1
5. Malpica N, Ortiz de Solorzano C, Vaquero JJ, Santos A, Vallcorba I, Garcia-Sagredo JM, Pozo Fd (1997) Applying watershed algorithms to the segmentation of clustered nuclei
6. Ancin H, Roysam B, Dufresne T, Chestnut M, Ridder G, Szarowski D, Turner J (1996) Advances in automated 3-d image analysis of cell populations imaged by confocal microscopy. *J Cytom* 25(3):22–234
7. Grau V, Mewes AUJ, Alcaniz M, Kikinis R, Warfield S (2004) Improved watershed transform for medical image segmentation using prior information. *IEEE Trans Med Imaging* 23(4):447–458
8. Schmitt O, Hasse M (2008) Radial symmetries based decomposition of cell clusters in binary and gray level images. *J Pattern Recognit* 41(6):1905–1923
9. Yang X, Li H, Zhou X (2006) Nuclei segmentation using marker-controlled watershed, tracking using mean-shift, and kalman filter in time-lapse microscopy. *IEEE Trans Circuits Syst I Regul Pap* 53(11):2405–2414
10. Jung C, Kim C (2010) Segmenting clustered nuclei using h-minima transform-based marker extraction and contour parameterization. *IEEE Trans Biomed Eng* 57(10):2600–2604
11. Mao K, Zhao P, Tan P (2006) Supervised learning-based cell image segmentation for p53 immunohistochemistry. *IEEE Trans Biomed Eng* 53(6):1153–1163
12. Lin G, Chawla MK, Olson K, Barnes CA, Guzowski JF, Bjornsson C, Shain W, Roysam B (2007) A multi-model approach to simultaneous segmentation and classification of heterogeneous populations of cell nuclei in 3d confocal microscope images. *Cytom Part A* 71A(9):724–736
13. Cinar Akakin H, Kong H, Elkins C, Hemminger J, Miller B, Ming J, Plocharczyk E, Roth R, Weinberg M, Ziegler R, Lozanski G, Gurcan M (2012) Automated detection of cells from immunohistochemically-stained tissues: application to ki-67 nuclei staining. In: SPIE, vol 8315
14. Yan P, Zhou X, Shah M, Wong S (2008) Automatic segmentation of high-throughput rna fluorescent cellular images. *IEEE Trans Inf Technol Biomed* 12(1):109–117
15. Faustino GM, Gattass M, Rehen S, de Lucena C (2009) Automatic embryonic stem cells detection and counting method in fluorescence microscopy images. In: Proceedings of IEEE international symposium on biomedical imaging: from nano to macro (ISBI), pp 799–802

16. Lou X, Koethe U, Wittbrodt J, Hamprecht F (2012) Learning to segment dense cell nuclei with shape prior. In: IEEE conference on computer vision and pattern recognition (CVPR), pp pp 1012–1018
17. Bernardis E, Yu S (2010) Finding dots: segmentation as popping out regions from boundaries. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), pp pp 199–206
18. Al-Kofahi Y, Lassoued W, Lee W, Roysam B (2010) Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Trans Biomed Eng* 57(4):841–852
19. Chang H, Han J, Borowsky A, Loss L, Gray JW, Spellman PT, Parvin B (2013) Invariant delineation of nuclear architecture in glioblastoma multiforme for clinical and molecular association. *IEEE Trans Med Imaging* 32(4):670–682
20. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Proceedings of advances in neural information processing systems (NIPS), pp 1097–1105
21. Cireşan DC, Giusti A, Gambardella LM, Schmidhuber J (2013) Mitosis detection in breast cancer histology images with deep neural networks. In: Medical image computing and computer-assisted intervention (MICCAI), pp 411–418
22. Hafiane A, Bunyak F, Palaniappan K (2008) Fuzzy clustering and active contours for histopathology image segmentation and nuclei detection. In: Proceedings of advanced concepts for intelligent vision systems, pp 903–914
23. Parvin B, Yang Q, Han J, Chang H, Rydberg B, Barcellos-Hoff MH (2007) Iterative voting for inference of structural saliency and characterization of subcellular events. *IEEE Trans Image Process* 16(3):615–623
24. Kong H, Gurcan M, Belkacem-Boussaid K (2011) Partitioning histopathological images: an integrated framework for supervised color-texture segmentation and cell splitting. *IEEE Trans Med Imaging* 30(9):1661–1677
25. Veta M, van Diest PJ, Kornegoor R, Huisman A, Viergever MA, Pluim JP (2013) Automatic nuclei segmentation in h&e stained breast cancer histopathology images. *PLoS one* 8(7):e70221
26. Kothari S, Chaudry Q, Wang M (2009) Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques. In: Proceedings of IEEE international symposium on biomedical imaging (ISBI), pp 795–798, 28 June 2009–1 July 2009
27. Su H, Xing F, Lee J, Peterson C, Yang L (2013) Automatic myonuclear detection in isolated single muscle fibers using robust ellipse fitting and sparse optimization. *IEEE/ACM Trans Comput Biol Bioinform* (99):1–1 (2013)
28. Qi X, Xing F, Foran D, Yang L (2012) Robust segmentation of overlapping cells in histopathology specimens using parallel seed detection and repulsive level set. *IEEE Trans Biomed Eng* 59(3):754–765
29. Xing F, Su H, Neltner J, Yang L (2014) Automatic ki-67 counting using robust cell detection and online dictionary learning. *IEEE Trans Biomed Eng* 61(3):859–870
30. Ali S, Madabhushi A (2012) An integrated region-, boundary-, shape-based active contour for multiple object overlap resolution in histological imagery. *IEEE Trans Med Imaging* 31(7):1448–1460
31. Xing F, Yang L (2013) Robust selection-based sparse shape model for lung cancer image segmentation. In: Medical image computing and computer-assisted intervention (MICCAI), pp 404–412
32. Huang Y, Wu Z, Wang L, Tan T (2014) Feature coding in image classification: a comprehensive study. *IEEE Trans Pattern Anal Mach Intell* 36(3):493–506
33. Wang J, Yang J, Yu K, Lv F, Huang T, Gong Y (2010) Locality-constrained linear coding for image classification. In: Proceedings of IEEE conference on computer vision and pattern recognition, pp 3360–3367
34. Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y (2009) Robust face recognition via sparse representation. *IEEE Trans Pattern Anal Mach Intell* 31(2):210–227
35. Yu K, Zhang T, Gong Y (2009) Nonlinear learning using local coordinate coding. In: Proceedings of neural information processing systems, vol 9, p 1

36. Liao S, Gao Y, Shen D (2012) Sparse patch based prostate segmentation in ct images. In: Medical image computing and computer-assisted intervention (MICCAI), Springer, pp 385–392
37. Zhang S, Li X, Lv J, Jiang X, Zhu D, Chen H, Zhang T, Guo L, Liu T (2013) Sparse representation of higher-order functional interaction patterns in task-based fmri data. In: Medical image computing and computer-assisted intervention (MICCAI), Springer, pp 626–634
38. Kårsnäs A, Dahl AL, Larsen R (2011) Learning histopathological patterns. *J Pathol Inf* 2:12
39. Xing F, Xie Y, Yang L (2015) An automatic learning-based framework for robust nucleus segmentation. *IEEE Trans Med Imaging* (99):1–1
40. Chang H, Nayak N, Spellman PT, Parvin B (2013) Characterization of tissue histopathology via predictive sparse decomposition and spatial pyramid matching. In: Medical image computing and computer-assisted intervention (MICCAI), pp 91–98
41. Chang H, Zhou Y, Spellman P, Parvin B (2013) Stacked predictive sparse coding for classification of distinct regions in tumor histopathology. In: IEEE international conference on computer vision (ICCV), pp 169–176
42. Zhou Y, Chang H, Barner K, Spellman P, Parvin B (2014) Classification of histology sections via multispectral convolutional sparse coding. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), pp 3081–3088
43. Alain G, Bengio Y (2014) What regularized auto-encoders learn from the data-generating distribution. *J Mach Learn Res* 15(1):3563–3593
44. Chen M, Weinberger KQ, Sha F, Bengio Y (2014) Marginalized denoising auto-encoders for nonlinear representations. In: Proceedings of the 31st international conference on machine learning (ICML), pp 1476–1484
45. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010) Stacked denoising auto-encoders: learning useful representations in a deep network with a local denoising criterion. *J Mach Learn Res* 11:3371–3408
46. Su H, Xing F, Kong X, Xie Y, Zhang S, Yang L (2015) Robust cell detection and segmentation in histopathological images using sparse reconstruction and stacked denoising autoencoders. *Med Image Comput Comput-Assist Interv (MICCAI)* 9351:383–390
47. Liu B, Huang J, Yang L, Kulikowsk C (2011) Robust tracking using local sparse appearance model and k-selection. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), pp 1313–1320
48. Comaniciu D, Ramesh V, Meer P (2003) Kernel-based object tracking. *IEEE Trans Pattern Anal Mach Intell* 25(5):564–577
49. Seo HJ, Milanfar P (2010) Training-free, generic object detection using locally adaptive regression kernels. *IEEE Trans Pattern Anal Mach Intell* 32(9):1688–1704
50. Chan TF, Vese LA (2001) Active contours without edges. *IEEE Trans Image Process* 10(2):266–277
51. Rolfe JT, LeCun Y (2013) Discriminative recurrent sparse auto-encoders. [arXiv:1301.3775](https://arxiv.org/abs/1301.3775)
52. Byun J, Verardo MR, Sumengen B, Lewis GP, Manjunath B, Fisher SK (2006) Automated tool for the detection of cell nuclei in digital microscopic images: application to retinal images. *Mol Vis* 12:949–960
53. Grady L, Schwartz EL (2006) Isoperimetric graph partitioning for image segmentation. *IEEE Trans Pattern Anal Mach Intell* 28(3):469–475

Chapter 16

Automatic Pancreas Segmentation Using Coarse-to-Fine Superpixel Labeling

**Amal Farag, Le Lu, Holger R. Roth, Jiamin Liu, Evrim Turkbey
and Ronald M. Summers**

Abstract Accurate automatic detection and segmentation of abdominal organs from CT images is important for quantitative and qualitative organ tissue analysis, detection of pathologies, surgical assistance as well as computer-aided diagnosis (CAD). In general, the large variability of organ locations, the spatial interaction between organs that appear similar in medical scans and orientation and size variations are among the major challenges of organ segmentation. The pancreas poses these challenges in addition to its flexibility which allows for the shape of the tissue to vastly change. In this chapter, we present a fully automated bottom-up approach for pancreas segmentation in abdominal computed tomography (CT) scans. The method is a four-stage system based on a hierarchical cascade of information propagation by classifying image patches at different resolutions and cascading (segments) superpixels. System components consist of the following: (1) decomposing CT slice images as a set of disjoint boundary-preserving superpixels; (2) computing pancreas class probability maps via dense patch labeling; (3) classifying superpixels by pooling both intensity and probability features to form empirical statistics in cascaded random forest frameworks; and (4) simple connectivity based post-processing. Evaluation of the approach is conducted on a database of 80 manually segmented CT volumes in sixfold cross validation. Our achieved results are comparable, or better to the state-of-the-art methods (evaluated by “leave-one-patient-out”), with a Dice coefficient of 70.7% and Jaccard Index of 57.9%. The computational efficiency of the proposed approach is drastically improved in the order of 6–8 min, compared to other methods of ≥ 10 hours per testing case.

A. Farag · L. Lu (✉) · H.R. Roth · J. Liu · E. Turkbey · R.M. Summers
Department of Radiology and Imaging Sciences, National Institutes of Health
Clinical Center, Bethesda, MD 20837, USA
e-mail: le.lu@nih.gov

R.M. Summers
e-mail: rms@nih.gov

16.1 Introduction

Image segmentation is a key step in image understanding that aims at separating objects within an image into classes, based on object characteristics and a prior information about the surroundings. This also applies to medical image analysis in various imaging modalities. The segmentation of abdominal organs such as the spleen, liver, and pancreas in abdominal computed tomography (CT) scans can be an important input to computer-aided diagnosis (CAD) systems, for quantitative and qualitative analysis and for surgical assistance. In the instance of quantitative imaging analysis of diabetic patients, a requisite critical step for the development of such CAD systems is segmentation specifically of the pancreas. Pancreas segmentation is also a necessary input for subsequent methodologies for pancreatic cancer detection. The literature is rich in methods of automatic segmentation on CT with high accuracies (e.g., Dice coefficients >90%), of other organs such as the kidneys [1], lungs [2], heart [3], and liver [4]. Yet, high accuracy in automatic segmentation of the pancreas remains a challenge. The literature is not as abundant in either single- or multi-organ segmentation setups.

The pancreas is a highly anatomically variable organ in terms of shape and size and the location within the abdominal cavity shifts from patient to patient. The boundary contrast can vary greatly by the amount of visceral fat in the proximity of the pancreas. These factors and others make segmentation of the pancreas very challenging. Figure 16.1 depicts several manually segmented 3D volumes of various patient pancreases to better illustrate the variations and challenges mentioned. From the above observations, we argue that the automated pancreas segmentation problem

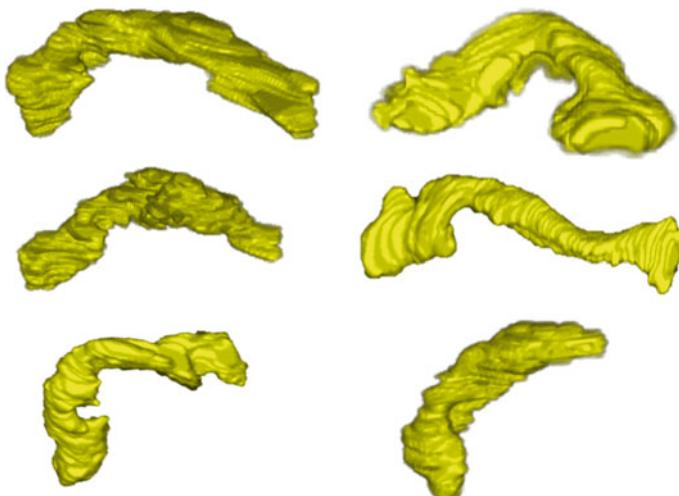


Fig. 16.1 3D manually segmented volumes of six pancreases from six patients. Notice the shape and size variations

should be treated differently, apart from the current organ segmentation literature where statistical shape models are generally used.

In this chapter, a new fully bottom-up approach using image and (deep) patch-level labeling confidences for pancreas segmentation is proposed using 80 single-phase CT patient data volumes. The approach is motivated to improve the segmentation accuracy of *highly deformable organs*, like the pancreas, by leveraging *middle-level representation of image segments*. First, over segmentation of all 2D slices of an input patient abdominal CT scan is obtained as a semi-structured representation known as superpixels. Second, classifying superpixels into two semantic classes of pancreas and non-pancreas is conducted as a multistage feature extraction and random forest (RF) classification process, on the image and (deep) patch-level confidence maps, pooled at the superpixel level. Two cascaded random forest superpixel classification frameworks are presented and compared. Figure 16.2 depicts the overall proposed first framework. Figure 16.9 illustrates the modularized flow charts of both frameworks. Our experimental results are carried out in a sixfold cross-validation manner.

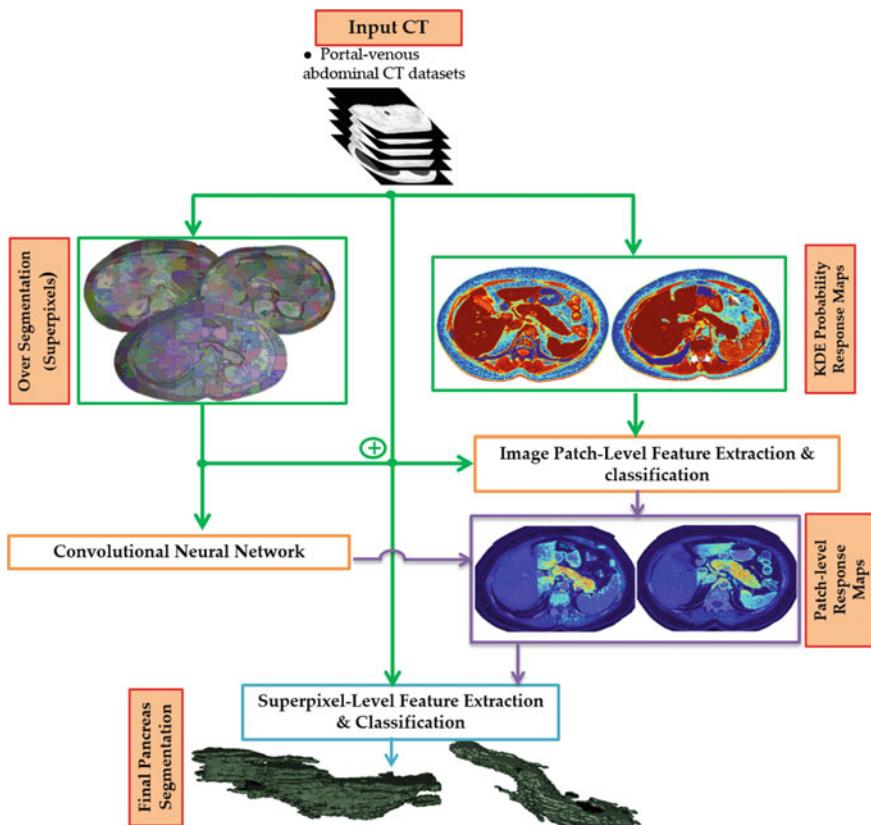


Fig. 16.2 Overall pancreas segmentation framework via dense image patch labeling

Our system runs at about two orders of magnitude more computationally efficiently to process a new testing case than the atlas registration based approaches [5–10]. The obtained results are comparable, or better than the state-of-the-art methods (evaluated by “leave-one-patient-out”), with a Dice coefficient of 70.7% and Jaccard Index of 57.9%. Under the same sixfold cross validation, our bottom-up segmentation method significantly outperforms its “multi-atlas registration and joint label fusion” (MALF) counterpart (based on our implementation using [11, 12]): Dice coefficients $70.7 \pm 13.0\%$ versus $52.51 \pm 20.84\%$. Additionally, another bottom-up supervoxel based multi-organ segmentation without registration in 3D abdominal CT images is also investigated [13] in a similar spirit, for demonstrating this methodological synergy.

16.2 Previous Literature

The organ segmentation literature can be divided into two broad categories: top-down and bottom-up approaches. In top-down approaches, a priori knowledge such as atlas(es) and/or shape models of the organ are generated and incorporated into the framework via learning based shape model fitting [3, 4] or volumetric image registration [7, 8, 10]. For bottom-up approaches segmentation is performed by local image similarity grouping and growing or pixel, superpixel/supervoxel-based labeling [14, 15] since direct representations of the organ is not incorporated. Generally speaking, top-down methods are targeted for organs which can be modeled well by statistical shape models [3] whereas bottom-up representations are more effective for highly non-Gaussian shaped [14, 15] or pathological organs.

Previous literature on pancreas segmentation from CT images have been dominated by top-down approaches which rely on atlas-based approaches or statistical shape modeling or both [5–10].

- Shimizu et al. [5] utilize three-phase contrast enhanced CT data which are first registered together for a particular patient and then registered to a reference patient by landmark-based deformable registration. The spatial support area of the abdominal cavity is reduced by segmenting the liver, spleen, and three main vessels associated with location interpretation of the pancreas (i.e., splenic, portal, and superior mesenteric veins). Coarse-to-fine pancreas segmentation is performed by using generated patient-specific probabilistic atlas guided segmentation followed by intensity-based classification and post-processing. Validation of the approach was conducted on 20 multi-phase datasets resulting in a Jaccard of 57.9%.
- Okada et al. [6] perform multi-organ segmentation by combining inter-organ spatial interrelations with probabilistic atlases. The approach incorporated various a priori knowledge into the model that includes shape representations of seven organs. Experimental validation was conducted on 28 abdominal contrast-enhanced CT datasets obtaining an overall volume overlap of Dice index 46.6% for the pancreas.

- Chu et al. [8] present an automated multi-organ segmentation method based on spatially divided probabilistic atlases. The algorithm consists of image-space division and a multi-scale weighting scheme to deal with the large differences among patients in organ shape and position in local areas. Their experimental results show that the liver, spleen, pancreas, and kidneys can be segmented with Dice similarity indices of 95.1, 91.4, 69.1, and 90.1%, respectively, using 100 annotated abdominal CT volumes.
- Wolz et al. [7] may be considered the state-of-the-art result thus far for single-phase pancreas segmentation. The approach is a multi-organ segmentation approach that combines hierarchical weighted subject-specific atlas-based registration and patch-based segmentation. Post-processing is in the form of optimized graph-cuts with a learned intensity model. Their results in terms of a Dice overlap for the pancreas is 69.6% on 150 patients and 58.2% on a subpopulation of 50 patients.
- Recent work by Wang et al. [10] proposes a patch-based label propagation approach that uses relative geodesic distances. The approach can be considered a start to developing some bottom-up component for segmentation, where affine registration between dataset and atlases were conducted followed by refinement using the patch-based segmentation to reduce misregistrations and instances of high anatomy variability. The approach was evaluated on 100 abdominal CT scans with an overall Dice of 65.5% for the pancreas segmentation.

The default experimental setting in many of the atlas-based approaches [5–10] is conducted in a “leave-one-patient-out” or “leave-one-out” (LOO) criterion for up to $N = 150$ patients. In the clinical setting, leave-one-out based dense volume registration (from all other $N-1$ patients as atlas templates) and label fusion process may be computationally impractical (10+ hours per testing case). More importantly, it does not scale up easily when large-scale datasets are present. On the other hand, efficient cascade classifiers have been studied in both computer vision and medical image analysis problems [16–18], with promising results.

16.3 Methods

In this section, the components of our overall algorithm flow (shown in Fig. 16.2) are first addressed (Sects. 16.3.1 and 16.3.2). The method extensions on exploiting sliding-window CNN-based dense image patch labeling and framework variations are described in Sects. 16.3.3 and 16.3.4.

16.3.1 Boundary-Preserving Over-segmentation

Over-segmentation occurs when images (or more generally grid graphs) are segmented or decomposed into smaller perceptually meaningful regions, “superpixels”.

Within a superpixel, pixels carry similarities in color, texture, intensity, etc., and generally align with image edges rather than rectangular patches (i.e., superpixels can be irregular in shape and size). In the computer vision literature, numerous approaches have been proposed for superpixel segmentation [19–23]. Each approach has its drawbacks and advantages but three main properties are generally examined when deciding the appropriate method for an application as discussed in [20]: (1) adherence to image boundaries; (2) computationally fast, ease of usage and memory efficient; especially when computational complexity reduction is of importance and (3) improvement on both quality and speed of the final segmentation.

Superpixel methods fall under two main broad categories: graph-based (e.g., SLIC [19], entropy rate [21] and [22]) and gradient ascent methods (e.g., watershed [23] and mean shift [24]). In terms of computational complexity, [22, 23] are relatively fast in $O(M \log M)$ complexity where M is the number of pixels or voxels in the image or grid graph. Mean shift [24] and normalized cut [25] are $O(M^2)$, or $O(M^{\frac{3}{2}})$, respectively. Simple linear iterative clustering (SLIC) [19] is both fast and memory efficient. In our work, evaluation and comparison among three graph-based superpixel algorithms (i.e., SLIC [19, 20], efficient graph-based [22] and Entropy rate [21]) and one gradient ascent method (i.e., watershed [23]) are conducted, considering the three criterion in [20]. Figure 16.3 shows sample superpixel results using the SLIC approach. The original CT slices and cropped zoomed-in pancreas superpixel regions are demonstrated. The boundary recall, a typical measurement used in the literature, to indicate how many “true” edge pixels of the ground-truth object segmentation are within a pixel range from the superpixels (i.e., object-level edges are recalled by superpixel boundaries). High boundary recall indicates minimal true edges were neglected. Figure 16.4 shows sample quantitative results. Based on Fig. 16.3, high boundary recalls, within the distance ranges between 1 and 6 pixels from the semantic pancreas ground-truth boundary annotation are obtained using the SLIC approach. The watershed approach provided the least promising results for usage in the pancreas, due to the lack of conditions in the approach, to utilize boundary information in conjunction with intensity information as implemented in graph-based approaches. The superpixel number range per axial image is constrained $\in [100, 200]$ to make a good trade-off on superpixel dimensions or sizes.

The overlapping ratio r of the superpixel versus the ground-truth pancreas annotation mask is defined as the percentage of pixels/voxels inside each superpixel that are annotated as pancreas. By thresholding on r , say if $r > \tau$ the superpixel will be labeled as pancreas and otherwise as background, we can obtain the pancreas segmentation results. When $\tau = 0.50$, the achieved mean Dice coefficient is $81.2 \pm 3.3\%$ which is referred as the “Oracle” segmentation accuracy since computing r would require to know the ground-truth segmentation. This is also the upper bound segmentation accuracy for our superpixel labeling or classification framework. $81.2 \pm 3.3\%$ is significantly higher and numerically more stable (in standard deviation) than previous state-of-the-art methods [5, 7–10], to provide considerable improvement space of our work. Note that both the choices of SLIC and $\tau = 0.50$ are calibrated using a subset of 20 scans. We find there is no need to evaluate different superpixel generation methods/parameters and τ s as “model selection” using the

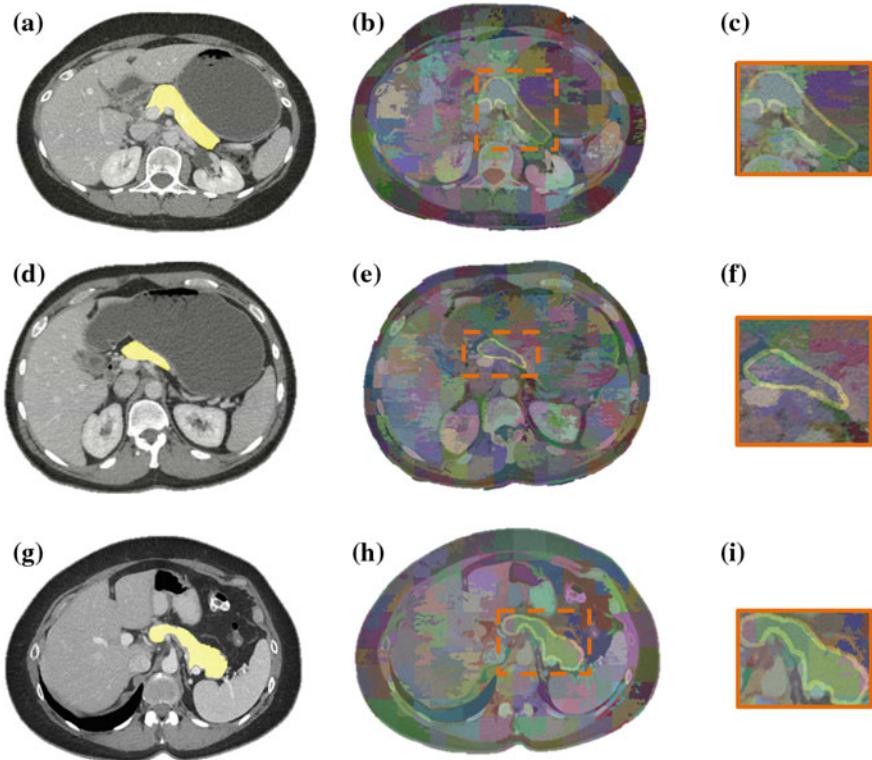
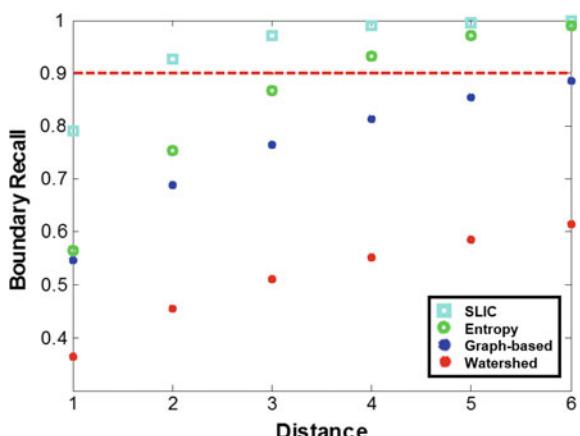


Fig. 16.3 Sample superpixel generation results from the SLIC method [19]. First column depicts different slices from different patient scans with the ground-truth pancreas segmentation in yellow (a, d and g). The second column depicts the over segmentation results with the pancreas contours superimposed on the image (b, e and h). Last, (c) (f) and (i) show zoomed-in areas of the pancreas superpixel results from b, e and h

Fig. 16.4 Superpixels boundary recall results evaluated on 20 patient scans (Distance in millimeters). The watershed method [23] is shown in red, efficient graph [22] in blue while the SLIC [19] and the Entropy rate [21] based methods are depicted in cyan and green, respectively. The red line represents the 90% marker



training folds in each round of sixfold cross validation. This superpixel calibration procedure is generalized well to all our datasets. Voxel-level pancreas segmentation can be propagated from superpixel-level classification and further improved by efficient narrow-band level-set based curve evolution [26], or the learned intensity model based graph-cut [7].

16.3.2 Patch-Level Visual Feature Extraction and Classification: P^{RF}

Feature extraction is a form of object representation that aims at capturing the important shape, texture, and other salient features that allow distinctions between the desired object (i.e., pancreas) and the surrounding to be made. In this work a total of 46 patch-level image features to depict the pancreas and its surroundings are implemented. The overall 3D abdominal body region per patient is first segmented and identified using a standard table-removal procedure where all voxels outside the body are removed.

(1) To describe the texture information, we adopt the Dense Scale-Invariant Feature transform (dSIFT) approach [27] which is derived from the SIFT descriptor [28] with several technical extensions. The publicly available VLFeat implementation of the dSIFT is employed [27]. Figure 16.5 depicts the process implemented on a sample image slice. The descriptors are densely and uniformly extracted from

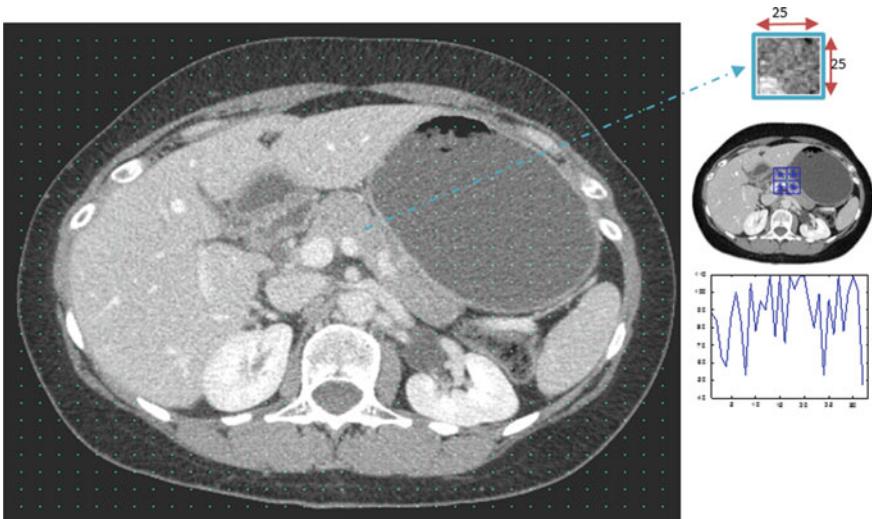


Fig. 16.5 Sample slice with center positions superimposed as *green dots*. The 25×25 image patch and corresponding D-SIFT descriptors are shown to the *right* of the original image

image grids with inter-distances of 3 pixels. The patch center position are shown as the green points superimposed on the original image slice. Once the positions are known, the dSIFT is computed with the geometry of $[2 \times 2]$ bins and bin size of 6 pixels, which results in a 32 dimensional texture descriptor for each image patch. The image patch size in this work is fixed at 25×25 which is a trade-off between computational efficiency and description power. Empirical evaluation of the image patch size is conducted for the size range of 15–35 pixels using a small subsampled dataset for classification, as described later. Stable performance statistics are observed and quantitative experimental results using the default patch size of 25×25 pixels are reported.

(2) A second feature group using the voxel intensity histograms of the ground-truth pancreas and the surrounding CT scans is built in the class-conditional probability density function (PDF) space. A kernel density estimator (KDE¹) is created using the voxel intensities from a subset of randomly selected patient CT scans. The KDE represents the CT intensity distributions of the positive $\{X^+\}$ and negative class $\{X^-\}$ of pancreas and non-pancreas voxels CT image information. All voxels containing pancreas information are considered in the positive sample set, yet, since negative voxels far outnumber the positive only 5% of the total number from each CT scan (by random resampling) is considered. Let, $\{X^+\} = (h_1^+, h_2^+, \dots, h_n^+)$ and $\{X^-\} = (h_1^-, h_2^-, \dots, h_m^-)$ where h_n^+ and h_m^- represent the intensity values for the positive and negative pixel samples for all 26 patient CT scans over the entire abdominal CT Hounsfield range. The kernel density estimators $f^+(X^+) = \frac{1}{n} \sum_{i=1}^n K(X^+ - X_i^+)$ and $f^-(X^-) = \frac{1}{m} \sum_{j=1}^m K(X^- - X_j^-)$ are computed where $K()$ is assumed to be a Gaussian kernel with optimal computed bandwidth, for this data, of 3.039. Kernel sizes or bandwidth may be selected automatically using 1D Likelihood-based search, as provided by the used KDE toolkit. The normalized likelihood ratio is calculated which becomes a probability value as a function of intensity in the range of $H = [0 : 1 : 4095]$. Thus, the probability of being considered pancreas is formulated as $y^+ = \frac{(f^+(X^+))}{(f^+(X^+)) + f^-(X^-))}$. This function is converted as a precomputed lookup table over $H = [0 : 1 : 4095]$, which allows very efficient $O(1)$ access time.

(3) Utilizing first the KDE probability response maps above and the superpixel CT masks described in Sect. 16.3.1, as underlying supporting masks to each image patch, the same KDE response statistics within the intersected subregions, P' of P , are extracted. The idea is that an image patch, P , may be divided into more than one superpixel. This set of statistics is calculated with respect to the most representative superpixel (that covers the patch center pixel). In this manner, object boundary-preserving intensity features are obtained.

(4) The final two features for each axial slice (in the patient volumes) are the normalized relative x-axis and y-axis positions $\epsilon[0, 1]$, computed at each image patch

¹<http://www.ics.uci.edu/~ihler/code/kde.html>.

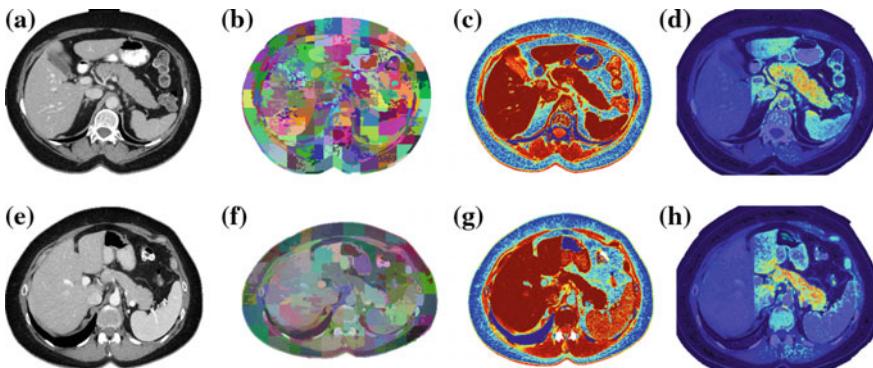


Fig. 16.6 Two sample slices from different patients are shown in **a** and **e**. The corresponding superpixels segmentation (**b**, **f**), KDE probability response maps (**c**, **g**) and RF patch-level probability response maps (**d**, **h**) are shown. In **c**, **g** and **d**, **h**, red represents highest probabilities. In **d**, **h** the purple color represents areas where probabilities are so small and can be deemed insignificant areas of interest

center against the segmented body region (self-normalized² to patients with different body masses to some extent). Once all of the features are concatenated, a total of 46 image patch-level features per superpixel are used to train a random forest (RF) classifier C_p . Image patch labels are obtained by directly borrowing the class information of their patch center pixels, based on the manual segmentation.

Sixfold cross validation for RF training is carried out. Response maps are computed for the image patch-level classification and dense labeling. Figure 16.6d, h show sample illustrative slices from different patients. High probability corresponding to the pancreas is represented by the red color regions (the background is blue). The response maps (denoted as P^{RF}) allow several observations to be made. The most interesting is that the relative x and y positions as features allow for clearer spatial separation of positive and negative regions, via internal RF feature thresholding tests on them. The trained RF classifier is able to recognize the negative class patches residing in the background, such as liver, vertebrae and muscle using spatial location cues. In Fig. 16.6d, h implicit vertical and horizontal decision boundary lines can be seen in comparison to Fig. 16.6c, g. This demonstrates the superior descriptive and discriminative power of the feature descriptor on image patches (P and P') than single pixel intensities. Organs with similar CT values are significantly depressed in the patch-level response maps.

In summary, SIFT and its variations, e.g., D-SIFT have shown to be informative, especially through spatial pooling or packing [29]. A wide range of pixel-level correlations and visual information per image patch is also captured by the rest of 14

²The axial reconstruction CT scans in our study have largely varying ranges or extends in the z-axis. If some anatomical landmarks, such as the bottom plane of liver, the center of kidneys, can be provided automatically, the anatomically normalized z-coordinate positions for superpixels can be computed and used as an additional spatial feature for RF classification.

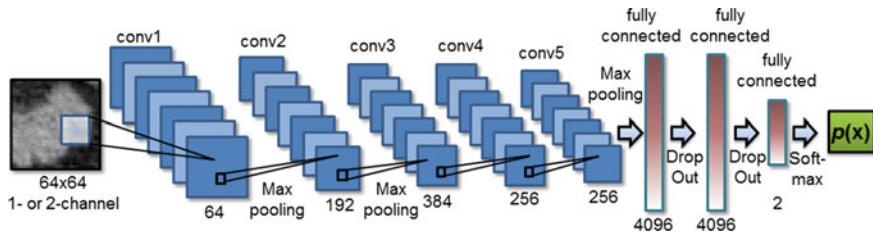


Fig. 16.7 The proposed CNN model architecture is composed of five convolutional layers with max pooling and two fully connected layers with DropOut [30] connections. A final two-way softmax layer gives a probability $p(x)$ of “pancreas” and “non-pancreas” per data sample (or image patch). The number and model parameters of convolutional filters and neural network connections for each layer are as shown

defined features. Both good classification specificity and recall have been obtained in cross validation using Random Forest implementation of 50 trees and the minimum leaf size set as 150 (i.e., using the *treebagger*(•) function in Matlab).

16.3.3 Patch-Level Labeling via Deep Convolutional Neural Network: P^{CNN}

In this work, we use Convolutional Neural Network (CNN, or ConvNet) with a standard architecture for binary image patch classification. Five layers of convolutional filters first compute, aggregate, and assemble the low level image features to more complex ones, in a layer-by-layer fashion. Other CNN layers perform max-pooling operations or consist of fully connected neural network layers. The CNN model we adopted ends with a final two-way softmax classification layer for “pancreas” and “non-pancreas” classes (refer to Fig. 16.7). The fully connected layers are constrained using “DropOut” in order to avoid over-fitting in training where each neuron or node has a probability of 0.5 to be reset with a 0-valued activation. DropOut is a method that behaves as a co-adaption regularizer when training the CNN [30]. In testing, no DropOut operation is needed. Modern GPU acceleration allows efficient training and run-time testing of the deep CNN models. We use the publicly available code base of *cuda-convnet2*.³

To extract dense image patch response maps, we use a straight-forward sliding-window approach that extracts 2.5D image patches composed of axial, coronal, and sagittal planes at any image positions (see Fig. 16.8). Deep CNN architecture can encode large-scale image patches (even the whole 224×224 pixel images [31, 32]) very efficiently and no hard crafted image features are required any more. In this work, the dimension of image patches for training CNN is 64×64 pixels which is

³<https://code.google.com/p/cuda-convnet2/>.

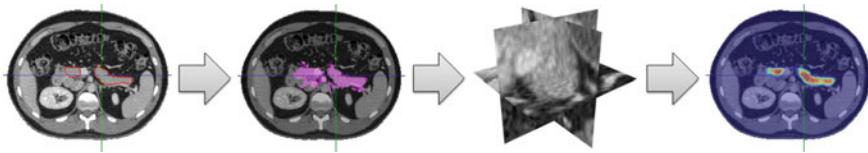


Fig. 16.8 Axial CT slice of a manual (gold standard) segmentation of the pancreas. From *Left* to *Right*, there are the ground-truth segmentation contours (in red); RF-based coarse segmentation $\{S_{RF}\}$; a 2.5D input image patch to CNN and the deep patch labeling result using CNN

significantly larger than 25×25 in Sect. 16.3.2. The larger spatial scale or context is generally expected to achieve more accurate patch labeling quality. For efficiency reasons, we extract patches every ℓ voxels for CNN feedforward evaluation and then apply nearest neighbor interpolation to estimate the values at skipped voxels. In our empirical testing, simple nearest neighbor interpolation seems sufficient due to the high quality of deep CNN probability predictions. Three examples of dense CNN based image patch labeling are demonstrated in Fig. 16.10. We denote the CNN model generated probability maps as P^{CNN} .

The computational expense of deep CNN patch labeling per patch (in a sliding-window manner) is still higher than Sect. 16.3.2. In practice, dense patch labeling by P^{RF} runs exhaustively at 3 pixel interval but P^{CNN} are only evaluated at pixel locations that pass the first stage of a cascaded random forest superpixel classification framework. This process is detailed in Sect. 16.3.4 where C_{SP}^1 is operated at a high recall (close to 100%) and low specificity mode to minimize the false negative rate (FNR) as the initial layer of cascade. The other important reason for doing so is to largely alleviate the training unbalance issue for P^{CNN} in C_{SP}^3 . After this initial pruning, the number ratio of non-pancreas versus pancreas superpixels changes from >100 to ~ 5 . The similar treatment is employed in our recent work [33] where all “Regional CNN” (R-CNN) based algorithmic variations [34] for pancreas segmentation is performed after a superpixel cascading.

16.3.4 Superpixel-Level Feature Extraction, Cascaded Classification, and Pancreas Segmentation

In this section, we trained three different superpixel-level random forest classifiers of C_{SP} 1, C_{SP} 2, and C_{SP} 3. These three classifier components further formed two cascaded RF classification frameworks (F-1, F-2), as shown in Fig. 16.9. The superpixel labels are inferred from the overlapping ratio r (defined in Sect. 16.3.1) between the superpixel label map and the ground-truth pancreas mask. If $r \geq 0.5$, the superpixel is positive while if $r \leq 0.2$, the superpixel is assigned as negative. For the rest of superpixels that fall within $0.2 < r < 0.5$ (a relatively very small portion/subset of all superpixels), they are considered ambiguous and not assigned a label and as such not used in training.

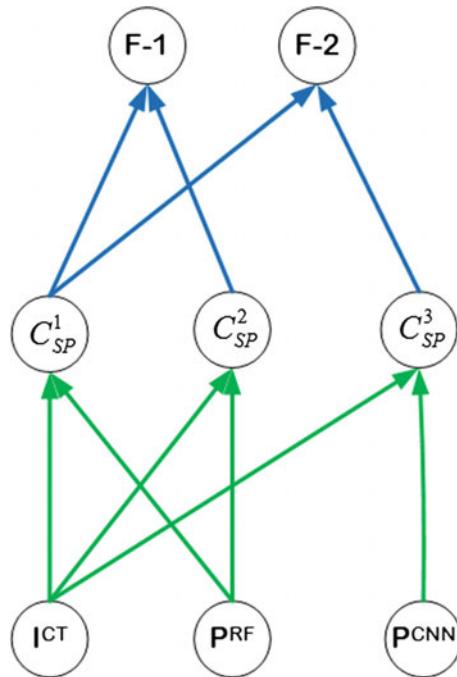


Fig. 16.9 The flow chart of input channels and component classifiers to form the overall frameworks 1 (F-1) and 2 (F-2). I^{CT} indicates the original CT image channel; P^{RF} represents the probability response map by RF-based patch labeling in Sect. 16.3.2 and P^{CNN} from deep CNN patch classification in Sect. 16.3.3, respectively. Superpixel level random forest classifier C_{SP}^1 is trained with all positive and negative superpixels in I^{CT} and P^{RF} channels; C_{SP}^2 and C_{SP}^3 are learned using only “hard negatives” and all positives, in the $I^{CT} \cup P^{RF}$ or $I^{CT} \cup P^{CNN}$ channels, respectively. Forming $C_{SP}^1 \mapsto C_{SP}^2$, or $C_{SP}^1 \mapsto C_{SP}^3$ into two overall cascaded models results in frameworks F-1 and F-2

Training C_{SP}^1 utilizes both the original CT image slices (I^{CT} in Fig. 16.9) and the probability response maps (P^{RF}) via the handcrafted feature based patch-level classification (i.e., Sect. 16.3.2). The 2D superpixel supporting maps (i.e., Sect. 16.3.1) are used for feature pooling and extraction on a superpixel level. The CT pixel intensity/attenuation numbers and the per-pixel pancreas class probability response values (from dense patch labeling of P^{RF} or P^{CNN} later) within each superpixel are treated as two empirical unordered distributions. Thus our superpixel classification problem is converted as modeling the difference between empirical distributions of positive and negative classes. We compute (1) simple statistical features of the first–fourth order statistics such as mean, std, skewness, kurtosis [35] and (2) histogram-type features of eight percentiles (20, 30, ..., 90%), per distribution in intensity or P^{RF} channel, respectively. Once concatenated, the resulted 24 features for each superpixel instance is fed to train random forest classifiers.

Due to the highly unbalanced quantities between foreground (pancreas) superpixels and background (the rest of CT volume) superpixels, a two-tiered cascade of random forests is exploited to address this type of rare event detection problem [36]. In a cascaded classification, C_{SP}^1 once trained is applied exhaustively on scanning all superpixels in an input CT volume. Based on the receiver operating characteristic (ROC) curves for C_{SP}^1 , we can safely reject or prune 97% negative superpixels while maintaining nearly $\sim 100\%$ recall or sensitivity. The remained 3% negatives, often referred as “hard negatives” [36], along with all positives are employed to train the second C_{SP}^2 in the same feature space. Combining C_{SP}^1 and C_{SP}^2 is referred to as Framework 1 (F-1) in the subsequent sections.

Similarly, we can train a random forest classifier C_{SP}^3 by replacing C_{SP}^2 ’s feature extraction dependency on the P^{RF} probability response maps, with the deep CNN patch classification maps of P^{CNN} . The same 24 statistical moments and percentile features per superpixel, from two information channels I^{CT} and P^{CNN} , are extracted to train C_{SP}^3 . Note that the CNN model that produces P^{CNN} is trained with the image patches sampled from only “hard negative” and positive superpixels (aligned with the second-tier RF classifiers C_{SP}^2 and C_{SP}^3). For simplicity, P^{RF} is only trained once with all positive and negative image patches. This will be referred to as Framework 2 (F-2) in the subsequent sections. F-1 only use P^{RF} whereas F-2 depends on both P^{RF} and P^{CNN} (with a little extra computational cost).

The flow chart of frameworks 1 (F-1) and 2 (F-2) is illustrated in Fig. 16.9. The two-level cascaded random forest classification hierarchy is found empirically to be sufficient (although a deeper cascade is possible) and implemented to obtain F-1: C_{SP}^1 and C_{SP}^2 , or F-2: C_{SP}^1 and C_{SP}^3 . The binary 3D pancreas volumetric mask is obtained by stacking the binary superpixel labeling outcomes (after C_{SP}^2 in F-1 or C_{SP}^3 in F-2) for each 2D axial slice, followed by 3D connected component analysis implemented in the end. By assuming the overall pancreas connectivity of its 3D shape, the largest 3D connected component is kept as the final segmentation. The binarization thresholds of random forest classifiers in C_{SP}^2 and C_{SP}^3 are calibrated using data in the training folds in sixfold cross validation, via a simple grid search. In [33], standalone Patch-ConvNet dense probability maps (without any post-processing) are processed for pancreas segmentation after using (F-1) as an initial cascade. The corresponding pancreas segmentation performance is not as accuracy as (F-1) or (F-2).

16.4 Data and Experimental Results

16.4.1 Imaging Data

80 3D abdominal portal-venous contrast-enhanced CT scans (~ 70 s after intravenous contrast injection) acquired from 53 male and 27 female subjects are used in our study for evaluation. Seventeen of the subjects are from a kidney donor transplant list of healthy patients that have abdominal CT scans prior to nephrectomy. The remaining

63 patients are randomly selected by a radiologist from the Picture Archiving and Communications System (PACS) on the population that has neither major abdominal pathologies nor pancreatic cancer lesions. The CT datasets are obtained from National Institutes of Health Clinical Center. Subjects range in the age from 18 to 76 years with a mean age of 46.8 ± 16.7 . Scan resolution has 512×512 pixels (varying pixel sizes) with slice thickness ranging from 1.5 to 2.5 mm on Philips and Siemens MDCT scanners. The tube voltage is 120 kVp. Manual ground-truth segmentation masks of the pancreas for all 80 cases are provided by a medical student and verified/modified by a radiologist.

16.4.2 Experiments

Experimental results are assessed using sixfold cross validation, as described in Sects. 16.3.2 and 16.3.4. Several metrics to evaluate the accuracy and robustness of the methods are computed. The Dice similarity index which interprets the overlap between two sample sets, $SI = 2(|A \cap B|)/(|A| + |B|)$ where A and B refer to the algorithm output and manual ground-truth 3D pancreas segmentation, respectively. The Jaccard index (JI) is another statistic used to compute similarities between the segmentation result against the reference standard, $JI = (|A \cap B|)/(|A \cup B|)$, called “intersection over union” in the PASCAL VOC challenges [37, 38]. The volumetric recall (i.e. sensitivity) and precision values are also reported (Fig. 16.10).

Next, the pancreas segmentation performance evaluation is conducted in respect to the total number of patient scans used for the training and testing phases. Using our framework F1 on 40, 60 and 80 (i.e., 50, 75, and 100% of the total 80 datasets) patient scans, the Dice, JI, Precision, and Recall are computed under sixfold cross validation. Table 16.1 shows the computed results using image patch-level features and multi-level classification (i.e., performing C_{SP}^1 and C_{SP}^2 on I^{CT} and P^{RF}) and how performance changes with the additions of more patients data. Steady improvements of $\sim 4\%$ in the Dice coefficient and $\sim 5\%$ for the Jaccard index are observed, from 40 to 60, and 60–80. Figure 16.11 illustrates some sample final pancreas segmentation results from the 80 patient execution for two different patients. The results are divided into three categories: good, fair, and poor. The good category refers to the computed Dice coefficient above 90% (of 15 patients), fair result as $50\% \leq Dice \geq 90\%$ (49 patients) and poor for $Dice < 50\%$ (16 patients).

Then, we evaluate the difference of the proposed F-1 versus F-2 on 80 patients, using the same four metrics (i.e., Dice, JI, precision, and recall). Table 16.1 shows the comparison results. The same sixfold cross-validation criterion is employed so that direct comparisons can be made. From the table, it can be seen that about 2% increase in the Dice coefficient was obtained by using F-2, but the main improvement can be noticed in the minimum values (i.e., the lower performance bound) for each of the metrics. Usage of deep patch labeling prevents the case of no pancreas segmentation while keeping slightly higher mean precision and recall values. The standard deviations also dropped nearly 50% comparing F-1 to F-2 (from 25.6 to

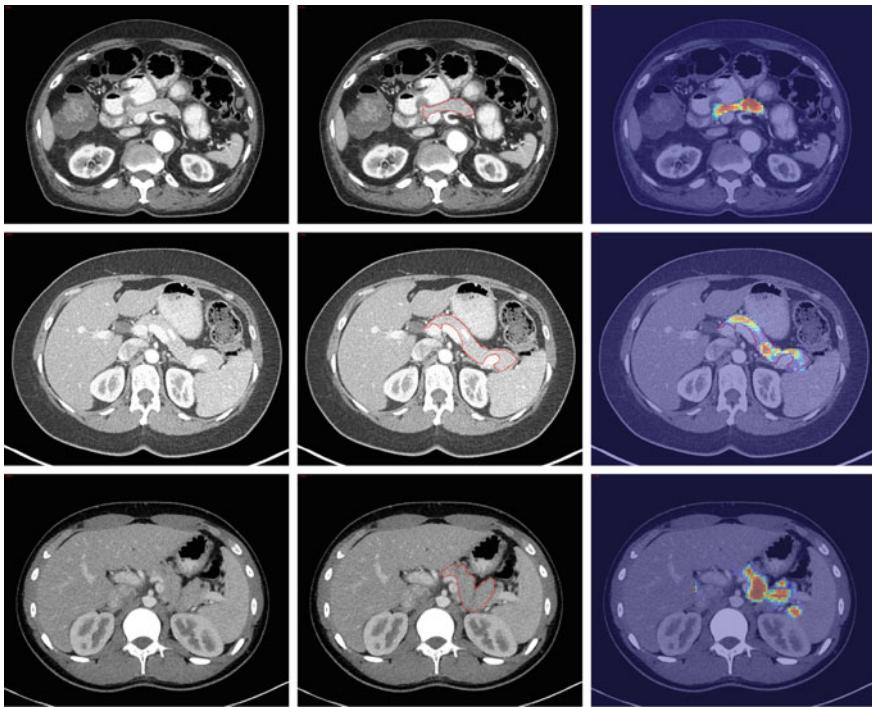


Fig. 16.10 Three examples of deep CNN-based image patch labeling probability response maps per row. Red color shows stronger pancreas class response and blue presents weaker response. From Left, Center to Right are the original CT image, CT image with annotated pancreas contour in red, and CNN response map overlaid CT image

Table 16.1 Examination of varying number of patient datasets using framework 1, in four metrics of Dice, JI, precision, and recall. Mean, standard deviation, lower and upper performance ranges are reported. Comparison of the presented framework 1 (F-1) versus framework 2 (F-2) in 80 patients is also presented

	N	SI (%)	JI (%)	Precision (%)	Recall (%)
F-1	40	60.4 ± 22.3	46.7 ± 22.8	55.6 ± 29.8	80.8 ± 21.2
F-1	60	64.9 ± 22.6	51.7 ± 22.6	70.3 ± 29.0	69.1 ± 25.7
F-1	80	68.8 ± 25.6	57.2 ± 25.4	71.5 ± 30.0	72.5 ± 27.2
F-2	80	70.7 ± 13.0	57.9 ± 13.6	71.6 ± 10.5	74.4 ± 15.1

13.0% in Dice; and 25.4–13.6% in JI). Note that F-1 has the similar standard deviation ranges with the previous methods [5, 7–10] and F-2 significantly improves upon all of them. From Figs. 16.1 and 16.6 it can be inferred that using the relative x-axis and y-axis positions as features aided in reducing the overall false negative rates. Based on Table 16.1, we observe that F-2 provides consistent performance improvements over F-1, which implies that CNN based dense patch labeling shows more

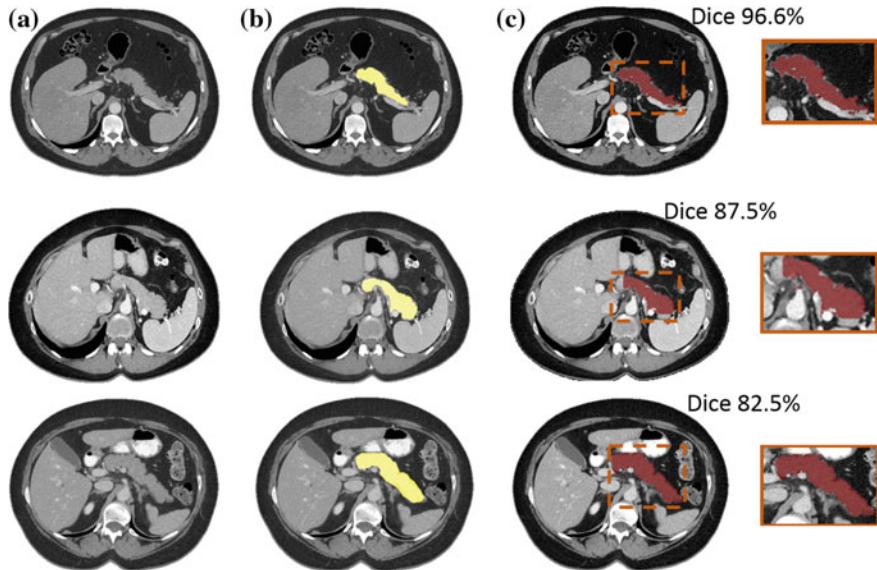


Fig. 16.11 Pancreas segmentation results with the computed Dice coefficients for one good (*Top Row*) and two fair (*Middle, Bottom Rows*) segmentation examples. Sample original CT slices for both patients are shown in (*Left Column*) and the corresponding ground-truth manual segmentation in (*Middle Column*) are in yellow. Final computed segmentation regions are shown in red in (*Right Column*) with Dice coefficients for the volume above each slice. The zoomed-in areas of the slice segmentation in the *orange boxes* are shown to the *right* of the image

promising results (Sect. 16.3.3) than the conventional hand-crafted image features and random forest patch classification alone (Sect. 16.3.2). Figure 16.12 depicts an example patient where F-2 Dice score is improved by 18.6% over F-1 (from 63.9 to 82.5%). In this particular case, the close proximity of the stomach and duodenum to the pancreas head in particular proves challenging for F-1 without the CNN counterpart to distinguish. The surface-to-surface overlays illustrate how both frameworks compare to the ground-truth manual segmentation.

F-1 performs comparably to the state-of-the-art pancreas segmentation methods while F-2 slightly but consistently outperform others, even under sixfold cross validation (CV) instead of the “leave-one-patient-out” (LOO) used in [5–10]. Note that our results are not directly or strictly comparable with [5–10] since different datasets are used for evaluation. If under the same sixfold cross validation, our bottom-up segmentation method can significantly outperform an implemented version of “multi-atlas and label fusion” (MALF) based on [11, 12], on the pancreas segmentation dataset studied in this work. Details are provided later in this section. Table 16.2 reflects the comparison of Dice, JI, precision and recall results, between our methods of F-1, F-2 and other approaches, in multi-atlas registration and label fusion based multi-organ

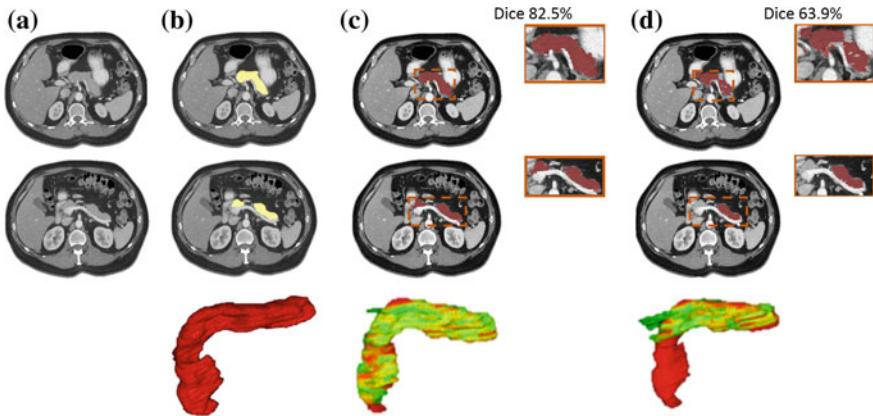


Fig. 16.12 Examples of pancreas segmentation results using F-1 and F-2 with the computed Dice coefficients for one patient. Original CT slices for the patient are shown in **Column a** and the corresponding ground-truth manual segmentation in **Column b** are in yellow. Final computed segmentation using F-2 and F-1 are shown in red in **Columns c, d** with Dice coefficients for the volume above first slice. The zoomed-in areas of the slice segmentation in the *orange boxes* are shown to the *right* of the images. Their surface-to-surface distance map overlaid on the ground-truth mask is demonstrated in **Columns c, d Bottom** and the corresponding ground-truth segmentation mask in **Column b Bottom** are in red. The *red color* illustrates higher difference and *green* for smaller distance

Table 16.2 Comparison of F-1 and F-2 in sixfold cross validation to the recent state-of-the-art methods [5–10] in LOO and our implementation of “multi-atlas and label fusion” (MALF) using publicly available C++ code bases [11, 12] under the same sixfold cross validation. The proposed bottom-up pancreas segmentation methods of F-1 and F-2 significantly outperform their MALF counterpart: $68.8 \pm 25.6\%$ (F-1), $70.7 \pm 13.0\%$ (F-2) versus $52.51 \pm 20.84\%$ in Dice coefficients (mean \pm std)

Reference	N	SI (%)	Jl (%)	Precision (%)	Recall (%)
[5]	20	–	57.9	–	–
[6]	28	–	46.6	–	–
[7]	150	69.6 ± 16.7	55.5 ± 17.1	67.9 ± 18.2	74.1 ± 17.1
[7]	50	58.2 ± 20.0	43.5 ± 17.8	–	–
[9]	100	65.5	49.6	70.7	62.9
[10]	100	65.5 ± 18.6	–	–	–
[8]	100	69.1 ± 15.3	54.6	–	–
Framework 1	80	68.8 ± 25.6	57.2 ± 25.4	71.5 ± 30.0	72.5 ± 27.2
Framework 2	80	70.7 ± 13.0	57.9 ± 13.6	71.6 ± 10.5	74.4 ± 15.1
MALF	80	52.5 ± 20.8	38.1 ± 18.3	–	–

segmentation [6–10] and multi-phase single organ (i.e., pancreas) segmentation [5]. Previous numerical results are found from the publications [5–10]. We choose the best result out of different parameter configurations in [8].

We exploit two variations of pancreas segmentation in a perspective of bottom-up information propagation from image patches to (segments) superpixels. Both frameworks are carried out in a sixfold cross-validation (CV) manner. Our protocol is arguably harder than the “leave-one-out” (LOO) criterion in [5, 7–10] since less patient datasets are used in training and more separate patient scans for testing. In fact, [7] does demonstrate a notable performance drop from using 149 patients in training versus 49 patients under LOO, i.e., the mean Dice coefficients decreased from $69.6 \pm 16.7\%$ to $58.2 \pm 20.0\%$. This indicates that the multi-atlas fusion approaches [5–10] may actually achieve lower segmentation accuracies than reported, if under the sixfold cross-validation protocol. At 40 patients, our result using framework 1 is 2.2% better than the reported results by [7] using 50 patients (Dice coefficients of 60.4% vs. 58.2%). Comparing to the usage of $N - 1$ patient datasets directly in the memory for multi-atlas registration methods, our learned models are more compactly encoded into a series of patch- and superpixel-level random forest classifiers and the CNN classifier for patch labeling. The computational efficiency also has been drastically improved in the order of 6–8 min per testing case (using a mix of Matlab and C implementation, ~50% time for superpixel generation), compared to others requiring 10 hours or more. The segmentation framework (F-2) using deep patch labeling confidences is also more numerically stable, with no complete failure case and noticeable lower standard deviations.

Comparison to R-CNN and its variations [33, 39]: The conventional approach for classifying superpixels or image segments in computer vision is “bag-of-words” [40, 41]. “Bag-of-words” methods compute dense SIFT, HOG, and LBP image descriptors, embed these descriptors through various feature encoding schemes and pool the features inside each superpixel for classification. Both model complexity and computational expense [40, 41] are very high, comparing with ours (Sect. 16.3.4). Recently, a “Regional CNN” (R-CNN) [34, 42] method is proposed and shows substantial performance gains in PASCAL VOC object detection and semantic segmentation benchmarks [37], compared to previous “Bag-of-words” models. A simple R-CNN implementation on pancreas segmentation has been explored in our previous work [39] which reports evidently worse result (Dice coefficient $62.9 \pm 16.1\%$) than our F-2 framework (Dice $70.7 \pm 13.0\%$) that spatially pools the CNN patch classification confidences per superpixel. Note that R-CNN [34, 42] is not an “end-to-end” trainable deep learning system: R-CNN first uses the pretrained or fine-tuned CNNs as image feature extractors for superpixels and then the computed deep image features are classified by support vector machine models.

Our recent work [33] is an extended version of pancreas segmentation from the region-based convolutional neural networks (R-CNN) for semantic image segmentation [37, 42]. In [33], (1) we exploit multi-level deep convolutional networks which sample a set of bounding boxes covering each image superpixel at multiple spatial scales in a zoom-out fashion [43]; (2) the best performing model in [33] is a stacked R^2 -ConvNet which operates in the joint space of CT intensities and the *Patch*-ConvNet dense probability maps, similar to F-2. With the above two method extensions, [33] reports the Dice coefficient of $71.8 \pm 10.7\%$ in fourfold cross val-

idation (which is slightly better than $70.7 \pm 13.0\%$ of F-2 using the same dataset). However, [33] cannot be directly trained and tested on the raw CT scans as in this work, due to the data high-imbalance issue between pancreas and non-pancreas superpixels. There are overwhelmingly more negative instances than positive ones if training the CNN models directly on all image superpixels from abdominal CT scans. Therefore, given an input abdomen CT, an initial set of superpixel regions is first generated or filtered by a coarse cascading process of operating the random forests based pancreas segmentation [44] (similar to F-1), at low or conservative classification thresholds. Over 96% original volumetric abdominal CT scan space has been rejected for the next step. For pancreas segmentation, these pre-labeled superpixels serve as regional candidates with high sensitivity ($>97\%$) but low precision (generally called Candidate Generation or CG process). The resulting initial DSC is 27% on average. Then [33] evaluates several variations of CNNs for segmentation refinement (or pruning). F-2 performs comparably to the extended R-CNN version for pancreas segmentation [33] and is able to run without using F-1 to generate pre-selected superpixel candidates (which nevertheless is required by [33, 39]). As discussed above, we would argue that these hybrid approaches combining or integrating deep and non-deep learning components (like this work and [33, 34, 39, 42, 45]) will co-exist with the other fully “end-to-end” trainable CNN systems [46, 47] that may produce comparable or even inferior segmentation accuracy levels. For example, [45] is a two-staged method of deep CNN image labeling followed by fully connected Conditional Random Field (CRF) post-optimization [48], achieving 71.6% intersection-over-union value versus 62.2% in [47], on PASCAL VOC 2012 test set for semantic segmentation task [37].

Comparison to MALF (under sixfold CV): For the ease of comparison to the previously well studied “multi-atlas and label fusion” (MALF) approaches, we implement a MALF solution for pancreas segmentation using the publicly available C++ code bases [11, 12]. The performance evaluation criterion is the same **sixfold patient splits for cross validation**, not the “leave-one-patient-out” (LOO) in [5–10]. Specifically, each atlas in the training folds is registered to every target CT image in the testing fold, by the fast free-form deformation algorithm developed in NiftyReg [11]. Cubic B-Splines are used to deform a source image to optimize an objective function based on the normalized mutual information and a bending energy term. Grid spacing along three axes are set as 5 mm. The weight of the bending energy term is 0.005 and the normalized mutual information with 64 bins are used. The optimization is performed in three coarse-to-fine levels and the maximal number of iterations per level is 300. More details can be found in [11]. The registrations are used to warp the pancreas in the atlas set (66, or 67 atlases) to the target image. Nearest neighbor interpolation is employed since the labels are binary images. For each voxel in the target image, each atlas provided an opinion about the label. The probability of pancreas at any voxel x in the target U was determined by $\hat{L}(x) = \sum_{i=1}^n \omega_i(x)L_i(x)$ where $L_i(x)$ is the warped i -th pancreas atlas and $\omega_i(x)$ is a weight assigned to the i -th atlas at location x with $\sum_{i=1}^n \omega_i(x) = 1$; and n is the number of atlases. In our sixfold cross validation experiments $n = 66$ or 67. We adopt the joint label fusion algorithm [12], which estimates

voting weights $\omega_i(x)$ by simultaneously considering the pairwise atlas correlations and local image appearance similarities at x . More details about how to capture the probability that different atlases produce the same label error at location x via a formulation of dependency matrix can be found in [12]. The final binary pancreas segmentation label or map $L(x)$ in target can be computed by thresholding on $\hat{L}(x)$. The resulted MALF segmentation accuracy in Dice coefficients are $52.51 \pm 20.84\%$ in the range of [0, 80.56%]. This pancreas segmentation accuracy is noticeably lower than the mean Dice scores of 58.2–69.6% reported in [5–10] under the protocol of “leave-one-patient-out” (LOO) for MALF methods. This observation may indicate the performance deterioration of MALF from LOO (equivalent to 80-fold CV) to sixfold CV which is consistent with the finding that the segmentation accuracy drops from 69.6 to 58.2% when only 49 atlases are available instead of 149 [7].

Furthermore, we take about 33.5 days to fully conduct the sixfold MALF cross-validation experiments using a Windows server; whereas the proposed bottom-up superpixel cascade approach finishes in ~ 9 h for 80 cases (6.7 min per patient scan on average). In summary, using the same dataset and under sixfold cross validation, our bottom-up segmentation method significantly outperforms its MALF counterpart: $70.7 \pm 13.0\%$ versus $52.51 \pm 20.84\%$ in Dice coefficients, while being approximately 90 times faster. Converting our Matlab/C++ implementation into pure C++ should expect further 2–3 times speed-up.

16.5 Conclusion and Discussion

In this chapter, we present a fully-automated bottom-up approach for pancreas segmentation in abdominal computed tomography (CT) scans. The proposed method is based on a hierarchical cascade of information propagation by classifying image patches at different resolutions and multi-channel feature information pooling at (segments) superpixels. Our algorithm flow is a sequential process of decomposing CT slice images as a set of disjoint boundary-preserving superpixels; computing pancreas class probability maps via dense patch labeling; classifying superpixels via aggregating both intensity and probability information to form image features that are fed into the cascaded random forests; and enforcing a simple spatial connectivity based post-processing. The dense image patch labeling can be realized by efficient random forest classifier on handcrafted image histogram, location and texture features; or deep convolutional neural network classification on larger image windows (i.e., with more spatial contexts).

The main component of our method is to classify superpixels into either pancreas or non-pancreas class. Cascaded random forest classifiers are formulated for this task and performed on the pooled superpixel statistical features from intensity values and supervisedly learned class probabilities (P^{RF} and/or P^{CNN}). The learned class probability maps (e.g., P^{RF} and P^{CNN}) are treated as the supervised semantic class image embeddings which can be implemented, via an open framework by various methods, to learn the per-pixel class probability response.

To overcome the low image boundary contrast issue in superpixel generation, which is however common in medical imaging, we suggest that efficient supervised edge learning techniques may be utilized to artificially “enhance” the strength of semantic object-level boundary curves in 2D or surface in 3D. For example, one of the future directions is to couple or integrate the structured random forests based edge detection [49] into a new image segmentation framework (MCG: Multiscale Combinatorial Grouping) [50] which permits a user-customized image gradient map. This new approach may be capable to generate image superpixels that can preserve even very weak semantic object boundaries well (in the image gradient sense) and subsequently prevent segmentation leakage.

Finally, voxel-level pancreas segmentation masks can be propagated from the stacked superpixel-level classifications and further improved by an efficient boundary refinement post-processing, such as the narrow-band level-set based curve/surface evolution [26], or the learned intensity model based graph-cut [7]. Further examination into the sub-connectivity processes for the pancreas segmentation framework that considers the spatial relationships of splenic, portal, and superior mesenteric veins with pancreas may be needed for future work.

References

1. Cuingnet R, Prevost R, Lesage D, Cohen L, Mory B, Ardon R (2012) Automatic detection and segmentation of kidneys in 3d CT images using random forests. In: MICCAI, pp 66–74
2. Mansoor A, Bagci U, Xu Z, Foster B, Olivier K, Elinoff J, Suffredini A, Udupa J, Mollura D (2014) A generic approach to pathological lung segmentation. IEEE Trans Med Imaging 33(12):2293–2310
3. Zheng Y, Barbu A, Georgescu B, Scheuering M, Comaniciu D (2008) Four-chamber heart modeling and automatic segmentation for 3d cardiac CT volumes using marginal space learning and steerable features. IEEE Trans Med Imaging 27(11):1668–1681
4. Ling H, Zhou S, Zheng Y, Georgescu B, Suhling M, Comaniciu D (2008) Hierarchical, learning-based automatic liver segmentation. In: IEEE conference on CVPR, pp 1–8
5. Shimizu A, Kimoto T, Kobatake H, Nawano S, Shinozaki K (2010) Automated pancreas segmentation from three-dimensional contrast-enhanced computed tomography. Int J Comput Assist Radiol Surg 5(1):85–98
6. Okada T, Linguraru M, Yoshida Y, Hor M, Summers R, Chen Y, Tomiyama N, Sato Y (2012) Abdominal multi-organ segmentation of CT images based on hierarchical spatial modeling of organ interrelations. In: Abdominal imaging - computational and clinical applications, pp 173–180
7. Wolz R, Chu C, Misawa K, Fujiwara M, Mori K, Rueckert D (2013) Automated abdominal multi-organ segmentation with subject-specific atlas generation. IEEE Trans Med Imaging 32(7):1723–1730
8. Chu C, Oda M, Kitasaka T, Misawa K, Fujiwara M, Hayashi Y, Nimura Y, Rueckert D, Mori K (2013) Multi-organ segmentation based on spatially-divided probabilistic atlas from 3d abdominal CT images. In: MICCAI, vol 2, pp 165–172
9. Wolz R, Chu C, Misawa K, Mori K, Rueckert D (2012) Multi-organ abdominal CT segmentation using hierarchically weighted subject-specific atlases. In: MICCAI, vol 1, pp 10–17
10. Wang Z, Bhatia K, Glocker B, Marvao A, Dawes T, Misawa K, Mori K, Rueckert D (2014) Geodesic patch-based segmentation. In: MICCAI, vol 1, pp 666–673

11. Modat M, McClelland J, Ourselin S (2010) Lung registration using the niftyreg package. In: Medical image analysis for the clinic-a grand challenge, pp 33–42
12. Wang H, Suh J, Das S, Pluta J, Craige C, Yushkevich P (2012) Multi-atlas segmentation with joint label fusion. *IEEE Trans Pattern Anal Mach Intell* 35(3):611–623
13. Zografos V, Menze B, Tombari F (2015) Hierarchical multi-organ segmentation without registration in 3d abdominal CT images. In: MICCAI medical computer vision workshop
14. Lucchi A, Smith K, Achanta R, Knott G, Fua P (2012) Supervoxel-based segmentation of mitochondria in EM image stacks with learned shape features. *IEEE Trans Med Imaging* 31(2):474–486
15. Lu L, Wu D, Lay N, Liu D, Nogues I, Summers R (2016) Accurate 3d bone segmentation in challenging CT images: bottom-up parsing and contextualized optimization. In: IEEE conference on WACV, pp 1–10
16. Lu L, Barbu A, Wolf M, Liang J, Bogoni L, Salganicoff M, Comaniciu D (2008) Simultaneous detection and registration for ileo-cecal valve detection in 3d CT colonography. In: European conference on computer vision. Springer, Berlin, pp 465–478
17. Liu M, Lu L, Ye X, Yu S, Huang H (2011) Coarse-to-fine classification using parametric and nonparametric models for computer-aided diagnosis. In: 20th ACM conference on information and knowledge management
18. Lu L, Devarakota P, Vikal S, Wu D, Zheng Y, Wolf M (2013) Computer aided diagnosis using multilevel image features on large-scale evaluation. In: Medical computer vision-MICCAI. Springer, Berlin, pp 161–174
19. Achanta R, Shaji A, Smith K, Lucchi A, Fua P, Susstrunk S (2012) Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans Pattern Anal Mach Intell* 34(11):2274–2282
20. Neubert P, Protzel P (2012) Superpixel benchmark and comparison. In: Proceedings of the forum Bildverarbeitung, pp 1–12
21. Liu M, Tuzel O, Ramalingam S, Chellappa R (2011) Entropy rate superpixel segmentation. In: IEEE conference on CVPR, pp 2099–2104
22. Felzenszwalb P, Huttenlocher D (2004) Efficient graph-based image segmentation. *Int J Comput Vis* 59(2):167–181
23. Vincent L, Soille P (1991) Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Trans Pattern Anal Mach Intell* 13(6):583–598
24. Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. *IEEE Trans Pattern Anal Mach Intell* 24(5):603–619
25. Cour T, Beneit F, Shi J (2005) Spectral segmentation with multiscale graph decomposition. In: IEEE proceedings on CVPR
26. Kohlberger T, Sofka M, Zhang J, Birkbeck N, Wetzl J, Kaftan J, Declerck J, Zhou S (2011) Automatic multi-organ segmentation using learning-based segmentation and level set optimization. In: MICCAI, vol 3, pp 338–345
27. Vedaldi A, Fulkerson B (2008) VLFeat: an open and portable library of computer vision algorithms. <http://www.vlfeat.org/>
28. Lowe D (2004) Distinctive image features from scale-invariant keypoints. *Int J Comput Vis* 60(2):91–110
29. Gilinsky A, Zelnik-Manor L (2013) Siftpack: a compact representation for efficient sift matching. In: IEEE conference on ICCV, pp 777–784
30. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15(1):1929–1958
31. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
32. Gao M, Bagci U, Lu L, Wu A, Buty M, Shin H-C, Roth H, Papadakis GZ, Depenninge A, Summers RM et al (2016) Holistic classification of CT attenuation patterns for interstitial lung diseases via deep convolutional neural networks. *Comput Methods Biomed Eng: Imaging Vis* 1–6
33. Roth H, Lu L, Farag A, Shin H, Liu J, Turkbey E, Summers R (2015) Deeporgan: multi-level deep convolutional networks for automated pancreas segmentation. In: MICCAI, pp 556–564

34. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE conference on CVPR, pp 580–587
35. Groeneveld R, Meeden G (1984) Measuring skewness and kurtosis. Stat 33:391–399
36. Viola P, Jones M (2004) Robust real-time face detection. Int J Comput Vis 57(2):137–154
37. Everingham M, Eslami S, Van Gool L, Williams C, Winn J, Zisserman A (2015) The pascal visual object classes challenge: a retrospective. Int J Comput Vis 111(1):98–136
38. Carreira J, Sminchisescu C (2012) CPMC: automatic object segmentation using constrained parametric min-cuts. IEEE Trans Pattern Anal Mach Intell 34(7):1312–1328
39. Roth H, Farag A, Lu L, Turkbey E, Liu J, Summers R (2015) Deep convolutional networks for pancreas segmentation in CT imaging. In: SPIE conference on medical imaging, pp 1–8
40. Carreira J, Caseiro R, Batista J, Sminchisescu C (2012) Semantic segmentation with second-order pooling. In: European conference on computer vision, pp 430–443
41. Chatfield K, Lempitsky V, Vedaldi A, Zisserman A (2011) The devil is in the details: an evaluation of recent feature encoding methods. In: BMVC, pp 1–12
42. Girshick R, Donahue J, Darrell T, Malik J (2015) Region-based convolutional networks for accurate object detection and semantic segmentation. IEEE Trans Pattern Anal Mach Intell, to appear
43. Mostajabi M, Yadollahpour P, Shakhnarovich G (2015) Feedforward semantic segmentation with zoom-out features. In: IEEE conference on CVPR, pp 3376–3385
44. Farag A, Lu L, Liu J, Turkbey E, Summers R (2010) A bottom-up approach for automatic pancreas segmentation abdominal CT scans. In: MICCAI abdominal imaging workshop
45. Chen L, Papandreou G, Kokkinos I, Murphy K, Yuille A (2015) Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: International conference on learning representation
46. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: MICCAI, pp 234–241
47. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: IEEE conference on CVPR, pp 3431–3440
48. Krhenbhl P, Koltun V (2011) Efficient inference in fully connected CRFs with Gaussian edge potentials. In: NIPS, pp 109–117
49. Dollr P, Zitnick L (2015) Fast edge detection using structured forests. IEEE Trans Pattern Anal Mach Intell 37:1558–1570
50. Arbelaez P, Pont-Tuset J, Barron J, Marqus F, Malik J (2014) Multiscale combinatorial grouping. In: IEEE conference on CVPR, pp 328–335

Part IV

Big Dataset and Text-Image Deep Mining

Chapter 17

Interleaved Text/Image Deep Mining on a Large-Scale Radiology Image Database

Hoo-Chang Shin, Le Lu, Lauren Kim, Ari Seff, Jianhua Yao
and Ronald Summers

Abstract Exploiting and effective learning on very large-scale ($>100K$ patients) medical image databases have been a major challenge in spite of noteworthy progress in computer vision. This chapter suggests an interleaved text/image deep learning system to extract and mine the semantic interactions of radiologic images and reports, from a national research hospital’s Picture Archiving and Communication System. This chapter introduces a method to perform unsupervised learning (e.g., latent Dirichlet allocation, feedforward/recurrent neural net language models) on document- and sentence-level texts to generate semantic labels and supervised deep ConvNets with categorization and cross-entropy loss functions to map from images to label spaces. Keywords can be predicted for images in a retrieval manner, and presence/absence of some frequent types of disease can be predicted with probabilities. The large-scale datasets of extracted key images and their categorization, embedded vector labels, and sentence descriptions can be harnessed to alleviate deep learning’s “data-hungry” challenge in the medical domain.

17.1 Introduction

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [1, 2] provides more than one million labeled images from 1,000 object categories. The accessibility of huge amount of well-annotated image data in computer vision rekindles deep convolutional neural networks (Convnets or CNNs) [3–5] as a premier learning tool, to solve the visual object class recognition tasks. Deep CNNs can perform significantly better than traditional shallow learning methods but much more training data are required [2, 3]. In the medical domain, however, there are no similar very large-scale

H.-C. Shin · L. Lu (✉) · L. Kim · A. Seff · J. Yao · R. Summers
Imaging Biomarkers and Computer-Aided Diagnosis Laboratory, Radiology and Imaging Sciences, National Institutes of Health Clinical Center, Bethesda, MD 20892-1182, USA
e-mail: le.lu@nih.gov

R. Summers
e-mail: rms@nih.gov

labeled image datasets available. On the other hand, gigantic collections of radiologic images and reports are stored in many modern hospitals' Picture Archiving and Communication Systems (PACS). The invaluable semantic diagnostic knowledge inhabiting the mapping between hundreds of thousands of clinician-created high-quality text reports and linked image volumes remains largely unexplored. One of our primary goals is to extract and associate radiologic images with clinically semantic scalar and vector labels, via interleaved text/image data mining and deep learning on a very large-scale PACS database ($\sim 780K$ imaging examinations). Scalar labels offer image categorization [2, 6] and vector labels embed a low-dimensional vector distance space for high-level tasks of image to disease terms auto-reporting [7, 8].

Building ImageNet database is mainly a manual process [1]: harvesting images returned from Google image search engine (according to the WordNet ontology hierarchy) and pruning falsely tagged images using crowd-sourcing, as Amazon Mechanical Turk (AMT). This does not facilitate our data collection and labeling needs due to the demanding difficulties of medical annotation tasks by general AMT knowledge workers and data privacy reasons. Thus we propose to mine image categorization labels from hierarchical, Bayesian document-clustering, e.g., generative latent Dirichlet allocation (LDA) topic model [9], using $\sim 780K$ high-quality radiology text reports in PACS. These reports contain natural language understanding grade semantics for the diagnostic descriptions or impressions for the linked images (in the same case), but are not machine-trainable labels. The radiology reports are text documents describing patient history, symptoms, image (understudied) observations and impression by board-certified radiologists, but do not contain machine trainable labels. We find that LDA generated image categorization labels are indeed valid, demonstrating good semantic coherence of clinician observers [10, 11], and can be effectively learned using 8-layer and 19-layer deep ConvNets using image inputs alone [3, 4]. The more recent deeper CNN model [4] also outperforms the standard one [3] (validation image recognition accuracy of 0.67 vs. 0.59 in the first level document-clustering labels of 80 classes, and 0.49 vs. 0.26 for the second level of 800 classes). Our deep CNN models on medical image modalities (mostly CT, MRI) are initialized with the model weights pre-trained from ImageNet [1] using Caffe [12], analogous to the deep feature generality from color to depth [13] and from natural to fMRI images [14].

Kulkarni et al. [8] have spearheaded the efforts of learning/generating the semantic connections between image contents and the sentences describing them (i.e., captions). Detecting object(s)/stuff of interest, attributes and prepositions and applying contextual regularization by conditional random field (CRF) is a feasible approach [8] because many useful tools are available in computer vision. There has not yet been much comparable development on large scale medical imaging understanding. Here we take a whole image based representation rather than object-centric formulation as [8]. From a large collection of radiology reports (see Fig. 17.1), we can extract the sentences describing key images (as "key frames in videos") using natural language processing. The tagged disease-related terms in these sentences are mapped into $\mathbb{R}^{1 \times 256}$ vectors using Google's *word2vec* tool [15, 16], trained on a corpus of ~ 1.2

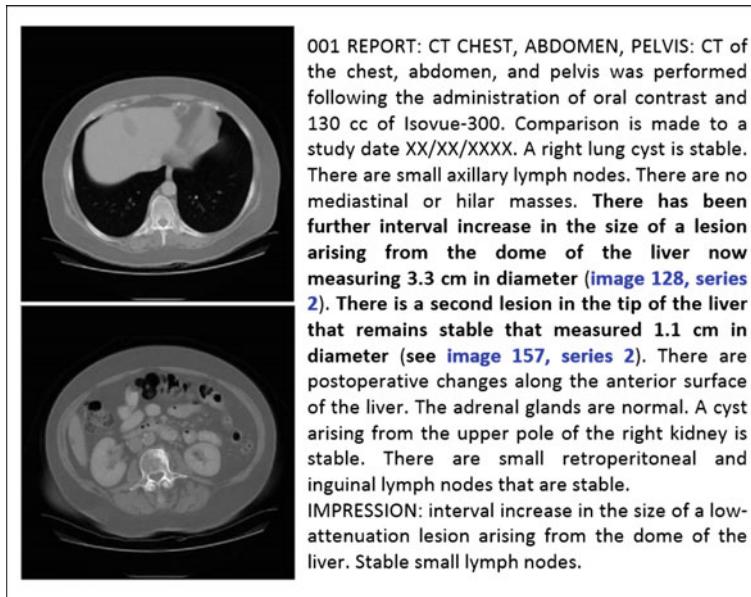


Fig. 17.1 Example of reports and key images

billion words from medical documents. Furthermore key images can be trained to directly predict disease-related terms using deep CNN with regression loss functions. The projected vector space potentially permits neural code based retrieval [7, 17] and auto-reporting [8].

While the keyword generation on medical images can provide a first-level interpretation of a patient's image, labeling based on categorization can be nonspecific. We further suggest mining more specific disease words in the reports mentioning the images to alleviate the issue of nonspecificity. Feedforward CNNs can be then used to train and predict the presence/absence of the specific disease categories.

17.1.1 Related Work

ImageCLEF medical image annotation tasks (2005–2007) have 9,000 training and 1,000 testing 2D images (converted as 32×32 pixel thumbnails [18]) with 57 labels. Local image descriptors and intensity histograms are used in a bag-of-features approach for this scene recognition-like problem. The 2013 ImageCLEF image modality classification challenge has 31 manually labeled categories (including non-medical), 2,845 training and 2,582 testing images. Image only, caption text only, and image-text joint inference are the three evaluation protocols. Note that ImageCLEF tasks are pseudo-artificial, having very limited implications on real clinical

diagnosis. Unsupervised Latent Dirichlet Allocation based matching from lung disease words (e.g., fibrosis, normal, emphysema) in radiology reports to 2D image blocks from axial CT chest scans (of 24 patients) is studied in [19]. This work is motivated by generative models of combining words and images [20, 21], under very limited word/image vocabulary, and remains mostly unknown in the last decade.

In the works [10, 22] the words were first mapped into vector space using recurrent neural networks, then images were projected into the label-associated word-vector embeddings, via minimizing the L_2 [22] or hinge rank losses [10] between the visual and label manifolds. The language model is trained on Wikipedia and tested on label-associated images from the CIFAR [22, 23] and ImageNet [1, 10] datasets. In comparison, our work is on a large corpus of unlabeled medical dataset of associated images and text, where the text-derived labels are computed and verified with human interventions. Graphical models are employed to predict the image attributes [24, 25], or to describe the images [8] using manually annotated datasets [26, 27]. Automatic label mining on large unlabeled datasets is presented in [28, 29], however the variety of the label space is limited (image caption/annotations). We analyze/mine the medical image semantics on both document and sentence levels. Deep CNNs are adopted to learn from image contents [4, 28].

17.2 Data

To gain the most comprehensive understanding on diagnostic semantics, we use all available radiology reports of $\sim 780k$ imaging examinations, stored in a national research hospital’s PACS since year 2000. $\sim 216K$ key 2D image slices (instead of $\sim 780k$ 3D image volumes) are studied here. The reason is that out of 3D patient scans, most imaging information represented are normal anatomies, i.e., not the focuses to be described in radiology reports. These key images were referenced (See Fig. 17.1) by radiologists manually during report writing, to provide a visual reference to pathologies or other notable findings. Key 2D images are more correlated with the diagnostic semantics in the reports than 3D scans. Not all patients have referenced key images (215,786 from 61,845 unique patients). Table 17.1 provides extracted database statistics. Table 17.2 shows examples of the most frequently occurring words in radiology reports. Leveraging on our deep learning models exploited in this paper, will make it possible to automatically select key images from 3D patient scans, to avoid mis-referencing.

Finding and extracting key images from radiology reports are done by natural language processing (NLP), i.e., finding a sentence mentioning referenced image. For example, “*There may be mild fat stranding of the right parapharyngeal soft tissues (series 1001, image 32)*” is listed in Fig. 17.1. The NLP steps are sentence tokenization, word matching/stemming, relation matching (does this number refer to image or series), and rule-based information extraction (e.g., translating “image 1013-78” to “images 1013-1078”). A total of $\sim 187k$ images can be retrieved and matched in this manner, whereas the rest of $\sim 28k$ key images are extracted according to their

Table 17.1 Statistics of the dataset. “Others” include CR (Computed Radiography), RF (Radio Fluoroscopy), US (Ultrasound)

Total number of		# words in documents		# image modalities	
# documents	~780k	mean	131.13	CT	~169k
# images	~216k	std	95.72	MR	~46k
# words	~1 billion	max	1502	PET	67
# vocabulary	~29k	min	2	Others	34

Table 17.2 Examples of the most frequently occurring words in the documents

Right	937k	Images	312k	Contrast	260k	Unremarkable	195k
Left	870k	Seen	299k	Axial	253k	Lower	195k
Impression	421k	Mass	296k	Lung	243k	Upper	192k
Evidence	352k	Normal	278k	Bone	219k	Lesion	180k
Findings	340k	Small	275k	Chest	208k	Lobe	174k
CT	312k	Noted	263k	MRI	204k	Pleural	172k

reference accession numbers in PACS. Our report extracted key image database is the largest one ever reported and highly representative to the huge collection of radiology diagnostic semantics over that last decade. Exploring effective deep learning models on this database opens new ways to parse and understand large-scale radiology image informatics.

17.3 Document Topic Learning with Latent Dirichlet Allocation

We propose to mine image categorization labels from unsupervised hierarchical, Bayesian document-clustering, e.g., generative latent Dirichlet allocation (LDA) topic model [9], using ~780K high-quality radiology text reports in PACS. Unlike images from ImageNet [1, 2] often with a dominate object appearing in the center, our key images are CT/MRI slices showing several coexisting organs/pathologies. There are high amounts of intrinsic ambiguity to define and assign a semantic label set to images, even for experienced clinicians. Our *hypothesis* is that the large collection of sub-million radiology reports statistically defines the categories meaningful to topic-mining (LDA) and visual learning (deep CNN).

LDA [9] is originally proposed to find latent topic models for a collection of text documents (e.g., newspapers). There are some other popular methods for document topic modeling, such as Probabilistic Latent Semantic Analysis (pLSA) [30] and Non-negative Matrix Factorization (NMF) [31]. We choose LDA for extracting latent topic labels among radiology report documents because LDA is shown to be more flexible

yet learns more coherent topics over large sets of documents [32]. Furthermore, pLSA can be regarded as a special case of LDA [33], and NMF as a semi-equivalent model of pLSA [34, 35]. LDA or partially labeled LDA topic models are considered as the most state-of-the-art techniques in [36].

LDA offers a hierarchy of extracted topics and the number of topics can be chosen by evaluating each model's *perplexity score* (Eq. 17.1), which is a common way to measure how well a probabilistic model generalizes by evaluating the log-likelihood of the model on a held-out test set. For an unseen document set D_{test} , the perplexity score is defined as in Eq. 17.1, where M is the number of documents in the test set (unseen hold-out set of documents), \mathbf{w}_d the words in the unseen document d , N_d the number of words in document d , with Φ the topic matrix, and α the hyperparameter for topic-distribution of documents.

$$\text{perplexity}(D_{test}) = \exp \left\{ -\frac{\sum_{d=1}^M \log p(\mathbf{w}_d | \Phi, \alpha)}{\sum_{d=1}^M N_d} \right\} \quad (17.1)$$

Lower perplexity score generally implies better fit of the model for a given document set [9].

Based on the perplexity score evaluated on 80% of the total documents used for training and 20% used for testing, the number of topics chosen is 80 for the document-level using perplexity scores for model selection. Although the document numbers in the topic space are approximately balanced, the distribution of image counts for topics are unbalanced, especially topic 77# consuming nearly half of the $\sim 216K$ key images. Ten second-hierarchy topic models are obtained on each of the first document-level topics, resulting 800 topics, where the number of second-hierarchy topics is also chosen based on the average perplexity scores evaluated on each document-level topics. To test the hypothesis of using the whole reports or only sentence directly describing the key images for latent topic mining, sentence-level LDA topics are obtained via only three sentences: the sentence mentioning the key image (Fig. 17.1), and the previous and following sentences as its proximity context. The perplexity scores keep decreasing with the increasing number of topics, and we chose the topic number 1000, as the rate of the perplexity score decreasing is very small beyond that point.

We observe that LDA generated image categorization labels are indeed valid, demonstrating good semantic coherence of clinician observers [10, 11]. The lists of key words and sampled images per topic label are subjective to two board-certified radiologists' review and validation. Examples of document-level topics with their corresponding images and topic key words are given in Fig. 17.2. Based on radiologists' review, our LDA topics discover semantics at different levels: 73 low-level concepts (e.g., pathology examination of certain body regions and organs: topic 47# of Sinus diseases; 2# Lesions of in solid abdominal organs, primarily kidney; 10# CT of pulmonary diseases; 13# Brain MRI; 19# Renal diseases, mixed imaging modalities; 36# Brain tumors), 7 mid- to high-level concepts (e.g., topic 77# including non-primary metastasis spreading across a variety of body parts, topic 79#

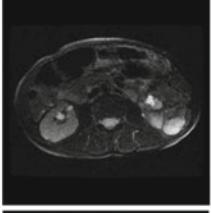
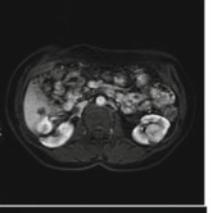
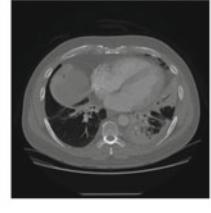
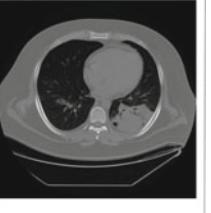
			
Topic 02: Lesions in solid abdominal organs, primarily kidney - keywords: kidney, renal, adrenal, mass, lesion, lesions, cysts, pole, cystic, solid, ...	Topic 10: CT of pulmonary disease - keywords: lobe, lung, upper, lower, chest, pulmonary, infiltrate, middle, scarring, disease, findings, consolidation, new, ...		

Fig. 17.2 Examples of document-level topics with their corresponding images and topic key words

addressing cases with high diagnosis uncertainty/equivocation, 72# Indeterminant lesion, 74# Instrumentation artifacts limiting interpretation). Low-level topic images are visually more coherent (i.e., may be easier to learn). High-level topics may show analogy to [37, 38]. About half of key images are associated with topic 77#, implying that the clinicians' image referencing behavior patterns heavily focus on metastatic patients. Even though LDA labels are computed with text information only, we next investigate the learnability of mapping key images towards these topic labels (at all semantic levels) via deep CNN models.

17.4 Image to Document Topic Mapping with Deep Convolutional Neural Networks

For each level of topics discussed in Sect. 17.3, we train deep ConvNets to map images into document categories, under Caffe [12] framework. While the images of some topic categories and some body parts are easily distinguishable (e.g. Fig. 17.2), the visual differences in abdominal parts are rather subtle. Distinguishing subtleties and high-level concept categories in the images could benefit from a more sophisticated model so that the model can handle these subtleties and complexities.

We split our whole key image datasets as follows: 85% used as the training dataset, 5% as the cross-validation (CV) and 10% as the test dataset. If a topic has too few images so that it cannot be divided into training/CV/test for deep CNN learning (normally rare imaging protocols), then that topic is neglected for CNN training

(e.g., topic 5# Abdominal ultrasound, 28#, 49# DEXA scans of different usages). In total, 60 topics were used for the document-level image-topic mapping, 385 for the second-hierarchy-document-level mapping, and 717 for the sentence-level mapping. Surprisingly, we find that transfer learning from the ImageNet pre-trained CNN parameters on natural images to our medical image modalities (mostly CT, MRI) significantly helps the image classification performance. Thus our convolutional and fully connected CNN layers are fine-tuned from the ImageNet model by default. Similar findings of the deep feature generality across different image modalities have been reported [13, 14] but are empirically verified with only much smaller datasets than ours. Our key image dataset is $\sim 1/5$ size of ImageNet [2], as the largest annotated medical image dataset to date.

Implementation and Results: All our CNN network settings are exactly same as the ImageNet Challenge “AlexNet” [3] and “VGG-19” [4] systems. For image categorization, we change the numbers of output nodes in the last softmax classification layer, i.e., 60, 385 and 717 for the document-level, document-level-h2, and sentence-level respectively. The networks are fine-tuned from the pre-trained ImageNet models so that our learning rates are smaller. For all the layers except the newly modified ones, the learning rate is set 0.001 for weights and biases, momentum 0.9, weight decay 0.0005 and a smaller batch size 50 (as opposed to 256 [4]). These adapted layers are initialized from random and their learning rates are set higher: learning rate: 0.01 for weight, 0.02 for bias, weight decay: 1 for weight, 0 for bias. All key images are resampled to the spatial resolution of 256×256 pixels, mostly from the original 512×512 . Then we follow [4] to crop the input images from 256×256 to 227×227 for training.

We would expect different learning difficulties or classification accuracies among LDA induced topics. Low-level classes can have key images of axial/sagittal/coronal slices with position variations and across MRI/CT modalities. Some body parts and topics, e.g., 63# Pelvic (female reproductive tract) imaging, are visually more challenging than others. Mid- to high-level concepts all demonstrate much larger visual appearance within-class variations since they are diseases occurring at different organs and only coherent at high level semantics. Table 17.3 provides the validation and top-1, top-5 testing in classification accuracy scores for each level of topic models using AlexNet [3] and VGG-19 [4] based deep CNN models. Out of three tasks, document-level-h2 is the hardest while document-level being relatively the easiest.

Table 17.3 Validation and top-1, top-5 test scores in classification accuracy using AlexNet [3] and VGG-19 [4] deep CNN models

	AlexNet 8-layers			VGG 19-layers		
	CV	top-1	top-5	CV	top-1	top-5
document-level	0.61	0.61	0.93	0.66	0.66	0.95
document-level-h2	0.35	0.33	0.56	0.67	0.66	0.84
sentence-level	0.48	0.48	0.56	0.51	0.50	0.58

Our top-1 testing accuracies are closely comparable with the validation ones, showing good training/testing generality and no observable over-fitting. All top-5 accuracy scores are significantly higher than top-1 values (increasing from 0.658 to 0.946 using VGG-19, or 0.607 to 0.929 via AlexNet in document-level), which indicates the classification errors or fusions are not uniformly distributed among other false classes. Latent “blocky subspace of classes” may exist (i.e., several topic classes form a tightly correlated subgroup) in our discovered label space.

It is shown that the deeper 19-layer model (VGG-19) [4] performs consistently better than the 8-layer model (AlexNet) [3] in classification accuracy, especially for document-level-h2. Compared with ImageNet 2014 results, top-1 error rates are moderately higher (34% vs. 30%) and top-5 test errors (6–8%) are comparable. In summary, our quantitative results are very encouraging, given less image categorization labels (60 vs. 1000) but much higher annotation uncertainties by unsupervised LDA topic models. Multi-level semantic concepts show good image learnability by deep CNN models which shed light on automatic parsing very large-scale radiology image databases.

17.5 Generating Image-to-Text Description

The deep CNN image categorization on multi-level document topic labels in Sect. 17.4 demonstrate promising results. The ontology of document-clustering discovered categories needs to be further reviewed and refined through a “with clinician in-the-loop” process. However, it is too expensive and time-consuming for radiologists to examine all of the 1880 (80 + 800 + 1000) topics generated with their keywords and images. Moreover, generating image descriptions as in [8, 24, 25] will result to be more easily understandable than the class label outputs of distinct document topic levels. To more precisely understand the semantic contents from a given key image, we propose to generate relevant keyword text descriptions [8] using deep language/image learning models.

ImageNet database [1] and ImageNet challenge has been very successful in pushing the image classification to the next level. However, in the real world there are so many labels, which are really difficult to be captured by manual labeling, label asreadily alignment and training/classification using a softmax cost function.

Also, in trying to mine labels from clinical texts, so that we can get and train labels for radiology images, there exists many ambiguities in text-based labeling space, e.g., [mr, mri, t1-/t2-¹ weighted], [cyst, cystic, cysts], [tumor, tumour, tumors, metastasis, metastatic], etc.

¹Imaging modalities of magnetic resonance imaging (MRI).

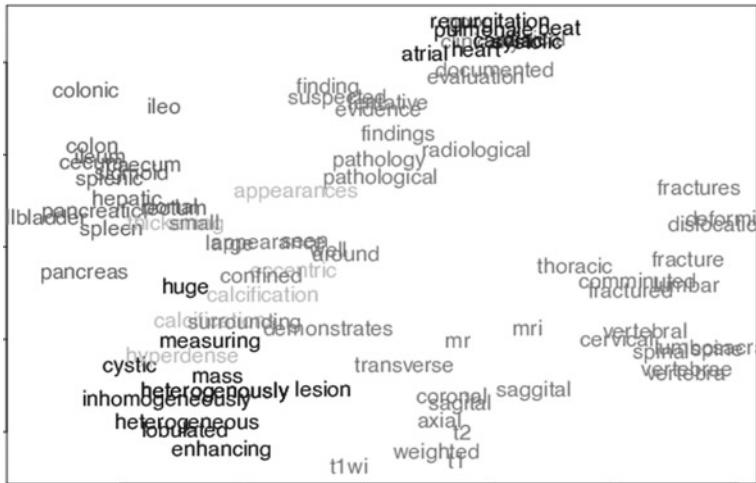


Fig. 17.3 Example words embedded in the vector space using word-to-vector modeling (visualized on 2D)

17.5.1 Removing Word-Level Ambiguity with Word-to-Vector Modeling

In radiology reports, there exist many recurring word morphisms in the text identification space, e.g., [mr, mri, t1–/t2-weighted²], [cyst, cystic, cysts], [tumor, tumour, tumors, metastasis, metastatic], etc. We train a deep word-to-vector model [15, 16, 39] to address this word-level labeling space ambiguities. Total ~ 1.2 billion words from all available our radiology reports and the biomedical research articles obtained from OpenI [40] are used. Therefore words with similar meaning are mapped or projected to closer locations in the vector space than dissimilar ones (i.e., locality-preserving mapping). Including texts from biomedical research articles resulted in a better word-to-vector model than using the radiology report text alone (semantically closer words were also closer in the vector space). Similar findings on unsupervised feature learning models, that robust features can be learned from a slightly noisy and diverse set of input, were reported by [41–43]. An example visualization of the word vectors on the 2D space using PCA is shown in Fig. 17.3.

Skip-gram model [15, 16] is employed with the mapping vector dimension of $\mathbb{R}^{1 \times 256}$ per word, trained using *hierarchical softmax* cost function, sliding-window size of 10 and frequent words sub-sampled in 0.01% frequency. It is found that combining an additional (more diverse) set of related documents, such as OpenI biomedical research articles, is helpful for the model to learn a better vector representation, withholding all the hyperparameters the same.

²Natural language expressions for imaging modalities of magnetic resonance imaging (MRI).

Table 17.4 Statistics of the word-lengths per sentences

# words/sentence	mean	median	std	max	min
Reports-wide	11.7	9	8.97	1014	1
Images references	23.22	19	16.99	221	4
Image references, no stopwords no digits	13.46	11	9.94	143	2
Image references, disease terms only	5.17	4	2.52	25	1

17.5.2 Using Sentences to Words Based Image Representation

Even the sentence referring a key image (and sentences before and after that) contain a variety of words, what we are mostly interested are the disease-related terms (which are highly correlated to diagnostic semantics). To obtain only the disease-related terms, we exploit the human disease terms and their synonyms from the Disease-Ontology (DO) [44] where 8,707 unique disease-related terms are collected. While the sentences referring an image and their surrounding sentences have 50.08 words on average, the number of disease-related terms in the three consecutive sentences is 5.17 on average with a standard deviation of 2.5. Therefore we chose to harness bi-gram language models to describe the images, achieving a good trade-off between the medium level complexity and not to miss out too many text-image pairs. A complete statistics about the number of words in the documents are shown in Table 17.4.

Each of the bi-gram disease terms are extracted so that we can train a deep CNN (to be described in the next section) to predict the bi-gram terms as a vector-level representation ($\mathbb{R}^{2 \times 256}$) describing the image. If multiple bi-grams can be extracted for a key image via its three content sentences, the image is trained as many times as the number of bi-grams with different target vectors $\{\mathbb{R}^{2 \times 256}\}$. If a disease term cannot form a bi-gram, then the term is ignored. This process is shown in Fig. 17.4. These bi-gram words of DO disease-related terms in the vector representation ($\mathbb{R}^{256 \times 2}$) are analogous to detecting multiple objects of interest and describing their spatial configurations in the image caption [8]. Deep regression CNN model is employed here to map a continuous output space from an image where the resulted bi-gram vector can be matched against a reference disease-related vocabulary, in the word-to-vector form using cosine similarity.

17.5.3 Bi-gram Deep CNN Regression

It has been shown [45] that deep recurrent neural network (RNN³) can learn the language representation for machine translation. To learn the image-to-text representation, we map the images to the vectors of word sequences describing the image.

³While RNN [46, 47] is one of the popular choices for learning language models [48, 49], deep convolutional neural network [3, 50] is more suitable for image classification.

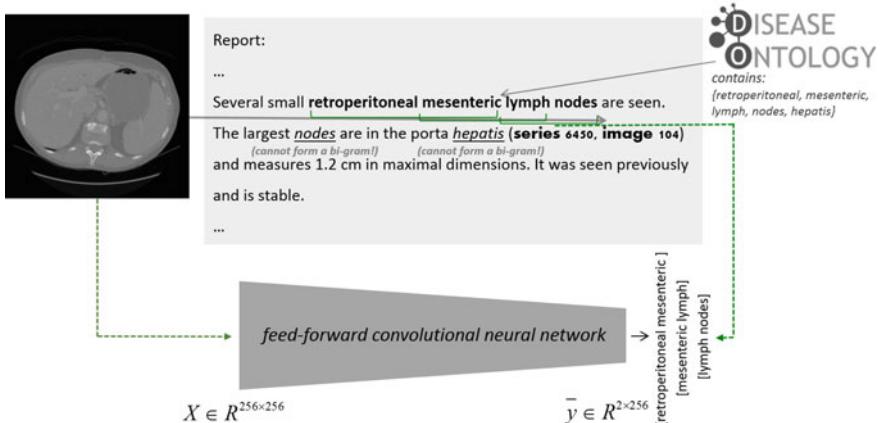


Fig. 17.4 An example of how word sequences are learned for an image. Bi-grams are selected from the image's referring sentences, containing disease-related terms of DO [44]. Each bi-gram is converted to a vector of $\mathbf{Z} \in \mathbb{R}^{256 \times 2}$ to learn from an image. Image input vectors as $\{\mathbf{X} \in \mathbb{R}^{256 \times 256}\}$ are learned through CNN via minimizing the cross-entropy loss between the target vector and output vector. The words “nodes” and “hepatis” in the second line are DO terms but ignored since they cannot form a bi-gram

This can be formulated as a regression CNN, replacing the softmax cost in Sect. 17.4 with the cross-entropy cost function for the last output layer of VGG-19 CNN model [4]:

$$E = -\frac{1}{n} \sum_{n=1}^N [g(\mathbf{z})_n \hat{g}(\bar{\mathbf{z}}_n) + (1 - g(\mathbf{z}_n)) \log(1 - g(\hat{\mathbf{z}}_n))], \quad (17.2)$$

where \mathbf{z}_n or $\hat{\mathbf{z}}_n$ is any uni-element of the target word vectors \mathbf{Z}_n or optimized output vectors $\hat{\mathbf{Z}}_n$. Sigmoid function is $g(x) = 1/(1 + e^x)$ and n is the number of samples in the database. We adopt the CNN model of [4] since it works consistently better than [3] in our image categorization task. Caffe [12] deep learning framework is employed. The CNN network is fine-tuned from the previous model on predicting the topic-level labels in Sect. 17.4, except for the last output layer. The newly modified output layer has 512 nodes for bi-grams as 256×2 (twice of the dimension \mathbb{R} of the word vectors). The cross-entropy cost decreases and converges during training in about 10,000 iterations since only fine-tuning is needed.

17.5.4 Word Prediction from Images as Retrieval

For any key image in testing, first we predict its categorization topic labels of each hierarchy level (document-level, document-level-h2, sentence-level) using three deep CNN models [4] in Sect. 17.4. Each LDA cluster keeps top 50 keywords

from document-clustering. All 50 keywords in each cluster of multi-level topics are mapped into the word-to-vector space as multivariate variables \mathbb{R}^{256} in Sect. 17.5.1. Thus we obtain 3 sets of 50 \mathbb{R}^{256} reference word vectors per testing image. Second, the image is mapped using the learned bi-gram CNN model in Sect. 17.5.3 to have a $\mathbb{R}^{256 \times 2}$ output vector. Third, in each set, we match each of the 50 \mathbb{R}^{256} reference vectors against the first and second half of the $\mathbb{R}^{256 \times 2}$ output vector (i.e., treated as two words in the word-to-word matching) where cosine similarity is used. The

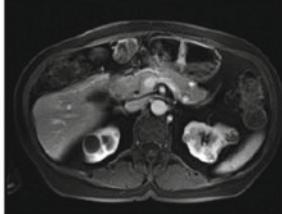
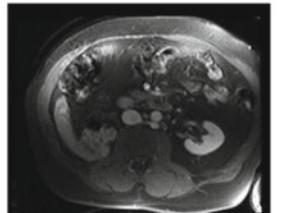
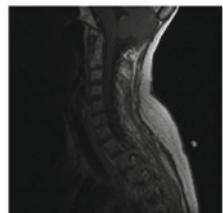
<i>input image</i>	<i>predicted key-words</i>	<i>original report</i>
	diameter mass kidney avg distance: 0.33	“... and solid lobulated mass arises from the anterior lower pole of right kidney and measures 1.6 cm in diameter ...”
	adenopathy masses lung avg distance: 0.21	“... dozens of masses of various sizes in or near right pleura and or peripheral lung without definite change ... by areas of right lung consolidation atelectasis and or confluent masses ...”
	diameter lesion kidney avg distance: 0.33	“... 2 apparently cystic lesion in the retroperitoneum adjacent to the crus 3 liver lesions for example series 17 image 22 series 16 image 172 and image 137 the lateral lesion ...”
	spine chest scoliosis avg distance: 0.42	“... it measures a few mm in diameter and is best appreciated on series 3 image 6 in the thoracic spine no definite enhancing lesions are present. in the lumbosacral spine ...”

Fig. 17.5 Illustrative examples of text generation results. Kidney appears in the third row image but its report does not mention this

closest keywords at three hierarchy levels (with the highest cosine similarity against either of the bi-gram words) are kept per image. The average count of “hits” (i.e., the generated keywords exactly appeared in the referencing sentences of the image) is 0.56. Text generation examples are shown in Fig. 17.5, with three keywords from three categorization levels per image. We are not focusing on generating naturally expressed sentences yet [8, 29] but predicting key words with diagnostic indications. Our “hits” quantitative evaluation can be done precisely and no human based subjective rating is needed. Generating caption-like natural language sentences will be studied for future work.

17.6 Conclusion and Discussion

It has been inherently unclear how to extend the significant success in image classification using deep convolutional neural networks, from computer vision to medical imaging. What are clinically relevant image labels to be defined, how to annotate the huge amount of medical images required by deep learning models, and to which extend and scale deep CNN is generalizable in medical image analysis are the open questions.

In this chapter, we present an interleaved text/image deep mining system to extract the semantic interactions of radiology reports and diagnostic key images at a very large unprecedented scale in the medical domain. Images are classified into different levels of topics according to their associated documents, and a neural language model is learned to assign field-specific (disease) terms to predict what is in the image. We demonstrate promising quantitative and qualitative results, suggesting a way to extend the deep image classification systems to learning medical imaging informatics from “big data” at a modern hospital scale. Although the image categorization labels extracted from unsupervised document topic models show strong correlations with human understanding, further clinician review and verification will be needed (especially for finer-level clusters/labels in the hierarchy).

Predicting words for key images by CNN regression shows good feasibility. More sophisticated NLP parsing on description sentences is needed [51]. The most relevant describing words are the appeared objects (anatomies and pathologies) [52] and their attributes [24, 25]. The NLP-based anatomy/pathology assertion and negation using describing sentences can generate reliable labels to predict. Then the weakly supervised object detection and segmentation using deep models [53–55] become critical tasks to solve under this setting (i.e., no object bounding-box given). Understanding high-level concepts from text and improving large-scale image-sentence embeddings under weakly supervised annotations [56, 57] should also be developed.

References

1. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: Computer vision and pattern recognition. IEEE, pp 248–255
2. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M et al (2014) Imagenet large scale visual recognition challenge. [arXiv:1409.0575](https://arxiv.org/abs/1409.0575)
3. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
4. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
5. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2014) Going deeper with convolutions. [arXiv:1409.4842](https://arxiv.org/abs/1409.4842)
6. Ordonez V, Deng J, Choi Y, Berg A, Berg T (2013) From large scale image categorization to entry-level categories. In: ICCV
7. Babenko A, Slesarev A, Chigorin A, Lempitsky V (2014) Neural codes for image retrieval. In: ECCV
8. Kulkarni G, Premraj V, Ordonez V, Dhar S, Li S, Choi Y, Berg A, Berg T (2013) Babytalk: understanding and generating simple image descriptions. IEEE Trans Pattern Anal Mach Intell 35(12):2891–2903
9. Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022
10. Frome A, Corrado G, Shlens J, Bengio S, Dean J, Ranzato M, Mikolov T (2013) Devise: a deep visual-semantic embedding model. In: NIPS, pp 2121–2129
11. Kiros R, Szepesvri C (2012) Deep representations and codes for image auto-annotation. In: NIPS, pp 917–925
12. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: convolutional architecture for fast feature embedding. [arXiv:1408.5093](https://arxiv.org/abs/1408.5093)
13. Gupta S, Girshick R, Arbelz P, Malik J (2014) Learning rich features from RGB-D images for object detection and segmentation. In: ECCV
14. Gupta A, Ayhan M, Maida A (2013) Natural image bases to represent neuroimaging data. In: ICML
15. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. [arXiv:1301.3781](https://arxiv.org/abs/1301.3781)
16. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems, pp 3111–3119
17. Ganin Y, Lempitsky V (2014) N4-fields: neural network nearest neighbor fields for image transforms. CoRR. [arXiv:1406.6558](https://arxiv.org/abs/1406.6558)
18. Deselaers T, Ney H (2008) Deformations, patches, and discriminative models for automatic annotation of medical radiographs. PRL 29:2003
19. Carrivick L, Prabhu S, Goddard P, Rossiter J (2005) Unsupervised learning in radiology using novel latent variable models. In: CVPR
20. Barnard K, Duygulu P, Forsyth D, Freitas N, Blei D, Jordan M (2003) Matching words and pictures. JMRL 3:1107–1135
21. Blei D, Jordan M (2003) Modeling annotated data. In: ACM SIGIR
22. Socher R, Ganjoo M, Manning CD, Ng A (2013) Zero-shot learning through cross-modal transfer. In: Advances in neural information processing systems, pp 935–943
23. Krizhevsky A, Hinton G (2009) Learning multiple layers of features from tiny images. Computer Science Department, University of Toronto, Technical report
24. Lampert CH, Nickisch H, Harmeling S (2014) Attribute-based classification for zero-shot visual object categorization. IEEE Trans Pattern Anal Mach Intell 36(3):453–465
25. Scheirer W, Kumar N, Belhumeur P, Boult T (2012) Multi-attribute spaces: calibration for attribute fusion and similarity search. In: CVPR
26. Lampert CH, Nickisch H, Harmeling S (2009) Learning to detect unseen object classes by between-class attribute transfer. In: CVPR, pp 951–958

27. Rashtchian C, Young P, Hodosh M, Hockenmaier J (2010) Collecting image annotations using amazon's mechanical turk. In: Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with amazon's mechanical turk. Association for Computational Linguistics, pp 139–147
28. Jaderberg M, Vedaldi A, Zisserman A (2014) Deep features for text spotting. In: ECCV, pp 512–528
29. Ordonez V, Kulkarni G, Berg TL (2011) Im2text: describing images using 1 million captioned photographs. In: Advances in neural information processing systems, pp 1143–1151
30. Hofmann T (1999) Probabilistic latent semantic indexing. In: Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval. ACM, pp 50–57
31. Lee DD, Seung HS (1999) Learning the parts of objects by non-negative matrix factorization. *Nature* 401(6755):788–791
32. Stevens K, Kegelmeyer P, Andrzejewski D, Buttler D (2012) Exploring topic coherence over many models and many topics. In: Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning. Association for Computational Linguistics, pp 952–961
33. Girolami M, Kabán A (2003) On an equivalence between PLSI and LDA. In: Proceedings of the 26th annual international ACM SIGIR conference on research and development in informaion retrieval. ACM, pp 433–434
34. Ding C, Li T, Peng W (2006) Nonnegative matrix factorization and probabilistic latent semantic indexing: equivalence chi-square statistic, and a hybrid method. In: Proceedings of the national conference on artificial intelligence, vol 21. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, p 342
35. Gaussier E, Goutte C (2005) Relation between PLSA and NMF and implications. In: Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval. ACM, pp 601–602
36. Ramage D, Rosen E (2011) Stanford topic modeling toolbox. <http://www-nlp.stanford.edu/software/tmt>
37. Kiapour H, Yamaguchi K, Berg A, Berg T (2014) Hipster wars: discovering elements of fashion styles. In: ECCV
38. Ordonez V, Berg T (2014) Learning high-level judgments of urban perception. In: ECCV
39. Mikolov T, Yih WT, Zweig G (2013) Linguistic regularities in continuous space word representations. In: HLT-NAACL, pp 746–751 (Citeseer)
40. Openi - an open access biomedical image search engine. <http://openi.nlm.nih.gov>. Lister Hill National Center for Biomedical Communications, U.S. National Library of Medicine
41. Shin HC, Orton MR, Collins DJ, Doran SJ, Leach MO (2013) Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data. *IEEE Trans Pattern Anal Mach Intell* 35(8):1930–1943
42. Vincent P, Larochelle H, Bengio Y, Manzagol PA (2008) Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning. ACM, pp 1096–1103
43. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J Mach Learn Res* 11:3371–3408
44. Schriml LM, Arze C, Nadendla S, Chang YWW, Mazaitis M, Felix V, Feng G, Kibbe WA (2012) Disease ontology: a backbone for disease semantic integration. *Nucleic Acids Res* 40(D1):D940–D946
45. Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. In: Advances in neural information processing systems
46. Rumelhart DE, Hinton GE, Williams RJ (1988) Learning representations by back-propagating errors. *Cognitive modeling*
47. Werbos PJ (1990) Backpropagation through time: what it does and how to do it. *Proc IEEE* 78(10):1550–1560

48. Bengio Y, Schwenk H, Senécal JS, Morin F, Gauvain JL (2006) Neural probabilistic language models. *Innovations in machine learning*. Springer, Berlin, pp 137–186
49. Mikolov T, Karafiat M, Burget L, Cernocky J, Khudanpur S (2010) Recurrent neural network based language model. In: *INTERSPEECH*, pp 1045–1048
50. LeCun Y, Huang FJ, Bottou L (2004) Learning methods for generic object recognition with invariance to pose and lighting. In: *Proceedings of the 2004 IEEE computer society conference on computer vision and pattern recognition, CVPR 2004*, vol 2. IEEE, pp II–97
51. Li S, Kulkarni G, Berg T, Berg A, Choi Y (2011) Composing simple image descriptions using web-scale n-grams. In: *ACM CoNLL*, pp 220–228
52. Mitchell M, Han X, Dodge J, Mensch A, Goyal A, Berg A, Yamaguchi K, Berg T, Stratos K, Daume H (2012) Midge: generating image descriptions from computer vision detections. In: *EACL*, pp 747–756
53. Mittelman R, Lee H, Kuipers B, Savarese S (2013) Weakly supervised learning of mid-level features with beta-Bernoulli process restricted Boltzmann machines. In: *CVPR*
54. Oquab M, Bottou L, Laptev I, Sivic J (2014) Weakly supervised object recognition with convolutional neural networks. Technical report. HAL-01015140, INRIA
55. Pinheiro P, Collobert R (2014) Weakly supervised object segmentation with convolutional neural networks. Technical report. Idiap-RR-13-2014, Idiap
56. Berg A, Berg T, Daume H, Dodge J, Goyal A, Han X, Mensch A, Mitchell M, Sood A, Stratos K, Yamaguchi K (2012) Understanding and predicting importance in images. In: *CVPR*
57. Gong Y, Wang L, Hodosh M, Hockenmaier J, Lazebnik S (2014) Improving image-sentence embeddings using large weakly annotated photo collections. In: *ECCV*

Author Index

B

Baumgartner, Christian F., 159
Bradley, Andrew P., 225

C

Carneiro, Gustavo, 11, 197, 225
Cherry, Kevin, 35
Comaniciu, Dorin, 49, 241

D

Dhungel, Neeraj, 225

F

Farag, Amal, 279

G

Gao, Mingchen, 97, 113
Georgescu, Bogdan, 49, 241
Gotway, Michael B., 181
Gurudu, Suryakanth R., 181

H

Huang, Junzhou, 137
Hurst, R. Todd, 181

K

Kendall, Christopher B., 181
Kim, Lauren, 35, 317
Kong, Xiangfei, 257

L

Lang, Bing, 73
Liang, Jianming, 181
Liu, David, 49, 241
Liu, Fujun, 63
Liu, Jiamin, 35, 279
Liu, Qingshan, 73
Lu, Le, 35, 113, 279, 317

M

Mollura, Daniel J., 97, 113

N

Ngo, Tuan Anh, 197
Nguyen, Hien, 49
Nogues, Isabella, 113

O

Oktay, Ozan, 159

R

Roth, Holger R., 35, 113, 279
Rueckert, Daniel, 159

S

Seff, Ari, 35, 317
Shin, Hoo-Chang, 113, 317
Shin, Jae Y., 181

Su, Hai, [257](#)

Summers, Ronald M., [3](#), [35](#), [113](#), [279](#), [317](#)

Xu, Zheng, [137](#)

Xu, Ziyue, [97](#), [113](#)

T

Tajbakhsh, Nima, [181](#)

Turkbey, Evrim, [279](#)

Y

Yang, Lin, [11](#), [63](#), [257](#)

Yao, Jianhua, [35](#), [113](#), [317](#)

X

Xie, Yuanpu, [257](#)

Xing, Fuyong, [11](#), [257](#)

Xu, Daguang, [241](#)

Xu, Jun, [73](#)

Z

Zhang, Shaoting, [257](#)

Zheng, Yefeng, [11](#), [49](#), [241](#)

Zhou, Chao, [73](#)

Subject Index

A

Abdomen detection, 248, 249
Automated image interpretation, 319

C

Cancer imaging, 87
Cardiac Cine Magnetic Resonance, 198
Cardiac magnetic resonance, 161, 162, 165
Cardiovascular and microscopy image analysis, 11
Carotid artery bifurcation detection, 49, 57, 58
Carotid intima-media thickness, 189, 191
Cascaded random forest, 279, 281, 290, 292, 299
Cascaded superpixel parsing, 279, 281, 282, 290, 292, 299
Cell detection, 63–65, 68–71, 137–146, 152, 154, 257–260, 262, 276
Cell segmentation, 257–259, 262, 264
CNN architectures, 113–117, 119–123, 129–131
Computational Pathology, 137
Computer-aided detection, 190
Computer-aided diagnosis (CAD), 3–5, 12, 19, 113, 115, 130, 132
Convolutional neural network, 97–105, 107–109, 181, 182, 247, 317, 323, 330

D

Dataset characteristics, 132
Deep convolutional neural network, 142, 146, 154

Deep convolutional neural networks, 35, 36, 113, 114, 132, 289, 297, 299

Deep learning, 3, 5–7, 11, 13–26, 36, 73, 75, 76, 78, 114, 119, 127, 130, 162, 187, 200, 227, 229, 238, 259, 318, 320, 321, 328, 330

Denoising auto-encoder, 56

Dense conditional random field, 99

Dense image patch labeling, 283, 299

Digital pathology, 73–75, 258

F

Fetal ultrasound imaging, 160, 168, 170
Fine-tuning, 181–189, 191
Fisher Vector Encoding, 105
Fully automated CT interpretation, 299
Fully convolutional neural networks, 159–161, 177

G

Global image context, 241, 242

I

Image recognition, 159, 160
Image super-resolution, 161
Interleaved Text/Image Mining, 317, 318, 330
Interstitial lung disease (ILD), 97–100, 102–110

K

Kidney detection, 241, 243, 248, 252, 254
Kidney segmentation, 241–243, 250, 253, 254

L

- Label propagation, 101, 106, 109
 Landmark detection, 50–52, 56–59
 Lesion detection, 190
 Level set method, 197, 199, 202, 222
 Local image context, 252, 254
 Lung cancer, 137, 144–148, 154
 Lung segmentation, 198, 199, 210, 212, 213,
 215, 216, 222

M

- Mammogram segmentation, 226, 227, 229
 Mammography, 11–14
 Medical diagnostic imaging, 35
 Medical image analysis, 76, 190
 Microscopic image analysis, 258
 Multi-label deep regression, 99

N

- Natural Language Processing, 318, 320
 Network sparsification, 49, 51, 59

O

- Organ localisation, 159, 160

P

- Pancreas segmentation, 279–281, 290, 292,
 293, 295–300

Patch-level confidence pooling, 281

- Polyp detection, 182, 183, 186, 190, 191
 Pulmonary embolism detection, 182, 190,
 191

R

- Random view aggregation, 40

S

- Scan plane detection, 161, 170, 177
 Segmentation of the Left Ventricle of the
 Heart, 198–200, 210
 Separable filter decomposition, 49, 55, 59
 Sparse kernel, 137, 150, 152, 154
 Structured output learning, 225, 227, 228

T

- Topic Models, 321, 322, 324, 325, 330
 Transfer learning, 113, 115–117, 119, 121,
 122, 124, 125, 128, 130–132, 191

U

- Unordered pooling, 105, 108, 109

V

- Video quality assessment, 182