Towards High Performing and Reliable Deep Convolutional Neural Network Models for Typically Limited Medical Imaging Datasets

by

Kaoutar Ben Ahmed

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Computer Science and Engineering
College of Engineering
University of South Florida

Co-Major Professor: Lawrence O. Hall, Ph.D.
Co-Major Professor: Dmitry B. Goldgof, Ph.D.
Shaun Canavan, Ph.D.
Ashwin Parthasarathy, Ph.D.
Robert Gatenby, M.D.

Date of Approval:
October 13, 2022

Keywords: CNN, Ensemble learning, Medical image analysis, Generalization, Shortcut learning

Copyright © 2022, Kaoutar Ben Ahmed

Dedication

I dedicate this work to each member of my family who believed and encouraged me. I deeply thank my brother, Dr. Mohamed Ben Ahmed, for his invaluable support. Thank you for always encouraging me to pursue my goals. Your hard work and perseverance have always inspired me to do my best.

Thank you Mom for always keeping me in your thoughts and prayers.

I dedicate this accomplishment to my husband and 4-year-old daughter, Lilia. I hope I've made you proud.

Acknowledgments

I'm very grateful to my supervisor Dr. Lawrence Hall for his continuous help and support during my Ph.D. study. His high standards and professionalism will forever inspire me. I'm also very thankful to my co-supervisor Dr. Goldgof for his dedicated guidance and mentorship. It has been an honor to join their research group and work under their supervision.

I wish to thank my committee members Dr. Robert Gatenby and Dr. Ashwin Parthasarathy who were generous with their expertise and valuable time. A special thanks to Dr. Shaun Canavan for allowing me to participate in his research project which enabled me to witness the application of my research to other domains.

I'm grateful to our department's staff members for their timely assistance and prompt responses.

I gratefully acknowledge the American Association of University Women for selecting me to receive the International Doctoral Fellowship. It's a great honor for me to receive this prestigious award.

I want to extend my gratitude to all my research group members and my co-authors for motivating me with their good work and dedication.

Table of Contents

List of Tables

List of Figures

Abstract

Artificial Intelligence (AI) is "The science and engineering of making intelligent machines, especially intelligent computer programs" [93]. Artificial Intelligence has been applied in a wide range of fields including automobiles, space, robotics, and healthcare.

According to recent reports, AI will have a huge impact on increasing the world economy by 2030 and it's expected that the greatest impact will be in the field of healthcare. The global market size of AI in healthcare was estimated at USD 10.4 billion in 2021 and is expected to grow at a high rate from 2022 to 2030 (CAGR of 38.4%) [124]. Applications of AI in healthcare include robot-assisted surgery, disease detection, health monitoring, and automatic medical image analysis. Healthcare organizations are becoming increasingly interested in how artificial intelligence can support better patient care while reducing costs and improving efficiencies.

Deep learning is a subset of AI that is becoming transformative for healthcare. Deep learning offers fast and accurate data analysis. Deep learning is based on the concept of artificial neural networks to solve complex problems.

In this dissertation, we propose deep learning-based solutions to the problems of limited medical imaging in two clinical contexts: brain tumor prognosis and COVID-19 diagnosis. For brain tumor prognosis, we suggest novel systems for overall survival prediction of Glioblastoma patients from small magnetic resonance imaging (MRI) datasets based on ensembles of convolutional neural networks (CNNs). For COVID-19 diagnosis, we reveal one critical problem with CNN-based approaches for predicting COVID-19 from chest X-ray (CXR) imaging: shortcut learning. Then, we experimentally suggest methods to mitigate this problem to build fair, reliable, robust, and transparent deep learning based clinical decision support systems. We discovered this problem with CNNs and using Chest Xray

imaging. However, the issue and solutions generally apply to other imaging modalities and recognition problems.

Chapter 1: Introduction

1.1 Deep Learning

Machine learning (ML) is a subset of Artificial Intelligence that focuses on enabling machines to make decisions by feeding them data. Deep learning (DL) is a subset of machine learning that uses mathematical models designed to loosely imitate the human brain. Deep learning can process large amounts of data at an astonishing speed without compromising accuracy.

The secret to deep learning's success is in the 'learning' concept. Deep learning models can learn from previous mistakes to improve their future results. Deep learning is based upon the artificial neural network (ANN). It's a loose imitation of human brain neural networks. The process starts when we feed a set of inputs to the ANN then the input is processed via a cascade of layers (hidden layers) of neurons (nodes) to produce the desired output. Each succeeding layer of nodes is activated when it obtains stimuli from the previous layer. Consequently, the hidden layers turn raw input into meaningful output.

With the composition of enough data transformations across the network's layers, very complex functions can be learned. Deep learning learns to express complex concepts in terms of new, simpler ones. For instance, deep learning models can represent the concept of an image via a collection of simpler concepts, such as corners and contours, which may be reduced to edges; as well as textures.

The basic example of a deep learning model is the Multilayer Perceptron (MLP). It consists of three types of layers, the input, the hidden, and the output layer. The data flow in the forward direction from the input to the output layer, which performs the desired task (prediction/classification). Multilayer Perceptrons are called universal approximators, they

can supposedly approximate any continuous function and are capable of solving non linear problems.

When using Deep learning, the time necessary for data pre-processing can be saved. Neural networks can normalize and extract features from raw data, which reduces the time and manual intervention.

1.2 Why Deep Learning for Medical Image Analysis

Medical professionals are increasingly recognizing the value that deep learning technology brings to the table. Deep learning is offering solutions that are changing the shape of healthcare. Many studies have shown that deep learning could benefit patients and health systems in clinical practice by delivering personalized and relevant information for patient care.

Deep learning can be used for diagnosis from medical imaging with high accuracy and speed. Deep learning has many breakthrough applications in healthcare, particularly, in medical image analysis. Scientists [34] used deep learning models to detect diabetic retinopathy, a leading cause of blindness. The trained deep neural network performs comparably to human specialists in doing diagnosis (AUC was 0.939 for retinal fundus photographs (RFP)). For medical imaging, there is a shortage of radiologists and MRI technicians, especially in low-to-middle-income countries. Deep learning can be considered an effective approach to a solution.

Deep learning tools can deliver very high diagnostic accuracy from medical imaging. It can deliver high-performing results comparable to human experts or sometimes surpass experts. In 2018, a paper [51] showed that a convolutional neural network trained for dermoscopic melanoma recognition can outperform a big group of 58 dermatologists with 30 experts among them. The trained CNN-based model was 7% more accurate than the dermatologists even when they had access to some extra clinical information (age, sex, etc.).

Similarly, a study [95] proposed a deep learning model capable of surpassing human specialists in breast cancer prediction from mammograms. The deep learnt model outperformed all of the radiologists, the AUC of the model was greater than the AUC of the average radiologist by an absolute margin of 0.115.

Additionally, researchers in [84] proposed a deep learning model capable of classifying 26 skin conditions using skin images and medical history data. The deep learnt model scored a top-1 accuracy of 0.66 outperforming 6 dermatologists and six primary care physicians and six nurse practitioners who scored top-1 accuracies of 0.63, 0.44, and 0.40 respectively).

Rapid response is critical in healthcare and may lead to improved patient outcomes. Authors in [139] proposed a fast and accurate deep learning model to perform diagnosis of critical neurological conditions like stroke and brain hemorrhage from head CT images. The model's prediction speed was 1.2 seconds, which is 150 times faster than human experts.

1.3 Learning from Small/Limited Medical Data

Despite deep learning's promising results in improving clinicians' diagnoses, forecasting the spread of diseases, and customizing treatments, healthcare data access is considered to be the biggest deep learning bottleneck. Access to high-quality big data in a standardized format is one of the major obstacle standing in the way of the adoption of AI-powered applications in healthcare. Without large, high-quality data sets, it is difficult to build useful deep learning solutions.

Today, data is highly dispersed. Merging data in one central location is highly challenging. Additionally, medical data has sensitive data and patients are not willing to share their intimate health records for fear of data theft.

AI technologies are waiting for health care providers to increase the availability of data. However, making changes will not be easy. A recent Mckinsey report [91] found that roughly one-quarter of the hospitals in the United States have not yet embraced the use of electronic health record systems. Therefore, even though there is interest in adopting deep learning

applications in healthcare, the adoption rate is slow due to the sensitive nature of medical data and regulatory barriers. Thus, faster alternatives are needed.

1.4 Learning from Biased/Noisy Medical Data

The overwhelming complexity of healthcare data is simultaneously the biggest advantage and most intractable challenge of adopting deep learning solutions in healthcare.

Electronic Healthcare Record (EHR) systems are largely not compatible across providers resulting in data collection that is localized rather than centralized. Medical imaging data is often obtained with distinctive imaging settings and protocols across hospitals, regions, and countries. While changes in image acquisition may not confound a human specialist, it often prevents high-quality deep learning models from being learned.

Having access to large amounts of medical data is not enough for developing meaningful deep learning solutions. Adequate curation, analysis, and annotation are critical.

Sampling bias is a major challenge faced by models trained with data from a single institution or multiple institutions with small population diversity. The decisions of the trained model may be unreliable due to differences between the sample population and the target population. Examples of bias sources include proportions of race and sex, imaging machines (vendors, types, acquisition protocols), and the prevalence of diseases, in addition to variations in imaging protocols and local practice.

1.5 Contributions

This dissertation discusses the critical challenges of developing AI-powered medical image analysis solutions. Limited medical imaging data is the main hurdle blocking and slowing down the process of adoption of deep learning-based solutions in healthcare. Medical data is limited not only in size, but it suffers from a sampling bias due to variations in imaging protocols, local practice, and other aspects which will be addressed in this work.

Convolutional neural networks require large amounts of data to train. To overcome the limited availability of medical image datasets, we suggest the use of transfer learning from a CNN pre-trained on ImageNet, a large-scale visual imaging dataset, and fine-tuned on a small set of MR images to predict the survival time of Glioblastoma patients. Our results demonstrate that an accuracy of 90.91% can be achieved using one single feature extracted from pre-treatment MR images. Also, we show that fine-tuned CNNs offer good performance without the need for additional feature extraction methods. Additionally, the impact of different sequences of MR images on classification performance was explored. Moreover, we experimented with deep features from pre and post-treatment imaging.

A novel ensemble training and evaluation of small CNNs was proposed and shown to effectively predict survival time even with a small dataset, limited sequences, and a variable set of acquisitions. An accuracy of 72.06% was obtained outperforming pre-trained models on ImageNet which only scored a predictive accuracy of 66.18%.

We suggest the effective use of snapshot ensembles in a novel fashion as a solution to the limited availability of labeled medical imaging datasets. We demonstrated an overall survival prediction for glioblastoma patients of 74% accuracy using a multi-branch 3D convolutional neural network, where each branch had a different MR image sequence as input.

On the other hand, we demonstrate that CNN-based disease diagnosis models can leverage data-source specific and unintended/undesired shortcuts when making predictions. Due to the black box nature of deep learning models, it is difficult to confirm what the model is basing its decisions upon. Shortcuts can be confounding and are spurious features that a model finds easy to rely on to make decisions rather than on generally discriminators . Those learned shortcuts might make the model perform highly under familiar settings that it had seen before but it usually fails under slightly different circumstances. Implementing a shortcut-based disease diagnosis solution in healthcare could lead to false diagnoses, especially when applied to data from a new source.

We point out shortcuts to look out for while training models. We then suggest steps for training data curation and preprocessing to mitigate shortcut learning. We experimentally demonstrate that the use of region-of-interest segmentation, histogram equalization, and noise-based augmentation can reduce the gap in performance between cross-validation results and unseen source predictions.

The proposed solutions here can apply to other imaging modalities and other recognition problems.

1.6 Organization

The remainder of this dissertation is organized as follows:

- Chapter 2 introduces the background concepts used in this work including transfer learning, ensemble learning, and shortcut learning.
- Chapter 3 surveys some related work about the application of deep learning to Glioblastoma prognosis and COVID-19 diagnosis.
- Chapter 4 provides a detailed description of all the datasets used and mentioned in this work.
- Chapter 5 presents the proposed survival analysis method from small medical imaging sets using transfer learning and fine-tuning.
- Chapter 6 suggests a novel ensemble training and evaluation for convolutional neural networks for limited-size medical datasets.
- Chapter 7 combines snapshot learning and ensemble learning to leverage multi-modal imaging data for higher model performance.
- Chapter 8 discusses a critical discovery of the shortcut learning problem in CNN-based COVID-19 diagnosis systems.

- Chapter 9 proposes a human-in-the-Loop based solution to mitigate shortcut learning for reliable AI-powered clinical decision support systems.
- Chapter 10 concludes and lists some limitations.
- Chapter 11 discusses some possible future research directions to advance this work.
- Chapter 12 lists the publications that resulted from this work.

Chapter 2: Background

2.1 Convolutional Neural Networks

The amount of visual data has exploded in the last few years. Cameras are available everywhere in the world producing a large amount of visual data each day. The majority of bits transferred through the Internet is visual data. Therefore, it's critical to develop algorithms that can utilize, understand, and analyze this data. Computer vision is the sub-field of artificial intelligence which attempts to replicate human vision. It enables computers to identify and "see" visual data (e.g. images, videos). Computer vision is an interdisciplinary field that touches on many different areas such as engineering, computer science, mathematics, biology, and others.

Convolutional neural networks are very popular for computer vision tasks like image recognition, object localization and detection, face recognition, semantic segmentation and others. A Convolutional neural network (CNN) is an artificial neural network that can detect patterns in visual data and make sense of them. Patterns in images can be edges, shapes, texture, etc. CNNs have a special type of hidden layers called convolutional layers which perform convolutional operations to transform inputs and send them to the following layer as output. Convolutional operations work by applying a filter over the input image to generate what's called a feature map. The filter is also called a feature extractor, it helps the model to focus on specific key features and capture spatial information in the original input. Pooling layers are another special type of layers that CNNs have which allow the downsampling of a particular feature map to make processing faster.

Convolutional neural networks are most often used for image processing problems but they can be applied to solve other problems such as natural language processing [146]. CNNs

offer the advantage of automatic feature extraction from imaging data. Thus, not requiring manual feature extraction. Additionally, a CNN does not necessary require the segmentation of the region of interest (ROI) by human experts. CNNs have shown success in multiple tasks in medical image analysis namely segmentation. CNN-based architectures like U-net [126], V-net [98], and Fully Convolutional Networks FCNs [156] have enabled precise localisation and exact region segmentation of medical images. In medical image analysis, convolutional neural networks are being used for disease diagnosis and prognosis. For example, CNNs are frequently employed to classify brain tumors and lung nodules into two or more categories.

2.2 Transfer Learning

Convolutional neural networks have many parameters. Thus, requiring large amount of labeled data to train. However, large-scale medical imaging datasets are rare due to the time and effort needed for labeling by human experts. To solve this dilemma, methods like transfer learning can be used.

Transfer learning is the process of using an existing model's architecture and weights pre-trained to solve one problem to solve a different, but related problem. The pre-trained model is then further trained (fine-tuned) on the destination task of interest. This method gives much faster results with a typically slightly lower error rate in comparison to training from scratch. Transfer learning has been widely adopted in various medical imaging applications. Obviously, there are significantly more labeled natural image datasets publicly available than medical image datasets. ImageNet is a natural image dataset containing more than 14 million images. Scientists have been studying the potential of transfer learning from models trained on ImageNet to perform medical imaging analysis tasks. This approach was used to solve medical imaging problems such as the detection of brain abnormality from MRI scans [8, 37], heart (cardiac) image analysis [92, 100], diagnosis of chest disorders from chest radiographs [110, 31], breast lesion classification [50, 140], skin cancer diagnosis [71, 89], Retinal disease detection from fundoscopy images [153, 80], prostate cancer diagnosis [41, 86], and others.

2.3 Ensemble Learning

Recent studies show that the most successful medical image analysis approaches proposed in the literature are based on a technique called ensemble learning. Ensemble learning is a method based on merging several learning models to achieve better predictive performance than could be obtained individually. Ensemble learning in the context of deep learning is a training method that can reduce the variance of predictions [54] and lead to achieving higher accuracy. It consists of combining multiple deep learning models concurrently to perform the intended prediction (parallel/cascade). There exist multiple techniques to join the individual predictions from each sub-network such as majority voting, weighted voting, and averaging. To implement a majority voting approach, an odd number of sub-models is used. We then send the inputs to each sub-model in parallel and output a prediction. Each model participates in prediction voting. The prediction with the most votes wins and it is the test output for that particular sample. Finally, we average all test outcomes as the system's Final prediction outcome. In medical imaging, studies used ensemble learning in its various forms ranging from the combination of multiple models [121, 21], to inference improvement of a one model [47], to efficient limited training data usage [68, 106].

2.4 Snapshot Learning

In some particular cases, a single model can take a significantly long time to train. In such a case, saving multiple snapshots of the model periodically can be an alternative. This approach is called snapshot learning [62]. A snapshot is the set of weights in the network at a particular epoch. This method found to be faster and often performs as well as training a small number of models, at no additional training cost. Effective ensembles require a diverse set of models having different distributions of prediction error. One way of insuring diversity is by using aggressive learning rate causing large changes in the model's weights. Cosine Annealing is a type of aggressive learning rate schedule based on starting at a high learning

rate value and then dropping it quickly to a minimum value near zero before raising it again to the initial high value.

2.5 Shortcut Learning

Shortcut learning is when a deep learning model finds and follows a "shortcut" strategy to achieve excellent results under familiar circumstances but fails under slightly different settings. For instance, a model can correctly identify and classify Cows in familiar contexts like grass fields, but it fails to detect a cow in unfamiliar backgrounds (beach, waves and boat) [18]. Shortcuts are spurious features that the model relies on to take decisions. Shortcut-based models can perform well on data from previously seen sources but poorly on data from never seen sources [48]. That is, the model exploits only a few predictive features and consequently suffers from generalization failures. To make sure that a model is using intended features for decision making, it has to work well not only on internal test sets, but also perform as intended on out-of-distribution test sets (such as you get from unseen data sources).

Chapter 3: Related Work

3.1 Deep Learning for Glioblastoma Prognosis from MRIs

A glioma is the most common primary tumor of the brain. It originates from glial cells. About 80% of malignant brain tumors are gliomas. According to the 2021 WHO Classification of Tumors of the Central Nervous System [85], glioblastoma is a sub-category of adult-type diffuse gliomas. It is considered a WHO grade IV glioma which represents approximately 57% of all gliomas and 48% of all primary malignant central nervous system (CNS) tumors [112]. Glioblastoma is highly invasive and can quickly spread throughout every region of the brain. Standard therapy consists of resection (surgery) followed by a combination of radiation and chemotherapy. Despite aggressive multimodal treatment, patients diagnosed with glioblastoma multiforme (GBM) have a dismal prognosis with an average survival time of slightly more than 1 year (15–16 months) [132].

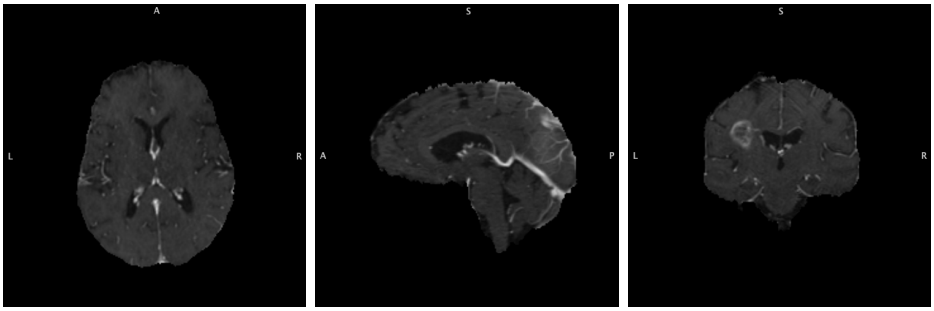
Neuro-imaging remains the cornerstone of tumor diagnosis and assessment during therapy. While treatment response continues to simply be assessed on changes in tumor size, new image analytic tools have been introduced to access additional information from clinical scans. Radiomics is a non-invasive method for the extraction of quantitative features from medical imaging that may not be apparent through traditional visual inspection [158]. By far, the most common neuro-imaging tool is magnetic resonance imaging (MRI). MRI provides an isotropic three dimensional (3D) picture of the brain with excellent soft tissue contrast and resolution without potentially harmful radiation. Typically, MRI imaging is performed in three different planes; axial, coronal, and sagittal. Commonly used MRI sequences include contrast enhanced T1-weighted (T1CE), T2-weighted, and fluid attenuation inversion recovery (FLAIR). Figure 3.1 shows a T1CE brain slice MR image of a Glioblastoma patient

in three planes. These sequences, because they are sensitive to different components of the tumor biology, play an integral role in providing refined anatomic and physiological detail. As seen in Figure 3.2, which shows multiple MRI sequences of a Glioblastoma patient, each sequence highlights a particular tumor component that it's sensitive to. For instance, T1-weighted post-contrast images are sensitive to the enhancing core and necrotic tumor core, whereas FLAIR and T2 highlight the peritumoral edema.

Recently, deep learning methods have been applied in computer vision and biomedical image analysis to improve the ability to extract features from images automatically and increase predictive power. They rely on advances in neural network architectures, GPU computing, and the advent of open source frameworks such as Tensorflow [1] and Pytorch [115]. When trained with sufficient data, deep neural networks can attain high accuracy in medical image analysis. The present state of the art is largely dominated by convolutional neural networks (CNN) with a number of effective architectures such as VGG-Net [129], Inception networks [133], ResNet [56], and EffecientNet [135]. Instead of manually engineering features, convolutional neural networks allow automatic convolutions to extract an enormous number of features from input images. Recent review works [19] show the dominance of convolutional neural network techniques applied to brain magnetic resonance imaging (MRI) analysis compared to other approaches.

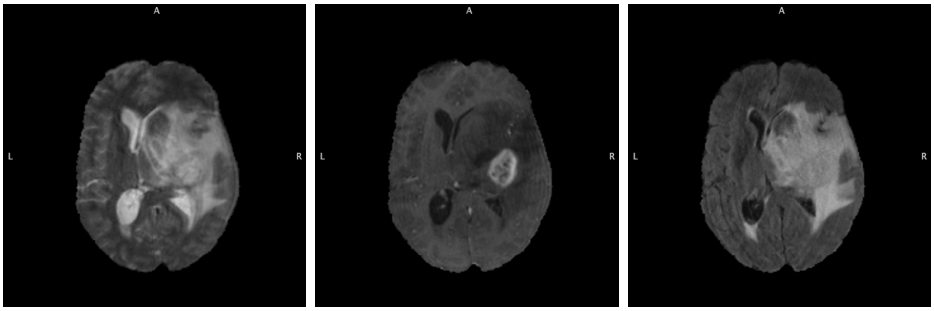
Survival time after a patient is diagnosed with high grade glioma depends on several factors such as age, treatment, tumor size, behavior, and location, as well as histologic and genetic markers [142]. Integration of automated image analytics can provide insights for diagnosis, therapy planning, monitoring, and prognosis prediction for gliomas. A new study [14] by UT Southwestern shows deep learning models can automatically classify IDH mutation status in brain gliomas from 3D MR imaging with more than 97% accuracy. This technology can potentially eliminate the current need for brain cancer patients to undergo invasive surgical procedures to help doctors determine the best treatment for their tumors.

This represents an important milestone towards non-surgical strategies for histological determination especially in highly eloquent brain areas such as the brain stem.



(a) Axial (b) Sagittal (c) Coronal

Figure 3.1: Different Planes of Brain MRI of a Glioblastoma Patient with the T1CE Sequence



(a) T2 (b) T1CE (c) FLAIR

Figure 3.2: Different Sequences of Brain MRI of a Glioblastoma Patient in the Axial Plane

The task of survival prediction for patient's with a glioma from MRI images is challenging. Several studies applying various methods and approaches using the BraTS dataset are reviewed in this section.

Authors of [7, 4, 117] proposed the use of some handcrafted and radiomics features extracted from automatically segmented volumes with region labels to train a random forest regression model to predict the survival time of GBM patients in days. The accuracies achieved by these studies on the validation set of the BraTS (Brain Tumor Image Seg-

mentation) 2019 challenge are respectively 52%, 51.7%, and 27.25% for classifying patients into three classes(long-survivors ($>$ 15 months), short-survivors ($<$ 10 months), and mid-survivors (between 10 and 15 months).)

Authors of [29] performed a two category (short- and long-term) survival classification task using a linear discriminant classifier that was trained with deep features extracted from a pre-trained convolutional neural network (CNN). The study achieved 68.8% accuracy when doing 5-fold cross validation on the BraTS 2017 dataset.

Ensemble learning was used by authors in [61]. They extracted handcrafted and radiomics features from automatically segmented MRI images of high grade gliomas and created an ensemble of multiple classifiers, including random forests, support vector machines, and multilayer perceptrons, to predict overall survival (OS) time on the BraTS 2018 testing set. They obtained an accuracy of 52%.

The authors of [94] achieved first place in the BraTS 2020 challenge for the overall survival prediction task (61.7% accuracy). They used the patient's age along with segmentation-derived features (their own segmentation model was used) to classify the patients into three groups (long, short and, mid-term survivors) using an ensemble of a linear regression model and random forests classifier.

Over the last decade, there was increasing interest in ensemble learning for tumor segmentation tasks as well. Ensemble learning was ubiquitous in the BraTS 2017-2020 challenges, being used in almost all of the top-ranked methods. The winner of the BraTS 2017 challenge for GBM tumor segmentation [145] was an ensemble of two fully convolutional neural network models (FCN), and a U-net each generating separate class confidence maps. Then, each class was created by averaging the confidence maps of the individual ensemble models for each voxel. This study reached dice scores of 0.90, 0.82, and 0.75 for the whole tumor, tumor core, and enhancing tumor, respectively for the BraTS 2017 validation set. Authors in [108] built an ensemble of UNet-based deep networks trained in a multi-fold setting to

perform segmentation of brain tumors from the T2 Fluid Attenuated Inversion Recovery (T2-FLAIR) sequences. They achieved a dice score of 0.882 for the BraTS 2018 set.

3.2 Deep Learning for COVID-19 Diagnosis from Xrays

At the end of the year 2019, we witnessed the start of the ongoing global pandemic caused by Coronavirus disease (COVID-19) which was first identified in December 2019 in Wuhan, China. As of May 2022, more than 515M cases are confirmed with more than 6.24M confirmed deaths Worldwide [152]. Coronavirus disease (COVID-19) is a contagious disease caused by the SARS-CoV-2 virus. Around 14% of patients with noticeable symptoms develop severe symptoms (more than 50% with lung involvement on imaging) [46]. Asymmetric peripheral ground-glass opacities without pleural effusions are some of the COVID-19 imaging features visible on chest X-rays and computed tomography (CT) of people who are symptomatic [82].

In the first few months of the pandemic, the testing ability was limited in the US and other countries. Testing for COVID-19 has been unable to keep up with the demand at times and some tests require significant time to produce results (days) [150]. Therefore, other timely approaches to diagnosis were worthy of investigation [147]. Chest X-rays (CXR) can be used to give relatively immediate diagnostic information. X-ray machines are available in almost all diagnostic medical settings, image acquisition is fast and relatively low cost.

Several groups have built open source databases of COVID-19 imaging data and multiple studies have been published claiming the possibility of diagnosing COVID-19 from chest X-rays using machine learning models with very high accuracy. Table 3.1 lists some of these studies along with their achieved performance results and the type of model validation used. As will be discussed, generalization to unseen sources is a problem.

Table 3.1: Papers for Automatic COVID-19 Prediction Based on CXR Images

Paper	**Performance Result**	**Testing Dataset**
Ozturk et al. [113]	0.99 AUC	train/test split from the same source
Abbas et al. [2]	0.94 AUC	train/test split from the same source
Farooq et al. [43]	96.23% Accuracy	train/test split from the same source
Lv et al. [87]	85.62% Accuracy	train/test split from the same source
Bassi and Attux [15]	97.80% Recall	train/test split from the same source
Rahimzadeh and Attar [119]	99.60% Accuracy	train/test split from the same source
Chowdhury et al. [32]	98.30% Accuracy	train/test split from the same source
Hemdan et al. [58]	0.89 F1-score	train/test split from the same source
Karim et al. [69]	83.00% Recall	train/test split from the same source
Krishnamoorthy et al. [75]	90% Accuracy	train/test split from the samesource
Minaee et al. [99]	90% Specificity	train/test split from the samesource
Afshar et al. [3]	98.3% Accuracy	train/test split from the samesource
Khobahi et al. [72]	93.5% Accuracy	train/test split from the samesource
Moura et al. [102]	90.27% Accuracy	train/test split from the samesource
Kumar et al. [76]	97.7% Accuracy	train/test split from the samesource
Castiglioni et al. [24]	0.80 AUC	train/test split from the samesource
Rahaman et al. [118]	89.3% Accuracy	train/test split from the samesource
DeGrave et al. [39]	0.995 AUC	train/test split from the samesource
Yeh et al. [154]	80.45% Specificity	train/test split from the samesource
Tartaglione et al. [137]	1.00 AUC	train/test split from the samesource
Hall et al. [52]	0.95 AUC	10-fold CV with mixed sources
Apostolopoulos et al. [10]	92.85% Accuracy	10-fold CV with mixed sources
Apostolopoulos et al. [9]	99.18% Accuracy	10-fold CV with mixed sources
Mukherjee et al. [104]	0.9908 AUC	10-fold CV with mixed sources
Das et al. [36]	1.00 AUC	10-fold CV with mixed sources
Razzak et al. [123]	98.75% Accuracy	10-fold CV with mixed sources
Basu et al. [16]	95.30% Accuracy	5-fold CV with mixed sources
Li et al. [81]	97.01% Accuracy	5-fold CV with mixed sources
Mukherjee et al. [105]	0.9995 AUC	5-fold CV with mixed sources
Moutounet-Cartan et al. [103]	93.9% Accuracy	5-fold CV with mixed sources
Ozturk et al. [114]	98.08% Accuracy	5-fold CV with mixed sources
Khan et al. [70]	89.6% Accuracy	4-fold CV with mixed sources

Chapter 4: Datasets

In this chapter, we describe all data sources used in this work along with some data acquisition and characteristics details.

4.1 Magnetic Resonance Imaging (MRIs) of Glioblastoma Patients

4.1.1 H. Lee Moffitt Cancer Center

H. Lee Moffitt Cancer Center and Research Institute is a nonprofit cancer treatment and research center located in Tampa, Florida. We obtained a set of deidentified imaging data of 22 Glioblastoma Multiforme (GBM) cases. Each case had four MRI sequences (including Tl-weighted (gadolinium-enhanced), T2-weighted, Fluid-attenuated Inversion Recovery (FLAIR), and the Apparent Diffusion Map (ADM)). The MRI scans were in the axial plane. The patients did not undergo surgery. For each patient, both pre and post-treatment scans were obtained. Fig. 4.1 shows examples of MRI sequences both pre and post-treatment.

4.1.2 Cancer Genome Atlas Glioblastoma Multiforme (TCIA-GBM)

The TCGA-GBM dataset [77] consists of a de-identified imaging set of brain cancer patients obtained from the Cancer Imaging Archive (TCIA). The Cancer Genome Atlas Glioblastoma Multiforme (TCGA-GBM) data collection includes CT, MR, and DX imaging modalities. For our study, we obtained the MR imaging modality. We excluded patients who underwent surgery or therapy and patients with missing survival data. Each case had several sequences and brain slices with tumor. After examining the scans, we kept 68 cases as our initial input dataset. The 68 cases are enumerated in Appendix B. The survival time ranged from 5 to 1731 days. The median value is 432.

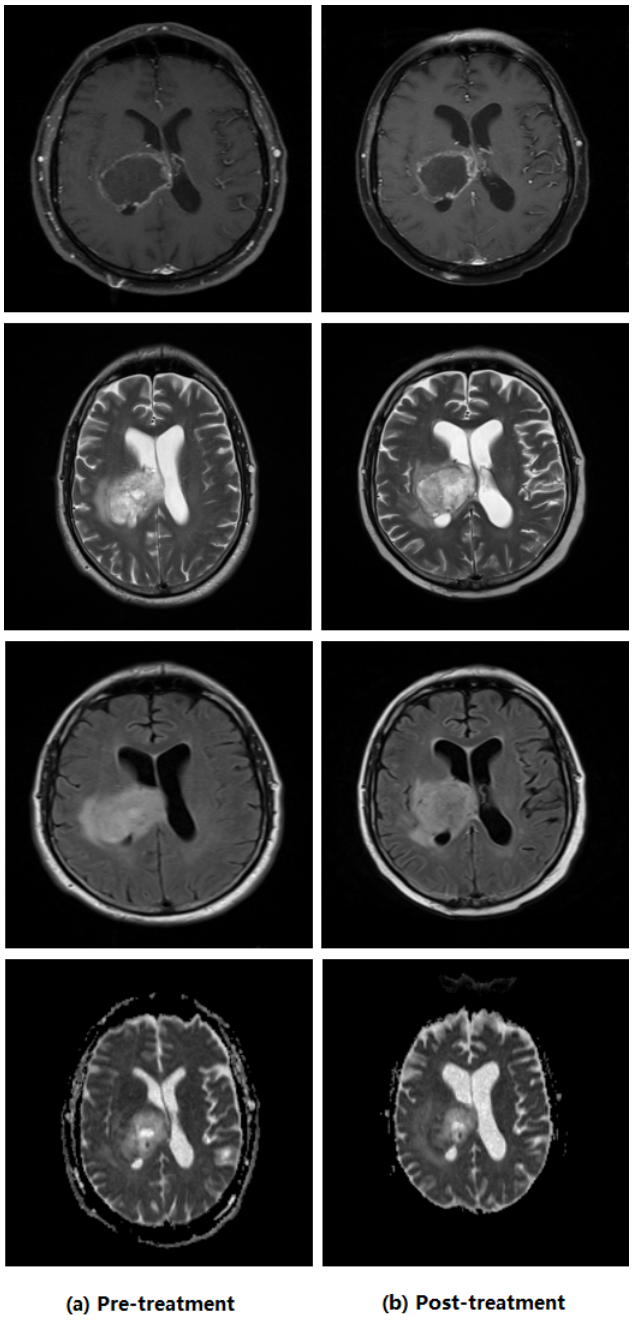


Figure 4.1: Examples of Multiple MRI Sequences Pre and Post Treatment, From Top to Bottom:T1-weighted, T2-weighted, FLAIR, and ADM Scans Respectively

4.1.3 Brain Tumor Segmentation (BraTS) Challenge Dataset

The Brain Tumor Segmentation (BraTS) Challenge is critical for benchmarking and has largely contributed to advancing machine learning applications in glioma image analysis.

The BraTS challenge was first held in 2012 and has taken place annually as part of the Medical Image Computing and Computer Assisted Intervention (MICCAI) conference ever since. It focuses on evaluating state-of-the-art methods for the segmentation of brain tumors in multimodal magnetic resonance imaging (MRI) scans. The glioma sub-regions considered for segmentation evaluation are: (1) The enhancing tumor (ET), (2) the tumor core (TC), and (3) the whole tumor (WT). The enhancing tumor sub-region is described by areas that are typically hypo-intense in T1CE when compared to T1. The tumor core (TC) consists of the bulk of the tumor, which is what is typically resected. The TC entails the ET, as well as the necrotic (fluid-filled) and the non-enhancing (solid) parts of the tumor. The WT corresponds to the complete extent of the disease, as it consists of the TC and the peritumoral edema (ED) [97].

To evaluate proposed segmentation methods, the participants are asked to upload their segmentation labels for regions in the image (e.g., edema, tumor) as a single multi-label file in the NIfTI format. Then metrics like Dice score and the Hausdorff distance are reported to rank teams' performances. In 2017, the BraTS challenge started to include another task, the prediction of overall survival time. Participants needed to use their produced segmentation in combination with other features to attempt to predict patient overall survival. The evaluation assessment of this task was based on an accuracy metric from a three-category classification (long-survivors (e.g., > 15 months), short-survivors (e.g., < 10 months), and mid-survivors (e.g., between 10 and 15 months)).

The Brain Tumor Segmentation (BraTS) challenge dataset is the largest publicly available glioma imaging dataset. It includes multi-institutional pre-operative clinically-acquired multi-sequence MRI scans of glioblastoma (GBM/HGG) and lower grade glioma (LGG) with overall survival data. Images were acquired from different institutions using MR scanners with different field strengths (1.5 T and 3 T). Segmentation ground truth labels are done by expert board-certified neuroradiologists. Scans are provided in NIfTI format and have (a) native (T1) and (b) post-contrast T1-weighted (T1CE), (c) T2-weighted (T2), and (d) T2

Fluid Attenuated Inversion Recovery (T2-FLAIR) sequences with 1–6-mm slice thickness. The BraTS data was made available after its pre-processing, i.e., co-registered to the same anatomical template, interpolated to the same resolution, and skull-stripped. The overall survival data is defined in days.

4.2 Chest X-rays (CXR) of Confirmed COVID-19 Patients

4.2.1 BIMCV-COVID-19

BIMCV-COVID-19 from Spain [38] is a large open-source dataset of chest X-ray images CXR (Computed Radiography (CR), Digital X-ray (DX)) and Computed Tomography (CT) images. It includes a subset of confirmed COVID-19 positive cases from the Valencian Region Medical ImageBank (BIMCV COVID-19(+)) and another subset for negative cases (BIMCV COVID-19(-)). Radiological reports are also included along with Polymerase chain reaction (PCR) results and other data.

4.2.2 COVID-19-AR

COVID-19-AR (USA) [40] is a collection of radiographic (X-ray) and CT imaging studies of patients from the University of Arkansas for Medical Sciences Translational Research Institute who tested positive for COVID-19. Each patient is described by a limited set of clinical data that includes demographics, comorbidities, selected lab data, and key radiology findings. The images are in DICOM format.

4.2.3 V2-COV19-NII

V2-COV19-NII (Germany) [151] is a repository containing image data collected by the Institute for Diagnostic and Interventional Radiology at the Hannover Medical School. It includes a dataset of COVID-19 cases with a focus on X-ray imaging. This includes images with extensive metadata, such as admission, ICU, laboratory, and anonymized patient data. The set contains raw, unprocessed, gray value image data as Nifti files.

4.2.4 COVIDGR

COVIDGR [134] is a set of X-ray images to assist in the diagnosis of the COVID-19 disease, built with the close collaboration of expert radiologists in Granada, Spain. All the images were in PosteriorAnterior (PA) view and were obtained from the same X-ray machine and under the same X-ray protocol.

4.2.5 C-CXRI-P

CC-CXRI-P (China) [143] is a dataset of chest X-Ray images (CXRs) constructed from cohorts from the China Consortium of Chest X-ray Image Investigation (CC-CXRI). All CXRs are classified into COVID-19 pneumonia due to SARS-CoV-2 virus infection, other viral pneumonia, bacterial pneumonia, other lung disorders, and normal controls.

4.2.6 Padchest

Padchest [22] is a large-scale dataset of high-quality chest X-ray imaging obtained from San Juan Hospital in Spain. The images in the dataset have multiple X-rays views and projections. Radiology reports are also included along with other clinical data.

4.2.7 Chexpert

CheXpert [64] is a large publicly available dataset for chest X-rays obtained from Stanford Hospital. Both frontal and lateral radiographs were included per patient. The images were labeled for the presence of 14 common respiratory diseases.

4.2.8 NIH

ChestX-ray8 [148] is a chest X-ray database containing over 100k frontal-view X-ray images of patients with 8 common lung diseases. The imaging labels were text-mined from the associated radiological reports using natural language processing.

Chapter 5: Transfer Learning from Small Medical Imaging Sets for Glioblastoma Prognosis[1]

5.1 Problem Statement

Glioblastoma is the most common type of malignant brain tumor and it usually grows and spreads quickly. The average age at diagnosis is 64 years and most patients survive approximately 12 to 16 months [132]. There remains no definitive cure for this disease but surgery, radiation therapy, systemic chemotherapy, and vasculogenesis inhibitors can improve overall survival time and alleviate symptoms.

Accurate prediction of survival time has the potential to both guide therapy and accurately inform patients of their prognosis. Magnetic Resonance Imaging (MRI) scans are the primary imaging modality used in the diagnosis and prognosis of brain tumors. Radiologists can usually confidently diagnose a GBM through visual interpretation of magnetic resonance images. An ongoing question, however, is whether survival time can be accurately predicted using imaging characteristics. Prior studies have suggested this might be possible using medical image analysis. Recently, machine learning techniques, especially deep learning, have been used to analyze clinical data to perform survival time prediction [6, 79].

Prediction of survival time from brain tumor magnetic resonance images (MRI) is not commonly performed and would ordinarily be a time-consuming process. However, current cross-sectional imaging techniques, particularly MRI, can be used to generate many features

[1]Parts of this chapter were published in Ahmed, Kaoutar B., et al. "Fine-tuning convolutional deep features for MRI based brain tumor classification." Medical Imaging 2017: Computer-Aided Diagnosis. Vol. 10134. SPIE, 2017. and
Renhao Liu, L. O. Hall, D. B. Goldgof, Mu Zhou, R. A. Gatenby and K. B. Ahmed, "Exploring deep features from brain tumor magnetic resonance images via transfer learning," 2016 International Joint Conference on Neural Networks (IJCNN), 2016, pp. 235-242, doi: 10.1109/IJCNN.2016.7727204.
Permission is included in Appendix C.

that may provide information on the patient's prognosis, including survival. This information can potentially be used to identify individuals who would benefit from more aggressive therapy.

Transfer learning is a machine learning technique that consists of transferring the knowledge gained to perform a given task in a source domain to perform a new but related task in a destination domain. The early experimental investigation of transfer learning in neural networks took place in 1976 by Bozinovski and Fulgosi [20]. The use of the previous knowledge as a starting point for a new task not only speeds up the learning process but also serves as a potential solution to the problem of the limited availability of training data in the medical domain.

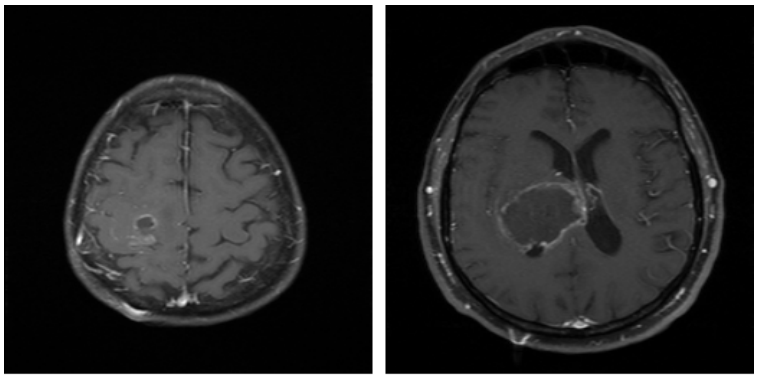
Another form of transfer learning is called fine-tuning. It requires modifying the actual architecture and re-training the entire model or only parts of it. The last fully connected layer has a number of nodes that depend on the number of classes in the dataset. Usually, this layer is replaced with a new one having the same number of neurons as the number of classes in the new dataset. We can fine-tune only the last fully connected layer newly added and freeze the rest or fine-tune the entire network or something in between.

Here, we show how transfer learning of features extracted from a deep neural network trained on ImageNet can be used to create an accurate classifier of overall survival of Glioblastoma patients from MRIs both when using a CNN as a feature extractor and when applying the fine-tuning approach.

5.2 Data Preprocessing

The dataset used here contained 22 cases of Glioblastoma Multiforme (GBM) which were obtained from the Moffitt Cancer Research Center. The region of interest (the brain tumor) was extracted manually with the help of a highly experienced radiologist (Dr. Robert Gatenby) using the T1-weighted MRI sequence. The dataset was divided into two classes based on survival time. A survival time of 365 days (which is when "long-term" survival

for Glioblastoma begins) was used as a cut-off. 12 cases were in the short-term class and 10 cases were in the long-term class. Figure 5.1 shows a MRI brain slice of a long vs short term survival Glioblastoma patients. We chose the MRI slice with the biggest tumor area for each



(a) Long term (b) Short term

Figure 5.1: MRI Brain Slice of a Long vs Short Term Survival Glioblastoma Patients

Glioblastoma patient. We also converted the 12-bit gray level images into the range [0,255], which is the range of color images in ImageNet. We then normalized all input images by subtracting the average image.

Here, we will be using the region of interest and tumor patch terms interchangeably. Since the model used here requires an input image size of 224×224 and the tumor size is commonly much smaller and typically variable, we had to resize the tumor patches. To extract the region of interest, we explored the "warp" and the "crop" methods for resizing. To create warped tumor patches, we drew a rectangle to cover the entire tumor as the ROI we wanted. Tumor sizes are different for each case, so the sizes of tumor patches extracted are different as well. We then resized all tumor patches into 224 × 224. To create cropped tumor patches, we drew a fixed size 40×40 pixel square tumor patch around the tumor. Then a resizing to 244x244 was done. The size of 40×40 was chosen as the average size of all tumors. Fig. 5.2 shows examples of the warped bounding boxes (exact fit box) around

the tumor and Fig. 5.3 shows examples of the cropped bounding boxes (standard size box) of the tumor patches.

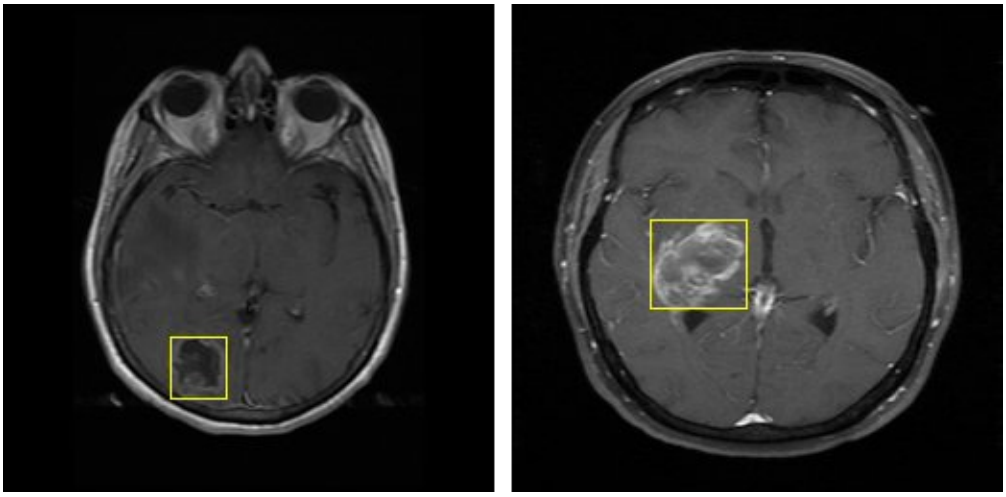


Figure 5.2: Examples of the Warped Bounding Boxes (Exact Fit Box) Around the Tumor

5.3 Pre-trained CNN Model

In deep learning, there are multiple challenges to training an entire CNN from scratch with randomly initialized weights. Essentially, training a deep convolutional neural network typically requires a large dataset like ImageNet which contains on the order of 1000 images per class, which is not an available option in the medical domain; another challenge is the high time cost needed for training modern CNNs even when using multiple GPUs.

Alternatively, pre-trained CNNs can be used as feature extractors for new tasks, especially for tasks with small datasets. To use a pre-trained CNN as a fixed feature extractor, we remove the last fully-connected layer of the pre-trained CNN (the layer leading to the output units) and treat the activations of the last hidden layer as a deep feature map for each input image. The extracted deep features vectors can be used to train another classifier. However, if we have a small data size we likely must reduce the number of features using a feature selection algorithm.

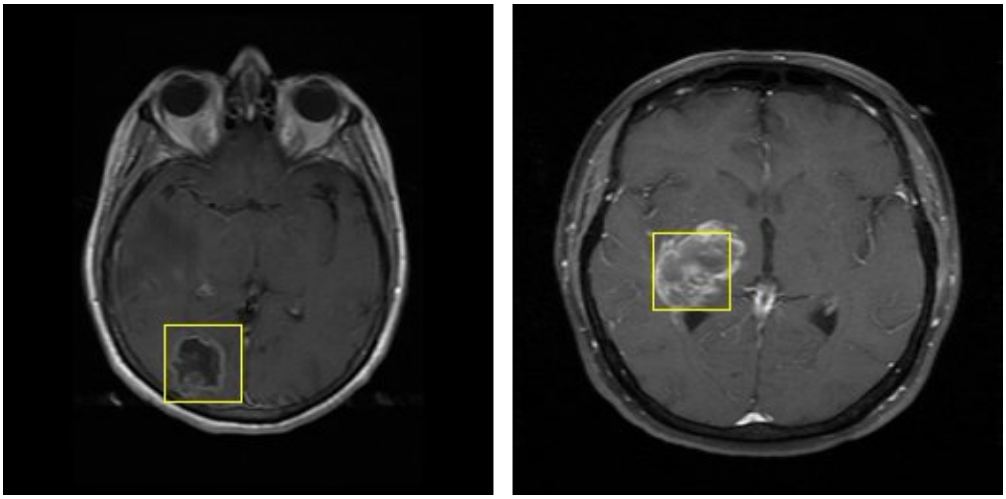


Figure 5.3: Examples of the Cropped Bounding Boxes (Standard Size Box) Around the Tumor

The pre-trained CNN we used in this study is the CNN-F presented in Chatfield's work [28]. The CNN-F architecture has 5 convolutional layers followed by 3 fully-connected layers. More details about the CNN parameters in each layer is provided in Table 5.1. The required input image size in CNN-F is 224×224 pixels. We chose CNN-F for its simple architecture, other pre-trained CNN architectures can be also considered for extended analysis. The CNN-F architecture was trained on ILSVRC-2012 challenge with the ImageNet dataset. Some of the parameter settings are: the momentum was 0.9; the weight decay was $5\text{x}10^{-4}$; the initial learning rate was 10^{-2} with a decreasing factor of 10. Detailed CNN architecture information and training settings can be found in [28].

We send the resized tumor patches as inputs to the model and it outputs a 4096-dimensional deep feature vector per image. The feature values come from the fully connected layer fc7. The pre-trained model was provided by the MATLAB toolbox "MatConvNet" [141]. Since our images are grayscale, we chose to deactivate the pre-trained information from the Green and Blue channels in the CNN-F model and only use the Red channel , which is the lowest frequency. Other channels can also be considered for further analysis.

Table 5.1: Detailed Parameters of the CNN-F Model

Layer	Parameters
conv1	64×11×11 st. 4, pad 0
conv2	256×5×5 st. 1, pad 2
conv3	256×3×3 st. 1, pad 1
conv4	256×3×3 st. 1, pad 1
conv5	256×3×3 st. 1, pad 1
fc6	4096 dropout
fc7	4096 dropout
fc8	1000 softmax

5.4 Experimental Results

5.4.1 CNN-F as a Feature Extractor

Here we explore the use of the pre-trained model CNN-F as a feature extractor, then use the extracted deep features to train other classifiers. We sent the 22 images as inputs to the model, then we extracted a 22×4096 deep feature vector. We chose to use symmetrical uncertainty in Weka [53], [155] to select features. First, we experimented with k-nearest neighbors classifier, we tested IBk with k=1 after selecting 1 feature. The results are shown in Tables 5.2 and 5.3 for the warped and cropped tumor patches respectively.

Table 5.2: Nearest Neighbor Classifier Accuracy Results from "Warped" Tumor Patches Using a Single Feature

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	36.36%	27.27%
T2-Weighted	63.64%	95.45%
FLAIR	86.36%	72.73%
ADM	22.73%	31.82%

Table 5.3: Nearest Neighbor Classifier Accuracy Results from "Cropped" Tumor Patches Using a Single Feature

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	36.36%	45.45%
T2-Weighted	40.91%	40.91%
FLAIR	40.91%	63.64%
ADM	18.18%	31.82%

With the warped tumor patches and pre-treatment deep features from FLAIR images, the accuracy was 86.36%, and from T2-weighted post-treatment images, the accuracy was 95.45%

We also applied the J48 decision tree and Random Forests classifiers from Weka and tested with leave-one-out cross-validation. In each fold, the best single feature was selected by the symmetrical uncertainty feature selector. For J48 settings, the confidence factor was 0.25; the minimum number of instances per leaf was 2. For Random Forests, the number of attributes to be used in random selection was 1; the number of trees to be generated was 200. The predictive accuracies for the "warped" tumor patches are shown in Table 5.4 and Table 5.5. The results in Table 5.6 and 5.7 are for the "cropped" tumor patches.

Table 5.4: J48 Classifier Accuracy Results from "Warped" Tumor Patches

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	40.91%	18.18%
T2-Weighted	68.18%	86.36%
FLAIR	86.36%	72.73%
ADM	27.27%	36.36%

In these experiments, post-treatment features yielded the best accuracy. For warping, the FLAIR and T2-weighted features were best. For cropping, the FLAIR Post-treatment feature yielded an accuracy in the 60% range.

Table 5.5: Random Forests Classifier Accuracy Results from "Warped" Tumor Patches

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	36.36%	27.27%
T2-Weighted	63.64%	95.45%
FLAIR	86.36%	72.73%
ADM	22.73%	31.82%

Table 5.6: J48 Classifier Accuracy Results from "Cropped" Tumor Patches

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	40.91%	40.91%
T2-Weighted	36.36%	36.36%
FLAIR	36.36%	68.18%
ADM	27.27%	31.82%

Table 5.7: Random Forests Classifier Accuracy Results from "Cropped" Tumor Patches

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	36.36%	45.45%
T2-Weighted	40.91%	40.91%
FLAIR	40.91%	63.64%
ADM	18.18%	31.82%

5.4.2 Fine-Tuning

A common practice for fine-tuning is to replace the last fully connected layer of the pre-trained CNN with a new fully connected layer that has as many neurons as the number of classes in the target task. In our experiment, we deal with 2-class classification tasks; therefore, the new fully connected layer has 2 neurons. Instead of randomly initializing the weights of the last fully connected layer, we kept the weights of the two classes that had the highest activation values among the 1000 original classes. The CNN was fine-tuned with stochastic gradient descent with a learning rate set to be less than the initial learning rate used in the pre-training phase. This ensures that the features learned from the larger dataset are not completely forgotten. In all our fine-tuning experiments we trained the RandomForests classifier from Weka on the learned features extracted from the second fully connected layer. We set the number of trees parameter to 200. We chose to use symmetrical

uncertainty in Weka to select the best 7 features. The number of features was chosen to be no larger than 1/3 the size of the data set. We tested with leave-one-out cross-validation. Tables 5.8 and 5.9 show the accuracy results of RandomForests for the warped and cropped tumor patches respectively.

Table 5.8: RandomForest Accuracy Results Using "Warped" Tumor Patches

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	68.18%	27.27%
T2-Weighted	18.18%	27.27%
FLAIR	63.64%	63.64%
ADM	31.82%	50%

Table 5.9: RandomForests Accuracy Results Using "Cropped" Tumor Patches

MRI Sequence	Pre-treatment features	Post-treatment features
T1-Weighted	40.91%	50%
T2-Weighted	72.73%	36.36%
FLAIR	63.64%	50%
ADM	27.27%	45.45%

In these experiments, the pre-treatment features allowed the best accuracy. For the exact masks, T1-weighted and Flair scored the highest accuracies. For the cropped method, T2-weighted and Flair were the best. We ran a cross-validation fine-tuning by fully training all the layers of the CNN with 21 images and testing with the 1 remaining image. A learning rate of 0.00001 was used for the first 4 iterations and then it was decreased every 4 iterations by an order of magnitude. We ran the fine-tuning for 10 iterations. We chose the pre-treatment Flair sequence for fine-tuning. It had a consistent accuracy for both exact and cropped masks. We graphed the mean squared error (MSE) of the output of the last fully connected layer fc8. Fig. 5.4 shows the MSE evolution for one fold cross validation tuning. As a stopping criterion, we chose the iteration before the MSE increases.

Using this method we were able to improve prediction results using fine-tuning from 63.64% to 81.81% with the pre-treatment Flair sequence.

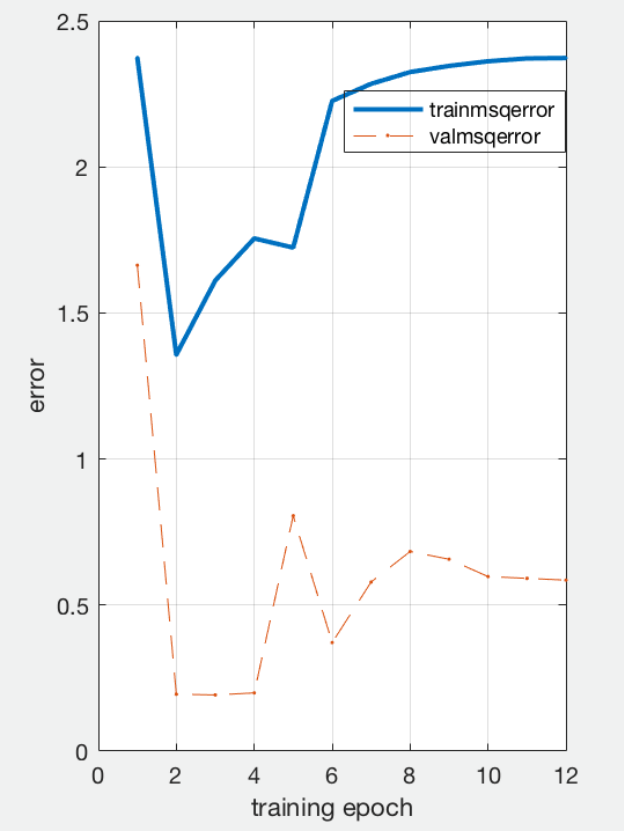


Figure 5.4: Mean Squared Error of the Last Fully Connected Layer of the Network

We also wanted to further test the effectiveness of other fine-tuning strategies. Also, it was desirable to explore how much improvement we could get from a sequence that did not result in good accuracy previously. So, we chose T1-weighted images from cropped tumor patches, pre-treatment, which had an accuracy of only 40.91%. A learning rate of 0.0001 was used for the first 10 iterations and then it was reduced to 0.00001. We fixed the weights of the five convolutional layers of the pre-trained network and only fine-tuned succeeding layers. This was motivated by the studies that noticed the generalization of the earlier features of a CNN and that they should be useful for different tasks, but later layers of the CNN become more specific to the details of the original dataset. Especially with small

Table 5.10: Evolution of the Prediction Accuracy of the Fine-tuned Model across Iterations

Iteration	Accuracy
Iter1	40.91%
Iter5	54.54%
Iter10	54.54%
Iter15	54.54%
Iter20	59.09%

training datasets, it may be a good idea to partially fine-tune the network instead of full tuning due to overfitting concerns. So, we only fine-tuned the fully connected layers. We then used dropout. We augmented the training dataset size by incrementally rotating the original 22 images. Rotations clockwise of every 5 degrees were used skipping the last 2. Thus generating additional training images. We ran the fine-tuning for 20 iterations. The same setup as previously used was done in a leave-one-out cross-validation. We note that all rotated images of a test image were removed from the training set. Table 5.10 shows the evolution of the prediction accuracy of the fine-tuned network across iterations.

5.5 Discussion

The results shown in Section 5.4.1 are promising on some MRI sequences. Accuracy is very low on others. Symmetrical uncertainty feature selection provided better accuracy on the warped images with a 95.45% accuracy on post-treatment T2-weighted images.

As seen in the results presented in Section 5.4.2, fine-tuning CNN-F improved the prediction accuracy from 63.64% to 81.81% using the pre-treatment Flair sequence. Also, it can be noticed in Table 5.10 that finetuning improved the model's accuracy from 40.91% to 59.09% using T1-weighted images from cropped tumor patches, pre-treatment. In addition, the model shows a slow but steady improvement in its performance. It is possible that with more iterations the accuracy will be further improved.

5.6 Summary

Convolutional neural networks and features extracted from them seem to work well for medical images. In this study, we explored feature learning and most importantly deep learning methods using MRI data of glioblastoma brain tumors to perform survival time prediction.

This chapter explored the potential of using a pre-trained convolutional neural network as a feature extractor to extract deep feature representations of brain tumor MRIs. The results demonstrate that deep features can enable classifiers to achieve a predictive power of 95% accuracy. From one feature an accuracy of 90.91% was obtained from pre-treatment images showing promise for differential treatment. This finding suggests a novel feature extraction method from MRI data.

As large datasets to train complex models are often not available, transferring features of a previously trained model by fine-tuning to the new task can help. Predictive accuracy of 81.8% was achieved using the Flair sequence. Fine-tuning improved accuracy by allowing a RandomForests classifier to get 4 more examples correct. A gain of 4 correctly predicted examples was also obtained when we tuned using T1-weighted images, for which less accuracy was initially obtained. Our experiments further confirm the potential of CNNs for medical imaging applications because both the use of pre-trained CNNs as a feature extractor and fine-tuned CNNs provide very good accuracy without the need for additional types of feature extraction.

Chapter 6: Ensemble Learning from Small Medical Imaging Sets for Glioblastoma Prognosis[2]

6.1 Problem Statement

The application of machine learning to survival prediction based on conventional imaging studies has demonstrated that such methods can be used to develop a decision support system that improves survival estimation and cancer prognosis in general. Here, we suggest a novel CNN ensemble training and evaluation approach to predict overall survival time in patients with Glioblastoma when dealing with small datasets.

6.2 Data Preprocessing

The input dataset consists of de-identified magnetic resonance images of Glioblastoma patients obtained from the Cancer Imaging Archive (TCIA). In this set of experiments, we used the T1-weighted sequence (gadolinium-enhanced) and from multiple transverse slices in the brain volume, we chose the single slice with the largest tumor region. Using the division criterion of 365 days (1 year), which is considered the beginning of long-term survival, there were 35 long-term survival cases and 33 short-term.

We extracted the region of interest (ROI) that contained the tumor using exact masks that were outlined manually with the help of a highly experienced radiologist (Dr. Robert Gatenby). As seen in Fig. 6.1, we drew a mask that covers the tumor area with a reasonable approximation. Since tumor sizes are different for every patient, the extracted patches are

[2]Parts of this chapter were published in K. B. Ahmed, L. O. Hall, R. Liu, R. A. Gatenby and D. B. Goldgof, "Neuroimaging Based Survival Time Prediction of GBM Patients Using CNNs from Small Data," 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), 2019, pp. 1331-1335, doi: 10.1109/SMC.2019.8913929.
Permission is included in Appendix C.

also different in size. We, therefore, resized all extracted patches to the size of the largest tumor, which was 120x120. The resized patches served as input to the convolutional neural networks. To avoid over-fitting when training a neural network with a small dataset we augmented the dataset using transformations such as flip and rotate. Each image is mirrored around the y-axis producing two samples for one image; then each one is incrementally rotated around the center point by 5 degrees resulting in 71 samples of each image. Consequently, image augmentation enabled us to increase the size of the dataset, originally 68, to roughly 10,000 augmented images.

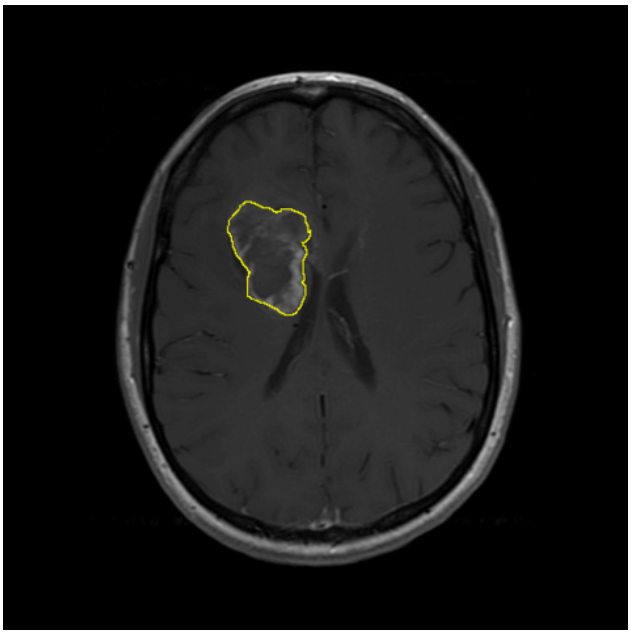


Figure 6.1: ROI Segmented with an Exact Tumor Mask

6.3 CNN Ensemble Training and Evaluation

With small data, a small convolutional neural network was built and used to classify the 68 cases into two classes, short and long-term survival. To decide the suitable network size, we experimented with several CNN architectures using leave-one-out cross-validation. The

Table 6.1: Detailed Parameters of the Proposed CNN Architecture

Layer	Parameters
Conv1	3×3×20 stride 1 padding 0 x2 pooling Leaky ReLU Batch Normalization Dropout 20%
Conv2	3×3×20 stride 1 padding 0 x2 pooling Leaky ReLU Batch Normalization Dropout 30%
FC1	40 Leaky ReLU Dropout 50%
Output	1 Sigmoid

detailed parameters of each layer are described in Table 6.1 and the overall architecture is illustrated in Fig. 6.2.

The first parameter in Table 6.1 consists of the kernel size and the number of filters: 'size × size × number'; the second and third indicate the convolutional stride and spatial padding. The fourth parameter indicates the max-pooling down-sampling factor. Dropout was added after each convolutional layer to avoid over-fitting. The fully connected layer (FC1) has 40 units. Leaky ReLU of factor 0.3 and batch normalization were also used to reduce over-fitting. Finally, the output activation function is a sigmoid. The weights were randomly initialized. The initial learning rate was 10^{-3}, which is reduced in each epoch when using Keras's SGD (Stochastic Gradient Descent) optimizer with a momentum of 0.9. The Dropout rate varies by layer as shown in Table 6.1. The total number of training epochs was 200. The 68 models, for leave-one-out testing, needed 48 hours for training on a GeForce GTX 1080 Ti with a batch size of 34. The framework was developed in Tensorflow.

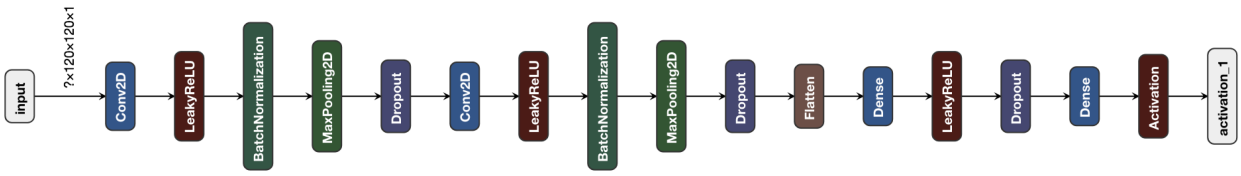


Figure 6.2: Proposed CNN Architecture

To overcome the limitation of the small training set size, we followed a leave-one-out training protocol to provide test accuracy. In each fold, 67 images and their augmentations are used for training; the remaining image is then used for testing. This training procedure results in 68 trained networks. The overall accuracy is then averaged over all trained networks.

To improve the evaluation outcome, we used an ensemble technique. Normally, we would report one single testing accuracy, which is the result of leave-one-out testing of the trained models saved on one particular epoch while training. However, since we did not create a validation set, due to the small data size, we will not be able to choose the exact best epoch for early stopping. Therefore, we created an ensemble of snapshots from different epochs. This ensemble can reduce variance in error. We used majority voting to report the class of a testing image. For each testing image, we report the class at each of the epochs. Then, we count the number of times the image was correctly classified across epochs. An image is overall correctly classified if the ensemble did a correct classification more than 50% of the time. The overall accuracy is then the average over the 68 cases. Fig. 6.3 shows the overall workflow of the proposed evaluation method.

6.4 Experimental Results

6.4.1 Results of Ensemble Training and Evaluation

We measured the training accuracy and loss during the training steps as shown in Fig. 6.4 It is observed that training accuracy increases as training proceeds. The final training

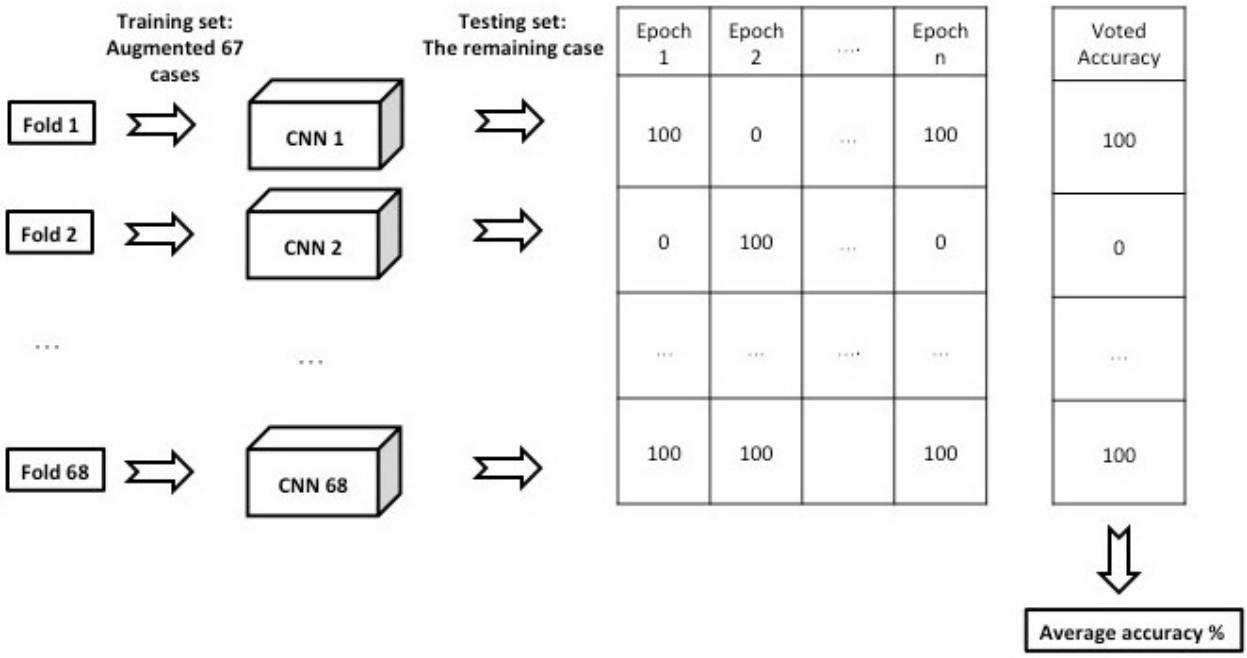


Figure 6.3: Flowchart of Leave-one-out Training and Ensemble Technique

Table 6.2: Accuracy Results of Individual Epochs vs the Ensemble

Epoch	Accuracy
75	50%
100	58.82%
125	51.47%
150	60.29%
175	55.88%
200	58.82%
The Ensemble	72.06%

accuracies per fold were around 95% at 200 epochs. To create the ensemble in each fold, we chose snapshots at epochs 75, 100, 125, 150, 175, and 200. As seen in Fig. 6.4, the training accuracy keeps increasing rapidly in the first section of epochs [0-75]. Then, it starts to stabilize afterward. Therefore we choose the epoch region [75-200]. We selected epochs separated by 25 steps to give more opportunity for the CNN weights to have fairly big changes. Table 6.2 presents the best results obtained after setting up the CNN and fine tuning the hyper parameters.

Figure 6.4: Loss and Accuracy of One Fold of the LOO Training by Epochs

6.4.2 Comparison to Other Methods

In this experiment, we dealt with the task of classifying MRI images of brain cancer patients into two groups short-term and long-term survival time. With less than the desired amount of data in training, transfer learning was explored. As a pre-trained model, we used VGG-16 with the python API with the TensorFlow backend. The model was pre-trained using the benchmark dataset ImageNet ILSVRC, which contains around 1.2M training images, 50,000 validation images, and 100,000 test images with 1,000 object categories. Our images had to be resized using bicubic interpolation to meet the input size requirement of the model. We also normalized the grayscale MRI images to be in the range of [0-255] using a linear transformation. While the input image for the pre-trained CNN is expected to be a color RGB image, our dataset consists of a grayscale image. To overcome this obstacle, we duplicated our images over three channels. The images were input to perform a forward propagation then features were extracted from the activations of the last fully connected

Table 6.3: Performance Comparison of Different Models

Model	Accuracy(%)	Specificity(%)	Sensitivity(%)
VGG16 – Last FC layer	66.18	65.71	66.67
Resnet-50	64.71	80.00	48.48
VGG16- last Conv layer	55.88	42.86	69.70
InceptionV3	54.41	51.42	57.57
CNN as Feature Extractor	50	54.29	45.45

layer. The extracted deep features were then used to train a classifier after selecting the best features using the symmetric uncertainty feature selector. For comparison purposes, we attempted the same methodology explained with two other pre-trained models: Resnet50 and Google's Inception v3. Additionally, we also tried using our trained small CNN presented in Section 6.3 as a feature extractor. We removed the Sigmoid block and extracted the 40-D feature vector and used it to train and evaluate a decision tree classifier.

Table 6.3 summarizes and compares the best results of the transfer learning experiments using pre-trained models. The extracted features were used to train and evaluate Weka's J48 decision tree after selecting the best features using the symmetric uncertainty feature selector. Default Weka parameters were selected for J48.

To compare with handcrafted features, we extracted 81 HoG (Histogram of Oriented Gradients) features [35], which are known to capture shape and edge information, from the input dataset and trained J48 and Random Forests classifiers. We evaluated the trained classifiers using 10-fold cross-validation. An accuracy of 54.41% was obtained using Random Forests while J48 achieved an accuracy of 39.71%.

6.5 Discussion

As can be seen in Table 6.2, our snapshot ensemble method succeeded in boosting the overall accuracy of the CNN. The best individual accuracy after 150 epochs of training was 60.29%.

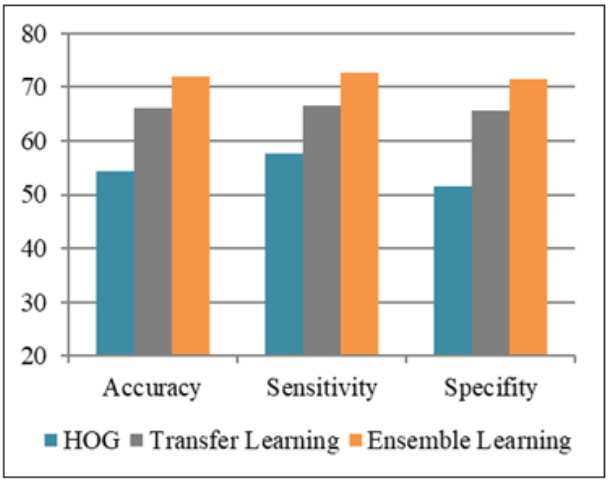


Figure 6.5: Comparison of Best Performance Results of Different Approaches

As seen in the Table 6.3, the features extracted from VGG16 enabled higher accuracy than the features extracted from the other two pre-trained models. We also attempted to take the features from the last convolutional layer of VGG16 before the first fully connected layer. However, the testing accuracy was only 55.88% using leave one out cross-validation. We can also see that pulling out features from the last fully connected layer of the small CNN and using them to train a classifier did not perform as well as using the training and evaluation method proposed in Section 6.3.

To allow comparison of all strategies Fig. 6.5 summarizes the overall best accuracy results of different discussed approaches. Training a small CNN from scratch and using ensemble learning outperforms the transfer-learning method and both strategies result in better performance than using HOG features. This result demonstrates the power of deep features compared to manually extracted features such as HoG.

6.6 Summary

This work shows the advantages and capability of deep features extracted from convolutional neural networks to generate radiomics signatures to predict the survival time of glioblastoma (GBM) patients using MRI brain scans as input data. We compared several

different approaches to this task. Our work also shows the possibility of training a CNN to predict survival time even with a small dataset and limited sequences and a variable set of acquisitions. An accuracy of 72.06% was obtained when training a small CNN using our ensemble training and evaluation method. Transfer learning using a pre-trained convolutional neural network is also a good strategy and achieved an accuracy of 66.18% using features from the VGG16 pre-trained model. This work has the potential to help practitioners leverage the benefits of deep learning and overcome the challenge of the limited availability of medical imaging for training.

Chapter 7: Snapshot Learning from Small Medical Imaging for Glioblastoma Prognosis[3]

7.1 Problem Statement

Applying deep learning algorithms to multi-sequence MR images is challenging due to several factors. The first is the size of the 3D data which can be expected to allow the most accurate classification. Second, is the difficulty in designing an appropriate neural network architecture for the desired objective. Third (and most critically), the limited clinical knowledge of deep learning experts and the limited availability of large medical image datasets with labels. Finally, the predictive power of deep convolutional neural network models is significantly enhanced by a large training dataset. Using a small number of cases to train a model using deep learning can result in over-fitting the training data such that generalization to future cases is poor. An ensemble of deep neural networks is known to be more accurate and robust. However, training multiple deep networks is computationally expensive. Huang et al. [62] introduced snapshot learning which is a fast and effective way of creating an ensemble of models without additional training costs by saving one model's weights snapshots at different points over the course of training. Our preliminary results [5] showed the effectiveness of an ensemble of snapshots in boosting the prediction performance using our limited size dataset. Here, we address the issues of limited and heterogeneous data sets in medical imaging using a snapshot ensemble learning method applied to overall

[3]Parts of this chapter were published in Ben Ahmed K, Hall LO, Goldgof DB, Gatenby R. Ensembles of Convolutional Neural Networks for Survival Time Estimation of High-Grade Glioma Patients from Multimodal MRI. Diagnostics (Basel). 2022 Jan 29;12(2):345. doi: 10.3390/diagnostics12020345. PMID: 35204436; PMCID: PMC8871067.
The authors hold copyright rights under a CC BY license.

survival prediction of glioblastoma patients. Its efficacy is demonstrated through application to MR images from the BraTS dataset.

7.2 Data Preprocessing

In this study, we used subsets of glioblastoma (GBM) cases from the BraTS 2019 training dataset. We randomly split the data into train/test subsets. The training subset consisted of 163 cases and the testing set had 46 cases. Among the 163 cases, there were 81 long-term and 82 short-term survival cases based on a 12-month cut-off (which is the middle threshold of the mid-term survivors class in BraTS data). Among the 46 test cases, there were 23 long-term and 23 short-term survivors. The data can be downloaded online from the CBICA's Image Processing Portal (IPP) after requesting permission from the BraTS team. Figure 7.1 shows the Kaplan–Meier survival graph corresponding to the training dataset.

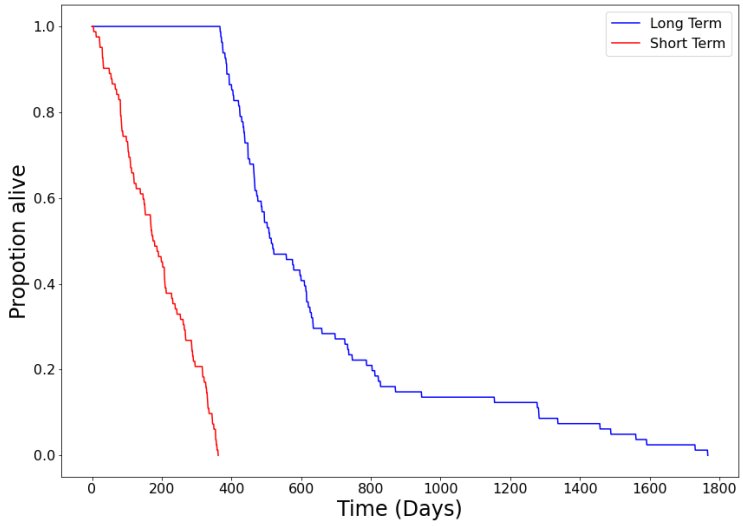


Figure 7.1: Kaplan-Meier Survival Graph of the Training Set

The first step is tumor segmentation, we decided to use the publicly available pre-trained model of Wang et al. [144] to automatically segment the training and testing set used in this study. The segmented volumes were used as a starting point for our task of survival class prediction.

Data augmentation is a fundamental way to help reduce model over-fitting when dealing with a small amount of training data. Hence, we performed transformations to each 2D slice consisting of vertical flipping and rotations of 10 degrees incrementally from 0 to 180 degrees. In addition, we applied elastic deformation by generating smooth deformations using random displacement fields that are sampled from a Gaussian distribution with a standard deviation of 17 (in pixels). The new values for each pixel were then calculated using bicubic interpolation. The augmentation was applied to each 2D slice which then gets stacked back in to form a 3D cube. The transformations resulted in increasing the size of training data from 163 images per sequence to roughly 10,000 augmented images, including the original images.

Each MRI volume is high resolution with a size of $240 \times 240 \times 155$ (155 brain slice images of size 240×240). To fit these volumes into the GPU memory for model training, we had to remove zero voxels and null slices, filled with air, that are unnecessary for building a good predictive model. The cropping was done using the Nilearn python package [109]. As a result, all the images were resized to a uniform size of $160 \times 160 \times 110$, which is the average size over all cases. Furthermore, we used the Keras [30] data generator to feed data in batches to our model instead of loading all data into memory at once. In addition, before feeding input images to the model, as a part of preprocessing, the images were normalized to have zero mean and unit standard deviation.

7.3 System Outline

As shown in Figure 7.2, the overall workflow of our system consists of three stages. Firstly, the starting point was the raw 3D MRI images acquired from the BraTS platform after

securing permission. The images were sent as inputs to the pre-trained segmentation network [144] resulting in a 3D segmented tumor in the NIfTI format. Image augmentation and pre-processing were done before the inputs were sent to our 3D convolutional neural network for automatic feature extraction and classifier training. Finally, the endpoint is the overall survival prediction. Ensemble learning was used to evaluate the prediction performance of the trained networks. More details on each phase are provided in the following sub-sections.

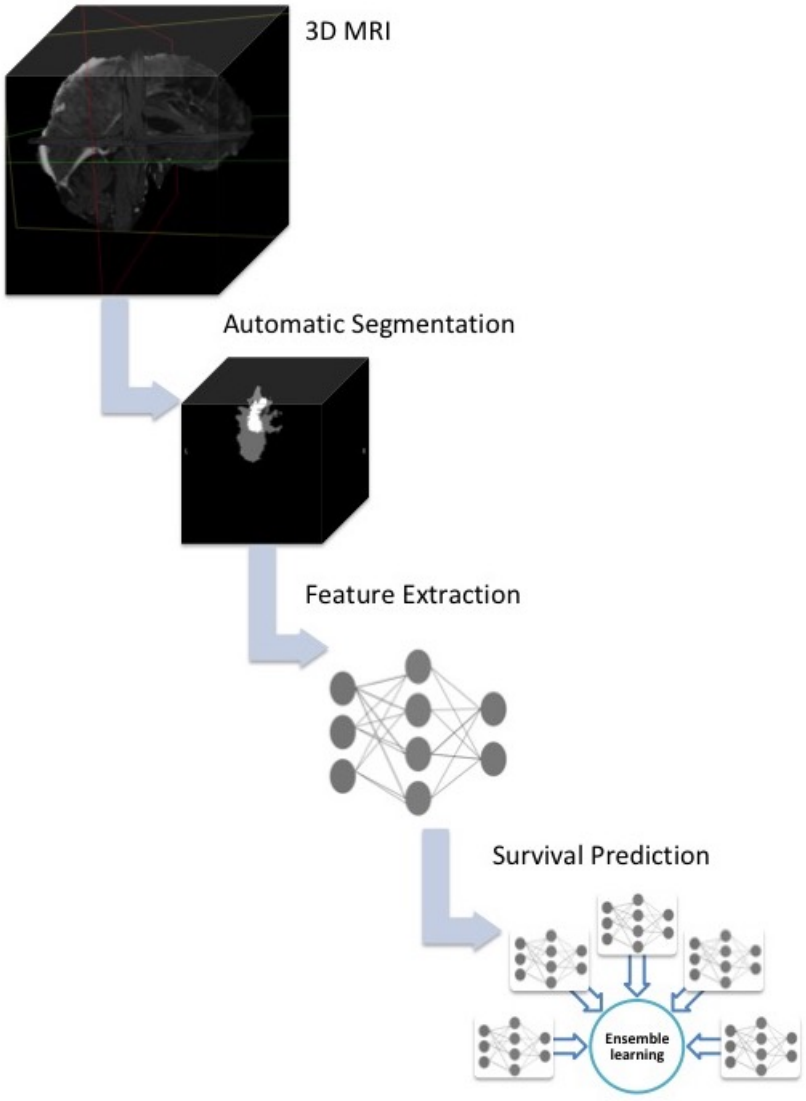


Figure 7.2: Pipeline of Proposed Overall Survival Prediction System

7.4 Main CNN Model Construction and Training

Finding the most efficient network parameters for a given dataset is important. It is known that parameters like batch size, number of epochs, regularization, and learning rate can significantly affect the final performance. There is no predefined method to select the ideal architecture and parameters. Therefore, we attempted multiple combinations of settings and parameters. Our model consists of three $3 \times 3 \times 3$ convolutional layers (Conv) with a stride of $1 \times 1 \times 1$ size and ReLU activation function followed by a $2 \times 2 \times 2$ max-pooling layer with a stride of $1 \times 2 \times 2$. The first two convolutional layers have 20 filters and the third has 40. Padding was used for the convolutional layers. A fully connected layer (FC) of 64 units follows a global average pooling operation. Finally, an output with sigmoid activation is used to generate a prediction. To reduce overfitting, a dropout of 50% was added after the fully connected layer. The developed model consists of a total of 35,709 parameters.

The implementation of the model was carried out with Keras 2.3.1 using TensorFlow 1.2 as the backend. Code is available on GitHub [1].

7.5 Experimental Results

7.5.1 Parameter Exploration Results

In this section, we report some preliminary experiments performed to determine the best parameters for our dataset. Here, for simplicity, we only used a T1 contrast-enhanced (T1CE) sequence as input.

7.5.1.1 Choice of Snapshots

The original snapshot learning method showed using the cosine annealing cyclic learning rate was most effective. It works by saving snapshots at the end of each cycle (X number of training epochs). Here, we compare the performance results of the following: The original

[1] https://github.com/kbenahmed89/Glioma-Survival-Prediction

method versus using a fixed learning rate and selecting snapshots based on the training accuracy evolution. When not using cyclic learning the stochastic gradient descent optimizer was used for training the models with the starting learning rate set to 10^{-3} and momentum set to 0.9. The network state was saved at each epoch. We selected snapshots based on the training accuracy graph. The accuracy of the model starts to increase rapidly in the beginning due to the high initial learning rate. Then, it slows down and keeps increasing at a lower rate and with high stability. During the last period of training, the training accuracy has already reached 100% thus over-fitting the training data. Therefore, we decided to avoid the last set of training epochs and only focus on the earlier phase of training. To create an ensemble of snapshots, we picked the weights of every five epochs to allow a fair amount of change in the network weights. We go back 5 steps from when the training accuracy reaches 80% (epoch 30) and then proceed back to choose every fifth epoch as a snapshot (epochs 30, 25, 20, 15, and 10). The performance results of testing the individual five snapshots on the 46 examples testing set are shown in Table 7.1 along with the results of the majority voting-based ensemble of snapshots.

To see if the original approach to snapshots might be better, we trained using a cosine annealing cyclic learning rate with a maximum learning rate of 10^{-3}. The training was done for 500 epochs and snapshots were saved at the end of each of the 10 cycles, which were 50 epochs long. The majority voting-based ensemble of the last five snapshots saved at the end of the last five cycles achieved a 52.17% accuracy on the unseen test set of 46 test cases using only T1CE MRI modality. This was not as good as our approach of choosing snapshots from the training data. Of course, it is possible that a different set of parameters would result in better performance, but we had little data to test with and wanted to leave the test set alone until parameters were chosen.

Based on this experiment, we decided that our choice of snapshots based on the training graph works best for this particular dataset. Therefore, we used this snapshot choice method for the remainder of the research.

Table 7.1: Performance Results of the Ensemble of Five Snapshots Using 3D T1CE Sequence

Method	Accuracy	Specificity	Sensitivity	AUC
Snapshot 1	63%	39%	87%	0.63
Snapshot 2	67%	74%	61%	0.67
Snapshot 3	70%	74%	65%	0.70
Snapshot 4	61%	74%	48%	0.61
Snapshot 5	59%	52%	65%	0.59
Ensemble	70%	70%	70%	0.70

Table 7.2: Performance Results of the Ensemble of Five CNNs Using 3D T1CE Sequence

Method	Accuracy	Specificity	Sensitivity	AUC
CNN 1	63%	48%	78%	0.63
CNN 2	54%	48%	61%	0.54
CNN 3	61%	78%	43%	0.61
CNN 4	61%	65%	57%	0.61
CNN 5	57%	43%	70%	0.57
Ensemble	67%	70%	65%	0.67

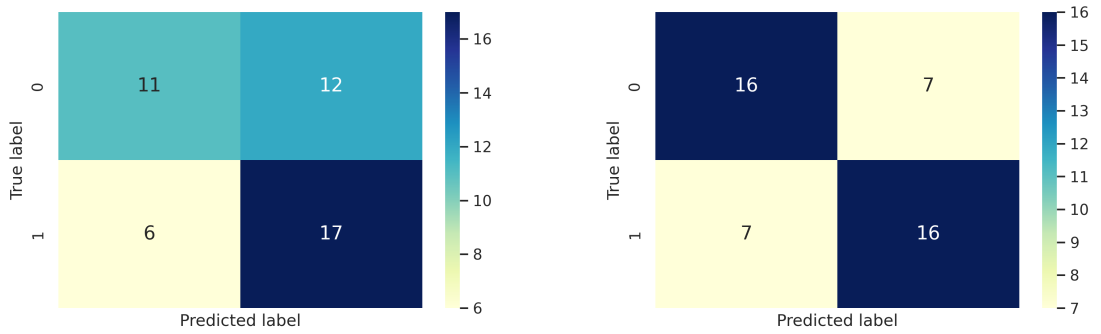
7.5.1.2 Ensemble of CNNs vs. Snapshot Learning

This experiment consisted of a performance comparison of an ensemble of five CNNs versus an ensemble of five snapshots from the same CNN. The CNNs have the same main architecture described in Section 7.4 but different random initializations. We decided to test at the epoch where the models reach a training accuracy of 80% to avoid overfitting. The results are presented in Table 7.2. The results are roughly equivalent to the ensemble of 5 fully trained models with much less training time.

7.5.1.3 Comparison of 3D T1CE vs. 2D T1CE MRIs

In this experiment, due to the high computational costs and time required for 3D image processing, we investigated whether the third dimension added significant value over the 2D analysis of the images. We decomposed each 3D scan into 155 separate slices. Then, we manually looked at each slice and chose the one with the largest visible tumor area.

Consequently, the new training set in this experiment consists of 163 images of size $240 \times 240 \times 1$. In order to increase the size of the dataset, image augmentation was also done using the same transformations explained in Section 7.2. We created a 2D version of the main CNN and kept the same architecture and parameters. For this dataset, after observing the training accuracy curve, we noticed that the model reaches 80% training accuracy quickly (epoch 28). Therefore, we decided to go back 5 steps from when the training accuracy reaches 90% (epoch 59) and then proceed back to choose every fifth epoch as a snapshot. Thus, epochs 59, 54, 49, 44, and 39 were used. The performance results of the snapshots ensemble using the 2D CNN model are compared to the 3D CNN in Figure 7.3.



(a) 2D (b) 3D

Figure 7.3: Confusion Matrix Comparison of 3D vs 2D CNN Using the T1CE Sequence

7.5.2 Combination of Multi-Sequence Data for Survival Prediction

In this section, we describe two methods attempted in order to merge the three MRI sequences (T1CE, T2, and FLAIR) to create a multi-sequence dataset for our model.

7.5.2.1 T1CE, T2, and FLAIR, Ensemble of Ensembles

Before merging the three sequences, we first trained our CNN model that has the same settings and using the same training procedure described in Section 7.4, this time with the

other two sequences (T2 and FLAIR) as inputs instead of T1CE. Results are reported for each sequence individually.

To create a combined decision, we used an ensemble of ensembles approach in which 5 snapshots from each of the three sequences T1CE, T2, and FLAIR together participate to build an outcome prediction. We saved 5 snapshots of individually trained CNNs on each of the three sequences. For the choice of snapshots, we used the same strategy explained in Section 7.5.1.1. Therefore, we have 5 snapshots of the CNN trained with T1CE sequence, 5 snapshots of the CNN trained with T2, and 5 snapshots of the CNN trained with FLAIR. In a given ensemble, for each sample, a snapshot from each of the three sequences is used to vote to get a class. The three snapshots are from the same position in the snapshot order (e.g., three snapshots at position 1, first ensemble) are voted. Then, three snapshots at position two, etc. This results in 5-voted decisions. We then combine the 5 voted decisions again using majority voting to form the outcome of that sample. Figure 7.5 compares the accuracy results of CNNs trained with individual sequences vs the ensemble of ensembles combination.

7.5.2.2 Multi-Branch CNN with Multi-Sequence MRIs

In this experiment, we combined the three sequences (T1CE, T2, and FLAIR) to train a multi-branch 3D CNN and test it on the multi-sequence MRI test set. The architecture of the CNN is illustrated in Figure 7.4.

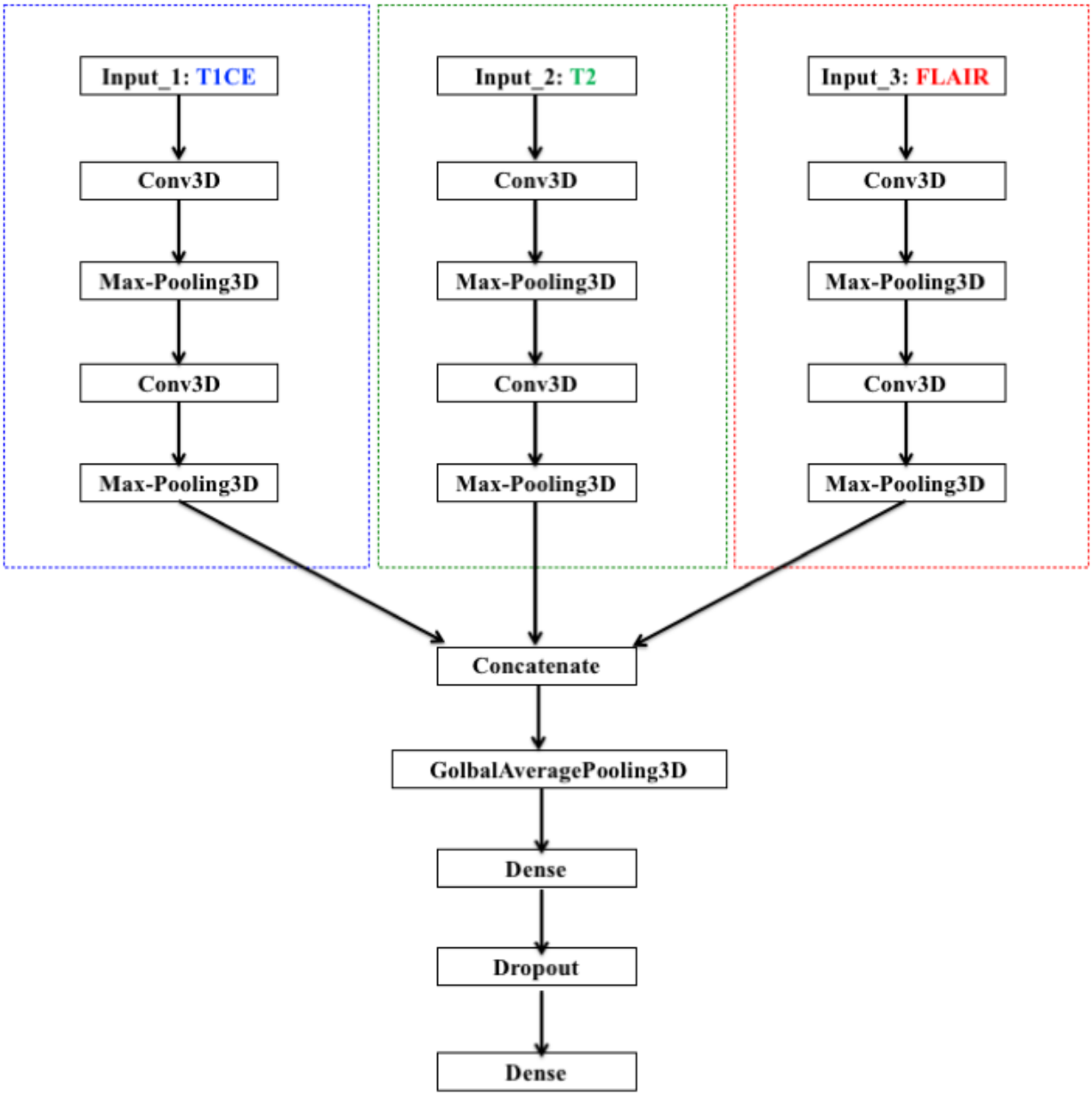


Figure 7.4: Multi-branch CNN Architecture

Separate CNN models operate on each sequence where each CNN has two 3D convolutional layers each followed by a max-pooling layer. Then, the results from the three models are concatenated for interpretation and ultimate prediction. The CNN was evaluated using a

snapshot ensemble with max voting as explained in Section V. Here, we used a lower learning rate (10^{-4}) than in the previous experiments. Thus, the model makes small changes between epochs. Therefore, we decided to enlarge the spacing between chosen epochs. We go back 5 steps from when the training accuracy reaches 90% (epoch 183) and then proceed back to choose every 25th epoch as a snapshot (epochs 183, 158, 133, 108, and 83).

The performance results of the multi-branch CNN are presented in Figure 7.5 and compared to the performance outcome of each sequence and the first ensemble method described in Section 7.5.2.1.

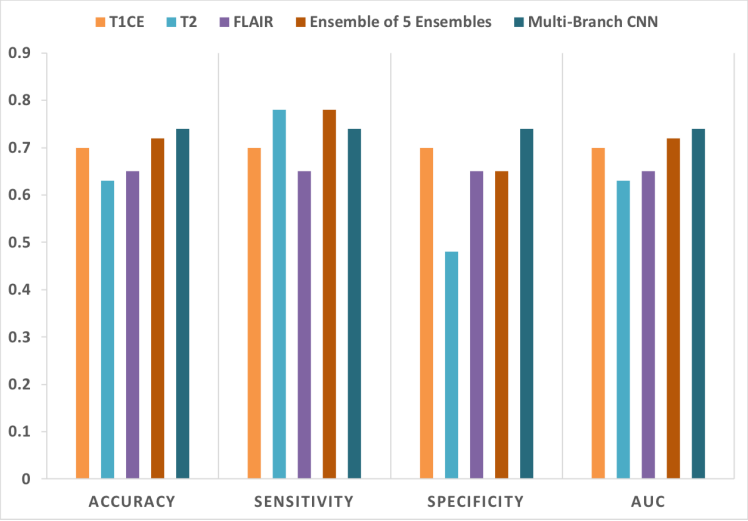


Figure 7.5: Performance Comparison of the Multi-branch CNN vs Individual Sequences vs Ensemble of the Five Voted Ensembles Method

7.5.3 Comparison with Other Methods

To further assess the prediction performance of our method, we compared our approach to the method proposed in [29]. In their paper, the authors used the BraTS 2017 training dataset with a threshold time of 18 months to classify cases into short-term and long-term survival. They extracted volumetric and location features from the 163 input images and trained a logistic regression classifier. They achieved an accuracy of 69% using 5-fold cross-

validation. The BraTS 2017 training dataset is a subset of the 2019 training dataset used here. To compare our results with their method, we used the same 163 cases and we also used an 18-month threshold to divide the data into two classes, short- and long-term survival. We then augmented the data and partitioned it to do the five-fold cross-validation experiment. Our snapshot ensemble was used on each of the five folds. The average accuracy of the five folds was 6% higher than in [29] at 75%.

7.6 Discussion

As demonstrated in Table 7.2, the overall survival prediction performance of the ensemble of CNNs method outperformed individual CNNs. It was 67% accurate whereas the maximum we could get if we only use one individual CNN was 63%.

From Table 7.1, a snapshot ensemble indeed can perform as well as an ensemble of multiple CNNs and is faster and less expensive to train. Clearly, training five separate models (sequentially) is five times slower than saving snapshots of a single model. Using a snapshots ensemble, we achieved a prediction accuracy of 70%. We can also see that the performance of an ensemble is sometimes the same as only one snapshot, but it usually outperforms individual snapshots. Generally, if it can not improve the performance, it does not cause a clear decrease in accuracy.

As seen in Figure 7.3, comparing these results to the performance of a CNN trained using 2D images, we can conclude that even though 3D processing is expensive and time-consuming, it brings relevant extra information to increase the model's performance from 61% to 70%.

Figure 7.5 summarizes the performance results of models trained using T1CE, T2, and FLAIR sequences standalone along with the results of the two attempted combining methods. The accuracy of the voting ensemble for T2 was 63% and for FLAIR was 65%. The 70% prediction accuracy using the T1CE sequence was slightly higher. This result probably reflects the sensitivity of T1CE to the size of the tumor core area, which is probably the

most important biological predictor of overall survival in GBM patients. Experiments with combining the three sequences together were done. The first method of combination we explored was voting an ensemble of sequences. This ensemble achieved a 72% prediction accuracy. The second combining method involved creating a multi-branch CNN where each sequence was sent as input to a branch then the results were combined in the final layer. The prediction accuracy of the ensemble using this multi-branch CNN reached 74%.

The result of the two experiments with sequence combination demonstrates the potential power in the combination of multiple sequences where each contributes unique information relevant to the overall survival prediction. We can also conclude that all the tumor areas are important to determine the survival of the patient. Though perhaps the tumor core area contains the most information.

Since the BraTS testing dataset is not released by the BraTS challenge team and the validation set is only available at the time of the challenge, we were unable to directly compare our method to those applied to the BraTS test/validation set. However, our prediction accuracy of 74% on unseen testing set from BraTS 2019 suggests good generalization ability to other datasets. In addition, in the experiment conducted in Section 7.5.3, our method achieved a prediction accuracy of 75%, thus outperforming the approach proposed in [29], which had 69% accuracy on the same dataset.

7.7 Summary

In summary, we demonstrated an overall survival prediction for glioblastoma patients at 74% accuracy using a multi-branch 3D convolutional neural network, where each branch was a different MR image sequence. The system classified patients into two classes (long- and short-term survival). In comparison to the same data with the best-known predictor, our approach was 6% more accurate. An important contribution of this work is the effective use of snapshot ensembles in a novel fashion toward a solution to the limited availability

of labeled images in many medical imaging datasets. Snapshots are a fast, low-cost way to generate an ensemble of classifiers.

Chapter 8: Shortcut Learning Challenge in CNN-based COVID-19 Diagnosis Systems[4]

8.1 Problem Statement

In the current COVID-19 era, a relatively large set of papers have been published proposing the use of deep convolutional neural networks to perform diagnosis or triage of COVID-19 from chest X-ray scans. The studies are reporting very high-performance results (more than 90% accuracy) which could be misleading and overestimated. In most of those papers authors used subsets of train/validation/test from the same data source, others opted for a cross-validation evaluation method, which also mixed train/validation/test sources. Here, we show how the use of the same sources in train/test sets leads to the high accuracy that these models have achieved. In addition, the majority of those papers used low-quality images, some are extracted from PDF files of scientific publications. This approach increases the likelihood of introducing image processing artifacts, which further increases the risk of learning confounders rather than real pathologic features. In this work, we discovered that none of the currently published papers with high performance reported testing on samples from truly unseen data sources. Those which did, have noticed a significant performance drop (as much as 20%) when testing on unseen sources indicating a failure to generalize. When a COVID-19 chest X-Ray model has high accuracy under familiar settings (internal testing) and a big drop in accuracy under slightly different circumstances (external testing) we suspect it is using shortcuts for diagnosis, a problem known as shortcut learning. In this

[4]Parts of this chapter were published in K. B. Ahmed, G. M. Goldgof, R. Paul, D. B. Goldgof and L. O. Hall, "Discovery of a Generalization Gap of Convolutional Neural Networks on COVID-19 X-Rays Classification," in IEEE Access, vol. 9, pp. 72970-72979, 2021, doi: 10.1109/ACCESS.2021.3079716.
The authors hold copyright rights under a CC BY license.

case, shortcuts/confounders are undesired and unintended features that are not medically relevant that the model relies on to perform COVID-19 diagnosis. This indicates that such models need further assessment/development before they can be broadly clinically deployed. To evaluate this challenge, we worked with 655 chest X-rays of patients diagnosed with COVID-19 and a set of 1,069 chest X-rays of patients diagnosed with other pneumonia that predates the emergence of COVID-19. An example of fine-tuning to improve performance at a new site is given.

8.2 Data Preprocessing

In our previous work [52], we used COVID-19 images from three main sources [33], [116] and [130]. Note that these sources were and still are largely used in the majority of research papers related to the prediction of COVID-19 from X-rays. We later identified several potential problems with these sources. Many of these images are extracted from PDF paper publications, are pre-processed with unknown methods, down-sampled, and are 3 channels (color). The exact source of the image is not always known and the stage of the disease is unknown.

For the COVID-19 class, three sources were used in this work, BIMCV-COVID-19+ (Spain) [38], COVID-19-AR (USA) [40] and V2-COV19-NII (Germany) [151]. For readability, we will label each dataset both by its name and also its country of origin, since the names of each dataset are similar and may confuse the reader.

Each patient in the datasets had different X-ray views (Lateral, AP, or PA) and had multiple sessions of X-rays to assess the disease progress. Radiology reports and PCR test results were included in both BIMCV COVID-19+ and COVID-19-AR (USA) sources. We selected patients with AP and PA views. After translating and reading all the sessions reports coupled with PCR results, only one session per patient was chosen based on the disease stage. We picked the session with a positive PCR result and the most severe stage.

As discussed in [125], using raw data in its original format is recommended. In our study, we included all raw COVID-19 datasets that were available to us (COVID-19-AR (USA) [40] and V2-COV19-NII (Germany) [151]). To avoid creating confounders based on the CXR view, we used frontal view(AP/PA) CXRs in both classes. To assure the validity of the ground truth, we made sure not to rely only on a positive RT-PCR but also on the associated CXR report confirming and supporting the test results.

For the non-COVID-19 class, pneumonia cases were used because they are expected to be the hardest CXR images to differentiate from COVID-19 and because a use case for deep learned models to detect COVID-19 will be for patients that have some lung involvement. The pneumonia class data came from 3 sources: (i) the National Institute of Health (NIH) dataset [148], (ii) Chexpert dataset [64] and (iii) Padchest dataset [22]. The NIH and Chexpert datasets had pneumonia X-ray images with multiple labels (various lung disease conditions), but for simplicity, we chose the cases that had only one label (pneumonia). Only X-rays with a frontal view (AP or PA) were used in this work. Three samples of COVID-19 and three pneumonia X-ray images are shown in Fig 8.1.

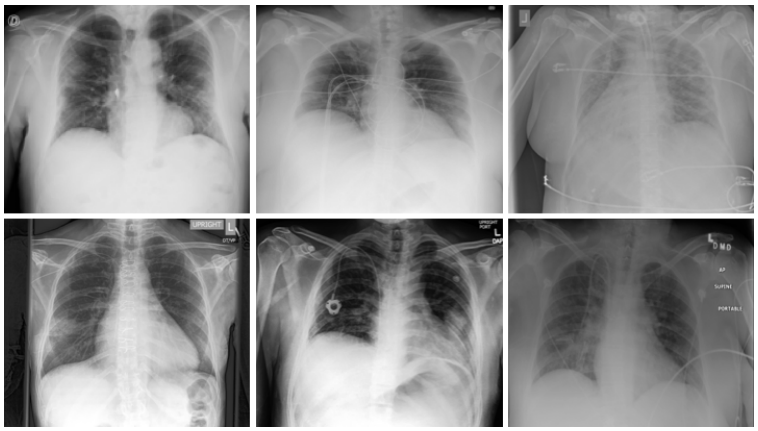


Figure 8.1: Samples of the Input X-rays: Top: COVID-19 Cases, Bottom: Pneumonia Cases

As a first step, we normalized all the images to 8 bits PNG format in the [0- 255] range. The images were originally 1 grayscale channel, we duplicated them to 3 channels for use with pre-trained deep neural networks. The reason behind this is that Resnet50, the model that we utilized as a base model was pre-trained on 8-bit color images. To reduce the bias that might be introduced by the noise present around the corners of the images (dates, letters, arrows ...etc), we automatically segmented the lung field and cropped the lung area based on a generated mask. We used a UNET model pre-trained by [120] on a collection of CXRs with lung masks. The model generates 256x256 masks. We adapted their open source code [120] to crop the image to obtain bounding boxes containing the lung area based on the generated masks. We resized the masks to the original image size. We then added the criteria to reject some of the failed crops based on the generated mask size. If the size of the cropped image is less than half of the size of the original image or if the generated mask is completely blank then we do not include it in the training or test set. Fig 8.2 illustrates the steps of mask generation and lung ROI cropping.

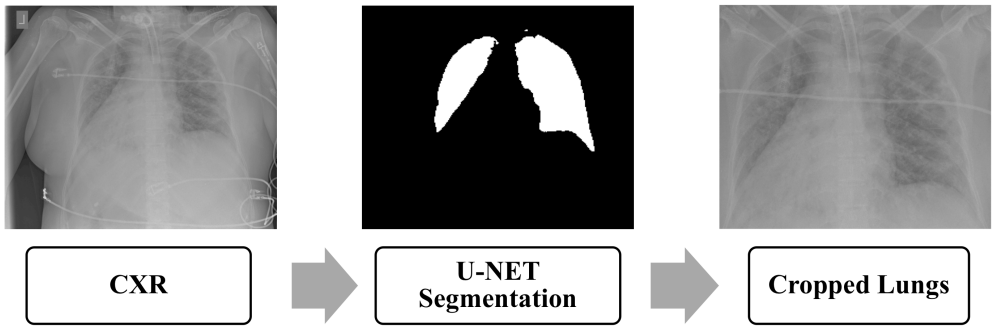


Figure 8.2: Pipeline of Lung ROI Cropping

For data augmentation, 2, 4, -2, and -4 degree rotations were applied and horizontal flipping was done followed by the same set of rotations. By doing so, we generated 10 times (original images, horizontal flipping, 4 sets of rotated images each from original and flipped

images) more images than the original data for training. We chose a small rotation angle as X-rays are typically not rotated much.

8.3 Model Training

In this study, pre-trained ResNet50 [56] was fine-tuned. As a base model, we used the convolutional layers pre-trained on ImageNet and removed the fully connected layers of Resnet50. Global Average pooling was applied after the last convolutional layer of the base model and a new dense layer of 64 units with ReLU activation function was added. Then, a Dense layer with 1 output with sigmoid activation was added using dropout with a 0.5 probability. All the layers of the base model were frozen during the fine-tuning procedure except the Batch Normalization layer to update the mean and variance statistics of the new dataset (X-rays). The total number of trainable parameters was 184K, which was helpful for training with a small dataset. The architecture is summarized in Table 8.1.

Table 8.1: ResNet50 Fine-Tuned Architecture

Resnet50
Output from base model
Global Average Pooling
Fully Connected (64), ReLU
Dropout=0.5
Batch Normalization
Fully Connected (1), Sigmoid
Trainable parameters: 184K

The model was fine-tuned using the Adam [74] optimizer for learning with binary-cross-entropy as the loss function and a learning rate of 10^{-4}. We set the maximum number of epochs to 200, but we stopped the training process when the validation accuracy did not improve for 5 consecutive epochs. The validation accuracy reaches its highest value of 97% at epoch 100.

8.4 Experimental Results and Discussion

In this section, we investigate the robustness and generalization of deep convolutional neural networks (CNNs) in differentiating between the COVID-19 positive and negative class (non-COVID-19 pneumonia). For this purpose, we did a baseline experiment similar to what the reviewed papers have conducted. CNN models were trained on 434 COVID-19 and 430 pneumonia chest X-ray images randomly selected from all the sources that we introduced in the previous section. For validation, a random set of 40 COVID-19 and 46 pneumonia cases were utilized. We then tested on unseen left-out data of 79 COVID-19 (30 from BIMCV COVID-19+, 10 from COVID-19-AR (USA), and 39 from V2-COV19-NII (Germany)) cases and 303 pneumonia (51 from NIH and 252 from Chexpert) samples. For comparison purposes, we used another fine-tuning methodology where we unfroze some of the base model convolutional layers. Thus, the weights of these layers get updated during the training process. In particular, we unfroze the last two convolutional layers of Resnet50. We also used the two fine-tuning strategies to train another model with VGG-16 as the base model, pretrained on ImageNet. The testing results are summarized in Table 8.2.

Table 8.2: Performance Results of Training on a Mixture of All Data Sources and Testing on Held-out Test Data from the Same Sources: Finetune1: Freeze All Base Model Layers, Finetune2: Unfreeze the Last 2 Convolutional Layers

CNN	Accuracy	Sensitivity	Specificity	AUC
Resnet50-Finetune1	**98.1%**	**96.2%**	**98.7%**	**0.997**
Resnet50-Finetune2	97%	95%	97.7%	0.995
VGG-16-Finetune1	88.2%	87.3%	88.4%	0.96
VGG-16-Finetune2	95.5%	93.7%	96%	0.98

As expected, and as seen in Table 8.2, both models and both fine-tuning methods were able to achieve high performance on an unseen test set from the same sources. In order to investigate the generalization of these models (which is the main focus of this chapter), we evaluated their performance on external data sources for which there were no examples in

the training data. Experiments were done with training data from just one source per class and testing data from sources not used in training (see Fig. 8.3). The Resnet-50 architecture with the Finetune1 method was used for the rest of the experiments in this chapter.

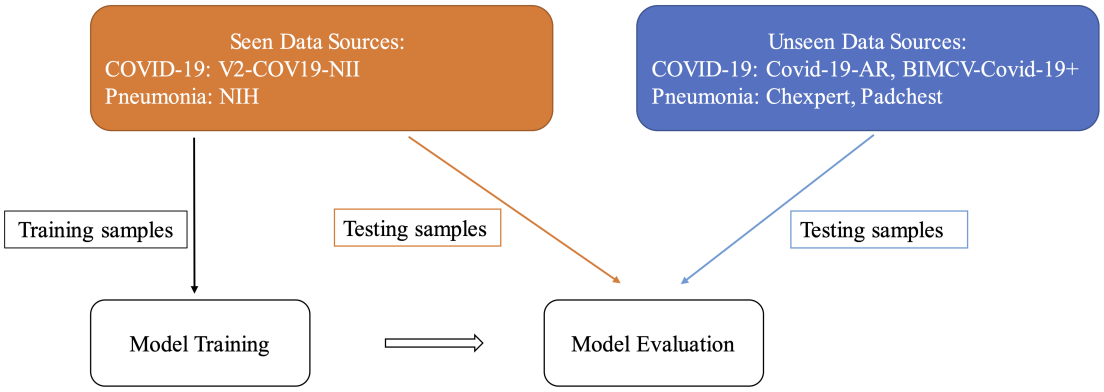


Figure 8.3: Workflow of the Generalization Gap Experiments

The data overview table at the top of Fig. 8.4 shows details of data splits used in our experiments with the total number of samples used for the training and testing phases. As seen in the table, we first trained the model using the V2-COV19-NII (Germany) data source for the COVID-19 class and NIH for pneumonia (Data Split 1). We then compared the AUC results on a randomly held-out subset from the seen sources (V2-COV19-NII (Germany) and NIH) versus unseen sources.

As seen in the AUC graph in Fig. 8.4 to the left, the model achieves perfect results (AUC = 1.00) on left-out test samples from seen sources (images from the same dataset source on which the model was trained), but it performs poorly (AUC = 0.38) on images from unseen sources. Using the McNemar's test [96], we calculated a p-value of $1.78e^{-70}$ which is way lower than the significance threshold, $alpha = 0.01$. There is a significant difference between the model's performance on seen vs unseen sources with 99% confidence.

Clearly, the model was unable to generalize well to new data sources, which might indicate that the model is relying on confounding information related to the data sources instead of

Class	Data Source	Total	Data Split 1: V2-COV19-NII / NIH			Data Split 2: BIMCV-COVID-19+ / PADCHEST		
			Train	Seen Test	Unseen Test	Train	Seen Test	Unseen Test
COVID-19	V2-COV19-NII	243	233	10	-	-	-	155
	COVID-19-AR	77	-	-	75	-	-	75
	BIMCV-COVID-19+	335	-	-	212	272	38	-
Pneumonia	NIH	303	283	20	-	-	-	205
	PADCHEST	352	-	-	246	293	59	-
	CHEXPERT	414	-	-	414	-	-	414

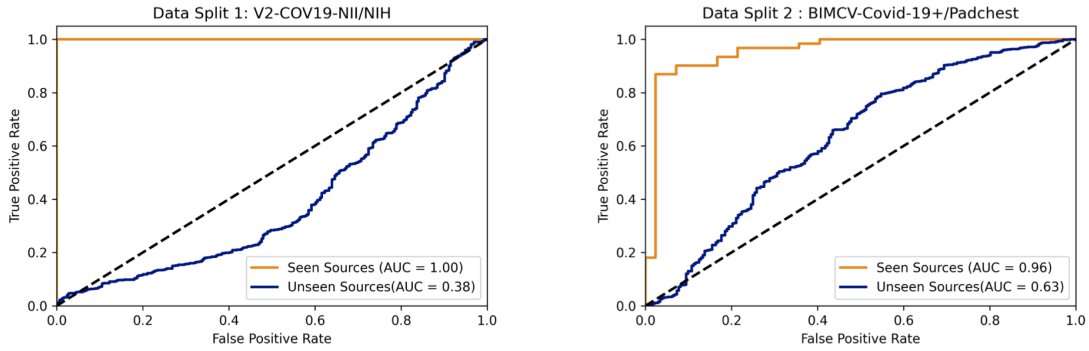


Figure 8.4: Overview of Data Splits (Top) and Comparison of AUC Results (Bottom) on Seen vs. Unseen Test Data Sources

the real underlying pathology of COVID-19. The fact that its performance (AUC=0.38) is less than AUC=0.5 (worse than random), strongly suggests that the model is relying on confounding information. The perfect score on the data from the seen dataset source also hints at confounders, as it is unlikely that any algorithm could perfectly distinguish COVID-19 positive versus pneumonia patients based on lung findings alone. On the other hand, it is highly likely that perfect classification could be performed based on the features related to the data source. To give a human analogy, a radiologist would find it easier to classify COVID-19+ versus COVID-19-negative chest X-rays by looking at the year in which the image was taken (pre-2020 versus post), rather than by looking at the image itself.

In an experiment to see if a model built with data from similar sources for the two classes (COVID-19 and Pneumonia) can result in more general models, we chose a second data split (data split 2) with BIMCV-COVID-19+ (Spain) data as the source for COVID-19 and Padchest for Pneumonia. These two sources come from the same regional healthcare system (Valencia, Spain), both were prepared by the same team and underwent the same

data pre-processing. We anticipated that reducing the differences between classes in terms of image normalization, hospitals, scanners, image acquisition protocols, etc would enable the model to only concentrate on learning medically-relevant markers of COVID-19 instead of source-specific confounders. Details about data split 2 can be found in the data overview table on top of Fig. 8.4.

The results in the AUC graph in Fig 8.4 to the right show that the model still exhibits high performance on seen sources but generalizes poorly to external sources. Using the McNemar's test [96], we calculated a p-value of $5.39e^{-82}$ which is way lower $alpha = 0.01$. Therefore there is a statistically significant difference between the model's performance on seen vs unseen sources with 99% confidence.

We can see that even having both classes from the same hospital system did not prevent the model from learning data-source-specific confounders. However, in contrast to the model trained on Data Split 1, this model has slightly worse performance on data from seen sources (AUC=0.96 for data split 2 vs AUC=1.00 for data split 1) and better performance on data from unseen sources (AUC=0.63 for data split 2 vs AUC=0.38 for data split 1). Notably, the second model's performance is better than random (AUC>0.5). This suggests that the algorithm may have learned some clinically salient features, although once again, the majority of its performance appears to be based on confounders.

We can also observe that it is possible that confounders found in some data sources can generalize across sources. For example when training using the BIMCV-COVID-19+ (Spain) data source, the model had an accuracy of 88% on COVID-19-AR (USA), which is an unseen source. However when training using V2-COV19-NII (Germany) data source, the model only achieved an accuracy of 68% on this same unseen source (COVID-19-AR (USA)).

As a possible solution, we tried fine-tuning the trained model from the previous experiment (data split 1) using multiple sources for each class, using a subset of 80 samples from BIMCV-COVID-19+ (Spain) for the COVID-19 class and a subset of 80 samples from Chexpert for the pneumonia class. Both these sources were considered unseen in the experiment

with data split 1 described in the data overview table on top of Fig. 8.4. As seen in Table 8.3, fine-tuning with subsets from unseen sources improves the model's overall performance on those sources. We hypothesize that fine-tuning helps the model ignore noisy features and data source-related confounders and instead concentrate on learning meaningful and robust features.

To investigate what the model is relying on this time, we applied the Grad-Cam algorithm [127] to test images to find highlighted activation areas. This is a method used to see which parts of the image are most influencing the algorithm's classification. We would expect a classifier relying on true pathologic features to primarily be relying on pixels from the lung fields, whereas a spurious classifier would rely on pixels from regions of the image irrelevant to diagnosis. The results were inconclusive (see Table A.1 of the Appendix A). Therefore, we cannot affirm whether the model is still relying on shortcuts/confounders to make decisions. This experimental result shows that a model could be adapted to work locally. Still to be shown is that it learns medically relevant features.

Table 8.3: Accuracy Results of Finetuning a Model Built on Multiple Sources from Both Classes to Adapt It to Work Locally

Class	Data Source	Before	After
COVID-19	BIMCV-COVID-19+ (Spain)	51%	98%
Pneumonia	Chexpert	12%	94%

8.5 Summary

In this chapter, we demonstrated that deep learning models can leverage data source-specific shortcuts to differentiate between COVID-19 and pneumonia labels. While we eliminated many shortcuts from earlier work, such as those related to large age discrepancies between populations (pediatric vs adult), image post-processing artifacts introduced by working from low-resolution PDF images, and positioning artifacts by pre-segmenting

and cropping the lungs, we still saw that deep-learning models were able to learn using data-source specific shortcuts. Several hypotheses may be considered as to the nature of these shortcuts. These shortcuts may be introduced as a result of differences in X-ray procedures due to the patient's clinical severity or patient control procedures. For instance, differences in disease severity may impact patient positioning (standing for ambulatory or emergency department patients vs supine for admitted and ICU patients). In addition, if a particular X-ray machine whose signature is learnable is always used for COVID-19 patients because it is in a dedicated COVID-19 ward, this would be another method to determine the class in a non-generalizable way.

Using datasets that underwent different pre-processing methods across classes can encourage the model to differentiate classes based on the pre-processing, which is an undesirable outcome. Thus, training the model on a dataset of raw data coming from many sources may provide a general classifier. Even within the same hospital, one must still check to be sure that something approximating what a human would use to differentiate cases is learned.

That being said, using a deep learning classifier trained on positive and negative datasets from the same hospital system, having undergone similar data processing, we were able to train a classifier that performed better than random on chest X-rays from unseen data sources, albeit modestly. Tuning with data from unseen sources provided much-improved performance. This suggests that this classification problem may eventually be solvable using deep learning models. However, the theoretical limit of COVID-19 diagnosis, based solely on chest X-ray remains unknown, and consequently, also the maximum expected AUC of any machine learning algorithm. Unlike other classification problems that we know can be performed with high accuracy by radiologists, radiologists do not routinely or accurately diagnose COVID-19 by chest X-ray alone. However, an imperfect classifier that has learned features that are not shortcuts may be combined with other clinical data to create highly accurate classifiers, and as such, this area warrants further inquiry.

Our results suggest that, for at least this medical imaging problem, when deep learning is involved it is important to have data from unseen sources (pre-processed in the same way) included in a test set. If there are no unseen sources available, careful investigation is necessary to ensure that what is learned both generalizes and is germane. It points out that future investigation into finding/focusing on features that generalize across sources is quite important. This will enable an evaluation of how helpful CXRs can truly be for COVID-19 diagnosis. All data and code used in this study are available on Github [2]

[2]https://github.com/kbenahmed89/Pretrained-CNN-For-Covid-19-Prediction-from-Automatically-Lung-ROI-Cropped-X-Rays.

Chapter 9: Human-in-the-Loop for Shortcut-Free Training Datasets of CNN-based COVID-19 Diagnosis Systems[5]

9.1 Problem Statement

Shortcuts exist in most data and rarely disappear by adding more data [48]. Instead, we demonstrate that modifying the training data to block specific shortcuts is a potential solution. Here, we again focus on the generalization challenge of deep learning based models trained to recognize COVID-19 infection from chest X-ray images. Furthermore, we experimentally show effective solutions to mitigate shortcut learning including lung field cropping, histogram equalization, and Gaussian noise-based augmentation. The issues with convolutional neural networks discussed here generally apply to other imaging modalities and recognition problems.

9.2 Data Preprocessing

To assure the validity of the ground truth, we made sure not to rely only on a positive RT-PCR but also, when possible, on the associated CXR report confirming and supporting the test results. Visual assessment of the imaging data allowed us to remove multiple cases with a lateral view even though the corresponding metadata says it's frontal. Other metadata mistakes include X-ray projection and position, for example, an AP Portable X-ray can be mislabeled as PA. As DICOM information on the type of projection may not be always included, some data sources [22] used pre-trained models for automatic X-ray view deter-

[5]Parts of this chapter were published in K. Ben Ahmed, L. O. Hall, D. B. Goldgof and R. Fogarty, "Achieving Multisite Generalization for CNN-Based Disease Diagnosis Models by Mitigating Shortcut Learning," in IEEE Access, vol. 10, pp. 78726-78738, 2022, doi: 10.1109/ACCESS.2022.3193700.
The authors hold copyright rights under a CC BY license.

Table 9.1: Details of Data Source Used in This Work

Class	Data Source	Number of Images	Image Type
COVID-19(+)	BIMCV-COVID-19(+)	468	PNG
	V2-COV19-NII	243	NIFTII
	CC-CXRI-P(+)	513	JPG
	COVID-19-AR	77	DICOM
	COVIDGR	426	JPG
COVID-19(-)	NIH	607	PNG
	Chexpert	420	JPG
	Padchest	310	PNG
	BIMCV-COVID-19(-)	500	PNG
	CC-CXRI-P(-)	542	JPG

mination. Thus, some incorrect labeling can occur. Additionally, unlike the usual negative X-ray mode ("bones white"), some images were in a positive mode ("bones black"). To avoid any problems these images may cause to the model's performance, we inverted them to the conventional mode ("bones white"). To avoid creating confounders based on the CXR view, we used frontal view(AP/PA) CXRs in both classes.

As a first step to preparing the data for training, we normalized all the images to 8-bit PNG format in the [0- 255] range. The images were originally 1 grayscale channel, we duplicated them into 3 channels. The reason behind this is that Resnet50, our base model, was pre-trained on 8-bit color (3 channel) images of the Imagenet corpus. For data augmentation, 2, 4, -2, and -4 degree rotations and horizontal flipping were applied. We chose a small rotation angle as X-rays are typically not rotated much. The training data was standardized by subtracting the mean of all data and dividing it by its standard deviation.

Table 9.1 summarizes all the data sources used in this work. Note that ground truth labels were extracted from the metadata and radiology reports were included with all of the datasets.

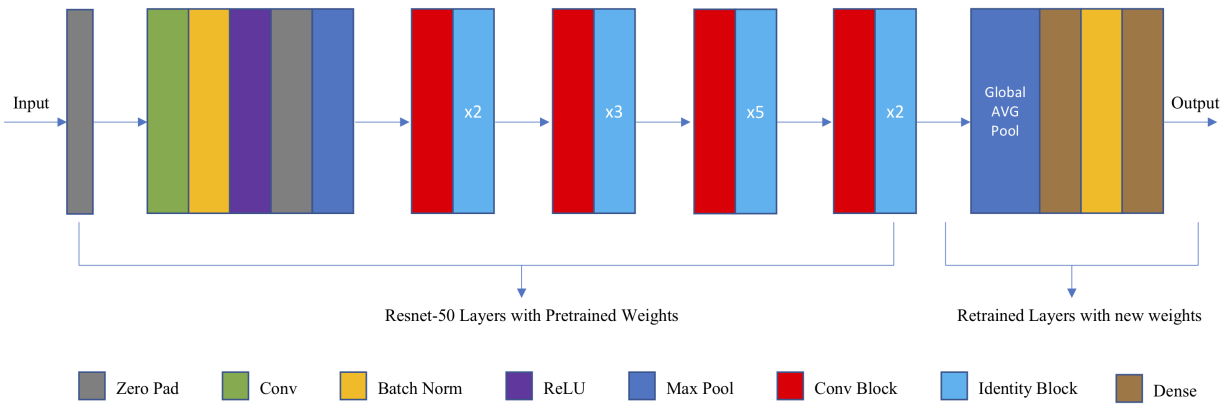


Figure 9.1: The Customized ResNet50 Architecture Deployed in the Proposed Approach

9.3 Baseline Model

In this chapter, we used ResNet50 pre-trained on ImageNet as a base model. We removed the fully connected layers of the base model. Then, global average pooling was applied after the last convolutional layer of the base model and a new dense layer of 64 units with the ReLU activation function was added. Then, a dense layer with 1 output and Sigmoid activation was added after applying batch normalization. The weights of the newly added layers were learned. However, all the layers of the base model were frozen during the fine-tuning procedure except the batch normalization layers to update the mean and variance statistics of the new dataset (X-rays). The total number of trainable parameters was 184K, which was helpful for training with a small dataset. Fig. 9.1 illustrates the architecture of the customized Resnet-50 used as a baseline model. Each convolution block has 3 convolution layers and each identity block also has 3 convolution layers. The model was trained using a cosine annealing cyclic learning rate with a maximum learning rate of 10^{-4}. The training was done for 1000 epochs and model snapshots were saved at the end of each of 20 cycles, which were 50 epochs long. Code and data are available on Github [3].

[3]https://github.com/kbenahmed89/COVID-19-Diagnosis-Model-External-Validation-Set

9.4 Overall Workflow

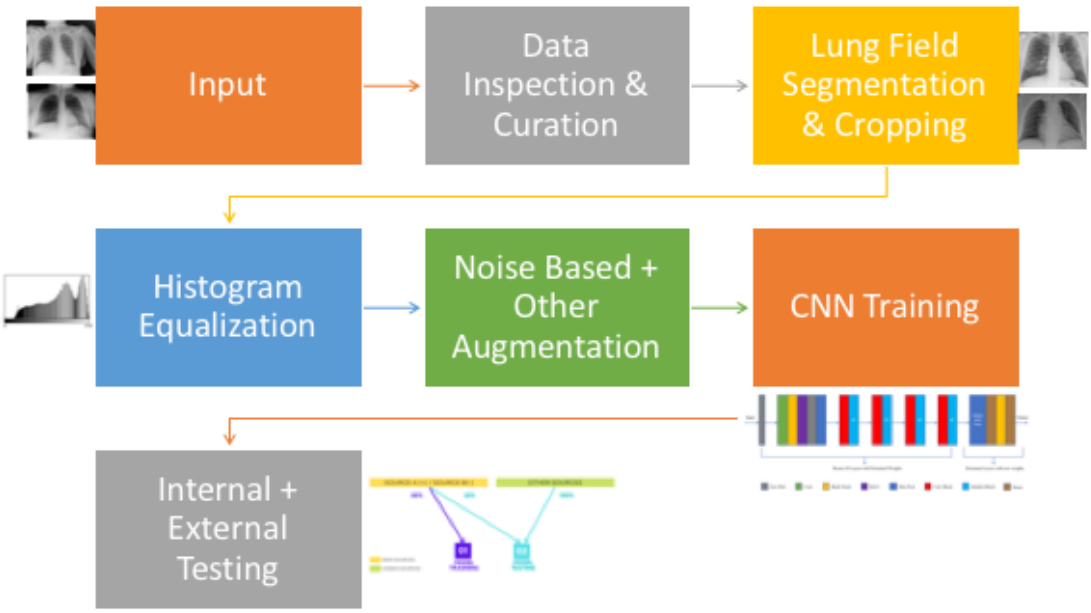


Figure 9.2: Overall Workflow of Suggested Approach to Mitigate Shortcut Learning

As illustrated in Fig. 9.2, the workflow starts with multiple sets of data coming from separate sources. The first step is the visual inspection of the images and the associated ground truth and metadata. Data curation is explained in more detail in Section 9.6.1. Then, we suggest lung field segmentation and cropping the inputs as a first pre-processing step to get rid of noise signals outside the lungs. Section 9.6.2 provides more experimental results of the impact of this step. Histogram equalization is a second pre-processing step that we recommend and the results of this step can be found in Section 9.6.3. To augment the size of the training set, we suggest using Gaussian noise augmentation in addition to the other traditional augmentations (rotation, flipping, etc.). Details of this type of augmentation are experimentally studied in Section 9.6.4. Once training data is prepared, we can send it as input to the CNN of choice to perform training. In this study, we used a modified Resnet-50 architecture, but any CNN model can be used to perform this task. Finally, a generalization

test, similar to the one we suggest in Section 9.5, needs to be performed to compare internal vs external source testing performance.

9.5 Generalization Test

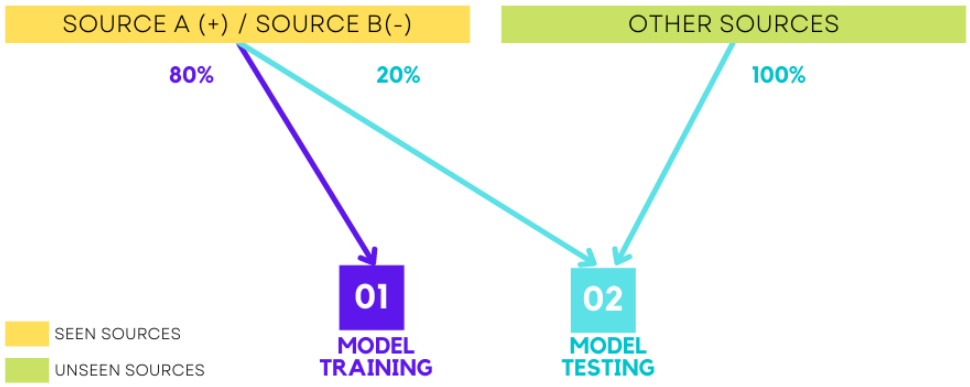


Figure 9.3: Generalization Test Experiment

In order to investigate the generalization ability of deep learned models (which is the main focus of this paper), evaluation was performed on external data sources from which there were no examples in the training data. Experiments were done with training data from just one source per class (for instance, source A for the COVID-19-positive class and source B for the COVID-19-negative class). To compare internal vs external testing, We split the data sources into seen (internal) and unseen (external). A subset (80%) from the seen data sources is used for model training. Then, we compare model testing performance using 1) the randomly chosen held-out subset (20%) from the seen sources versus 2) testing samples from unseen external data sources (see Fig. 9.3).

9.6 Experiments and Results

9.6.1 Data Inspection and Curation

One of the fundamental aspects to achieve a reliable contribution of AI in detecting COVID-19 from CXRs is the compilation of an adequate set of images in terms of both quality and quantity. Currently, due to data security and patient privacy, there is limited availability of open source datasets with the desired quality and quantity to build a diagnostic system with clinical value. Recently, researchers have begun to realize that convolutional neural networks trained to identify COVID-19 from CXRs may not be learning underlying diagnostic features, but also exploit confounding information. For example, it was shown that CNNs were able to predict where a radiograph was acquired (hospital system, hospital department) with extremely high accuracy [157]. In this section, we show the impact of some situational dataset-related shortcuts, we experimentally identified, to facilitate risk analysis and enable mitigation strategies while preparing training datasets.

9.6.1.1 X-ray Position

The PA erect view is the preferred imaging view in general, but if the patient is not able to stand up it is common to do an AP Portable view image. Most of the open source COVID-19 datasets are in AP Portable X-ray view due to the severe sickness of the patients which makes them unable to leave the bed to perform an upright PA scan. The use of bedside mobile CXR apparatus is frequently associated with more severe diseases (due to lack of patient mobility). If the training data in the majority or all of the negative cases consist of PA views (patient standing), the model can easily learn to make the classification based on the patient's position and use apparatus portability labels as a signal indicating the disease and its severity.

Results in [157] show that CNNs can ignore disease-related features and separate portable radiographs from the inpatient wards and emergency department with 100% accuracy using distinctive text indicating laterality and the use of a portable scanner.

9.6.1.2 Devices and Tubes

Table 9.2: External Testing Results of the Model Trained on CC-CXRI-P Dataset

Class	Data Source	Presence of Devices	Size	Accuracy
COVID-19(+)	V2-COV19-NII	Majority	142	0.96
	COVIDGR(+)	None	326	0.38
	COVID-19-AR	Majority	69	0.94
COVID-19(-)	Chexpert	Majority	205	0.23
	NIH	Some	197	0.6
	COVIDGR(-)	None	236	0.79

Another confounding factor might be the presence of medical devices like ventilation equipment or ECG cables. Many of the positive images for COVID-19 present intubated patients, with electrodes and their cables, pacemakers, among other potential markers. If there is an absence of medical devices in most or all the training samples in the negative class, the model can associate images with patient treatment instead of disease status. Experimentally, we used the CC-CXRI-P dataset [143] to train a model to differentiate COVID-19 cases from normal cases. The model achieved an internal testing accuracy of 99% on unseen samples from the same dataset. The extremely high performance led us to further investigate. A visual inspection of the subset of the CC-CXRI-P dataset used for this experiment shows that all patients in the positive class have medical devices visible and none of the patients in the negative class have any tubes or devices. Thus, suggesting a potential shortcut learning situation. To confirm the issue, we tested the model on data from unseen sites with and without the presence of devices.

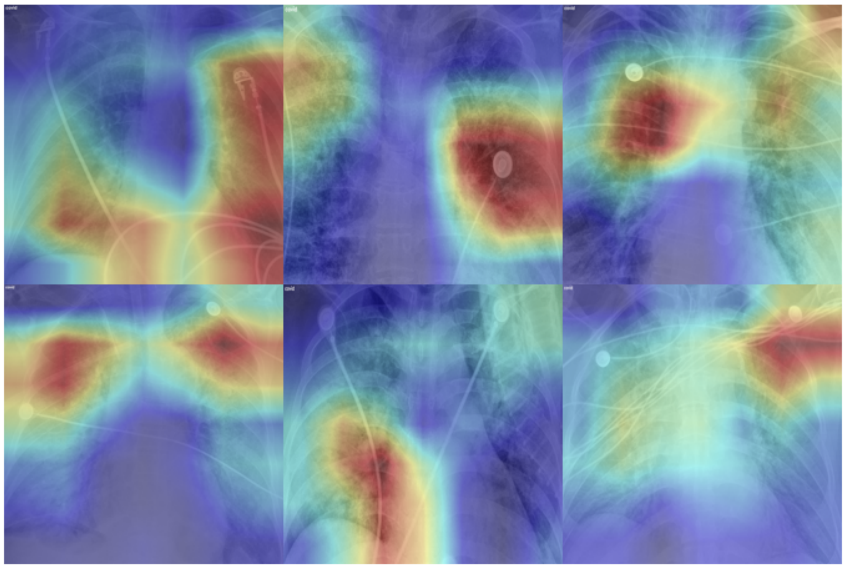


Figure 9.4: Gradcam Showing the Model Focusing on Medical Devices

Table. 9.2 shows the external testing results of the model by source. The results suggest that the model is associating COVID-19 positive infection with the presence of medical devices. The accuracy was high on COVID-19 positive sources where the majority of its patients had devices (V2-COV19-NII: 96% and COVID-19-AR: 94%). Whereas the accuracy was very low on a source where the patient had no visible devices (COVIDGR+ : 38%). Similarly, it can be seen that the model is associating COVID-19 negative infection with the absence of devices. The accuracy was good on negative sources that only had few or no devices visible (NIH: 60% and COVIDGR-: 79%) but very low on sources with devices (Chexpert: 23%).

To visually explain the decision of a CNN, Gradient-weighted Class Activation Mapping (GradCAM) [127] can be used. The GradCam visualization shown in Fig. 9.4 demonstrates that the highly activated areas (reddish and yellow) are located around tubes and devices.

Datasets that provide additional annotations on the presence of medical devices are needed. Alternatively, novel algorithms to automatically detect (and perhaps remove) the devices are required.

9.6.1.3 Rib Suppression

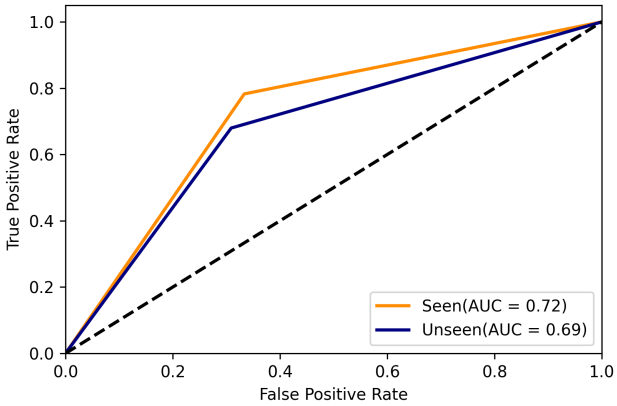
In addition to the confounders discussed above, another potential source of bias is the ribs. Ribs on some CXR datasets can generally appear to be more prominent than in other datasets. Therefore the appearance of the ribs can be a potential confounder that the model can rely on to make decisions. A study [60] shows that rib suppression improves model generalization when identifying lung nodules from X-rays. Thus, it may improve the COVID-19 classifier generalization as well.

9.6.1.4 Both Classes from the Same Source

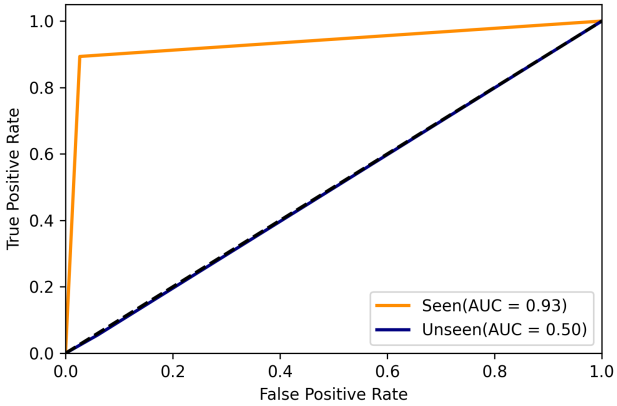
The heterogeneity of the images coming from different sources makes the CNN learn characteristics that are not themselves truly indicative of COVID-19. As a solution, we suggest designing the training data using both the COVID-19 positive and negative classes from one single source. In our attempt to experimentally demonstrate the impact of this solution, we built two training sets that correspond to two scenarios. The first is the case where both classes are from the same source and the second is the scenario where both classes come from different sources.

In the first scenario, we used the BIMCV-COVID-19 data source. This source published both positive and negative cases. We chose samples to include in the COVID-19 positive class if both a confirmed PCR test and radiological findings (visible in the X-ray and noted in the radiological report) are present. Whereas samples in the negative class are ones with a negative PCR test and no findings present in the CXR or documented in the reports. Fig. 9.5a compares the internal vs external performance of a model trained with data where both classes belong to the same source. Note that the performance is decent and the testing results are consistent across sites, both seen and unseen.

In the second scenario, for the positive class, we used the same samples from the experiment above (BIMCV-COVID-19+). Whereas for the negative class, we used samples from a different source (NIH). We picked images with the "no findings" label, which refers



(a) Both Classes from Same Source



(b) Both Classes from Different Sources

Figure 9.5: Comparison of Internal(Seen) vs External(Unseen) Test Results When Training with Data Where (a) Both Classes from Same Source vs (b) from Different Sources

to having no pulmonary diseases, to be consistent with the negative class used in the first scenario. Fig. 9.5b compares the internal vs external performance of a model trained with data where both classes belong to different sources. As seen in the figure, there is a big accuracy gap when testing on data from seen sites vs unseen ones. The model shows almost perfect performance on data from seen sources while it fails at classifying data from unseen sites. Thus, suggesting shortcut learning.

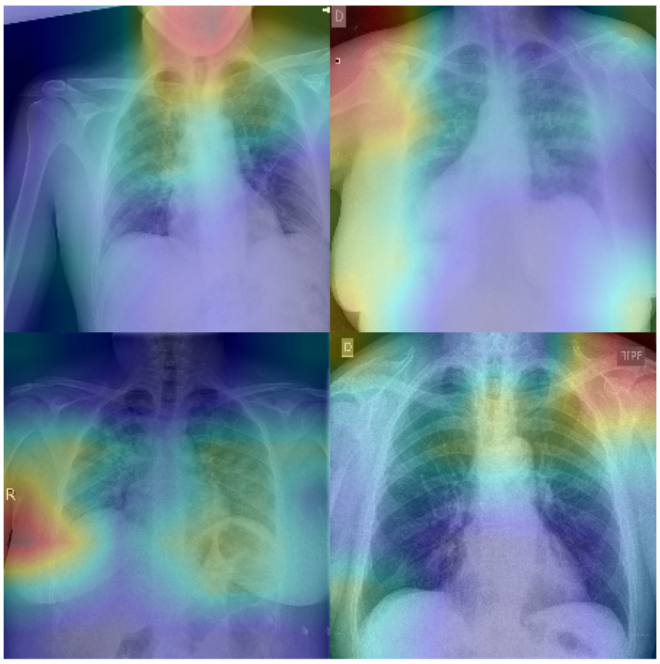
The results suggest that COVID-19 positive and COVID-19 negative classes should be consistently sourced and go through consistent image acquisition and pre-processing

pipelines. This will minimize systematic structural differences between the classes. Thus, eliminating bias. However, a shortcut could still be learned that will render a classifier ineffective when a local acquisition protocol changes. For the rest of this chapter, we focus on a typical case where having all classes from the same source is not possible.

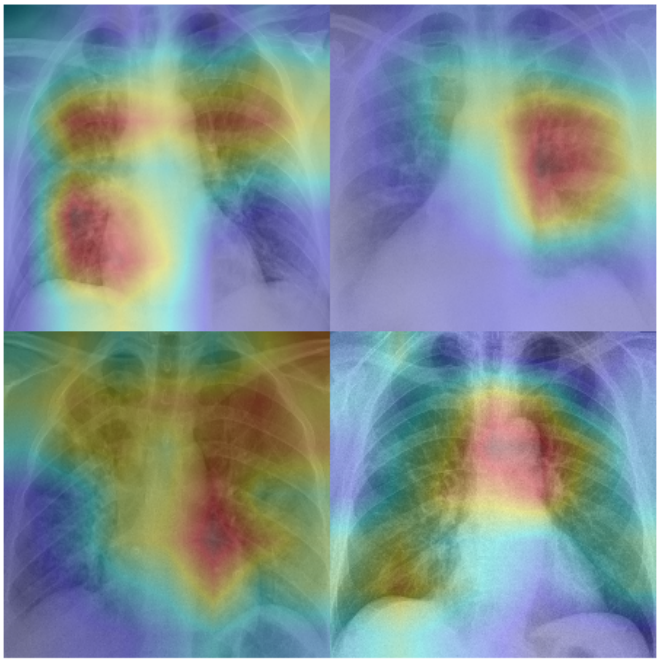
9.6.2 Lung Segmentation and Cropping

The first step to debiasing a deep learning model trained to identify COVID-19 from CXR images is the process of lung segmentation and cropping. The majority of the CXR images available online have markers present at the edges of the images (dates, projection labels, portability markers, arrows ...etc). Deep learning models may easily rely on those markers for learning and completely or partially ignore the lung area, which contains the desired information in this context. Thus, the learned model will be unreliable for clinical decision support even though it can achieve very high classification results. Therefore, we recommend lung field segmentation and cropping as the first step in data pre-processing and preparation. A model trained using cropped CXRs will be forced to focus on information in the lungs, rather than the burned-in annotations, when making decisions. This will increase its generalization and reliability. Fig. 9.6a shows a GradCam visualization of our baseline model, trained using original uncropped images. Heat maps that highlight the regions or pixels which had the highest impact on predicting the actual class are placed on top of the original images. The higher intensity in the heat map indicates the higher importance of the corresponding pixel in the decision. The heatmaps suggest that the model is activated by the noise areas and is ignoring the lung field. Fig. 9.6b shows more focus on the lung area and will be discussed in the proceeding.

The utility of lung field segmentation and cropping in medical image classification is supported in previous studies [138]. A comprehensive review of deep learning based medical image segmentation techniques may be found in [59].

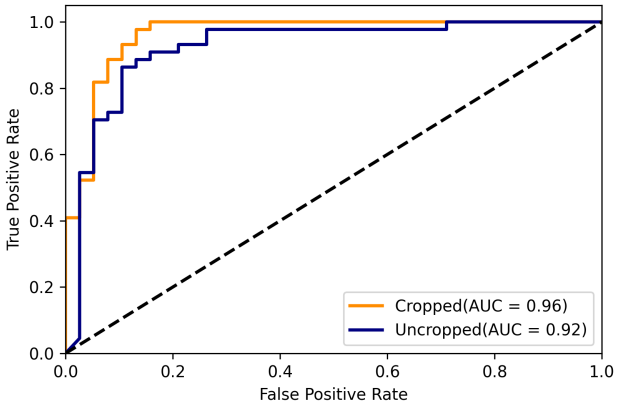


(a) GradCam Results When Using Original CXRs



(b) GradCam Results When Using Cropped CXRs

Figure 9.6: An Example of an Input CXR Before vs After Lung Cropping



(a) Internal Testing

	Precision	Recall	F1-score	Accuracy	AUC
Cropped	0.53	0.69	0.56	0.57	0.59
Uncropped	0.5	0.55	0.52	0.54	0.53

(b) External Testing

Figure 9.7: Performance Comparison of Models Trained with Cropped vs Uncropped Images

To assess the impact of lung segmentation and cropping we compare the performance of a model trained using uncropped images versus another model trained using cropped inputs. For the cropped inputs, we used the method described in the previous chapter, Section 8.2 and Figure 8.2 .

For model training, BIMCV-COVID-19+ data was used as the source for the COVID-19 positive class and Padchest for the COVID-19 negative class. A randomly chosen 20% of the data was held out for internal testing. For external testing of a model on unseen sites, we used samples from all the other sources not used in training.

We tested the last model's snapshot (20th or 1000th epoch). Fig. 9.7a compares the internal performance of models trained with uncropped vs cropped images. While Fig. 9.7b compares the external performance of the two models.

The results suggest that the model's accuracy is statistically significantly higher when images are cropped. Using the McNemar's test [96], we calculated a p-value of 0.02 which

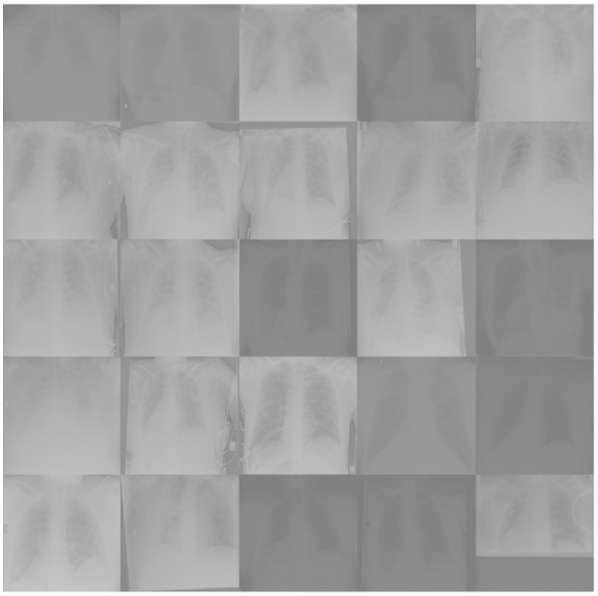
provides 98% confidence the difference between the model's performance on unseen sources before and after cropping is significant. Moreover, Fig. 9.6b shows GradCam results after lung field cropping. It can be seen that the model shifts focus to the lung area instead of the X-ray's corners where the noise was present. However, the significant drop in AUC shows that even after lung segmentation, a present shortcut is still influencing the classification model. Thus, more ways to avoid confounders must be evaluated to design a proper COVID-19 diagnosis system using CXR images.

9.6.3 Histogram Equalization

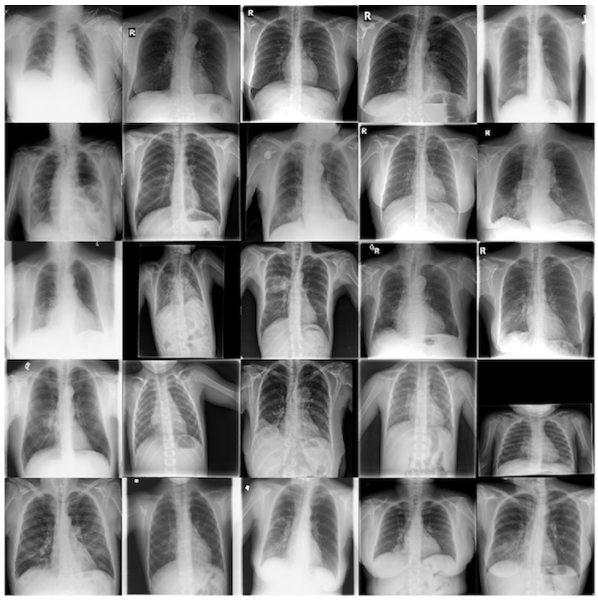
Systematic differences in image contrast between the datasets can result in biased model training and negatively impact model generalization. As seen in Fig. 9.8, datasets are easily distinguished from each other due to the obvious difference in the brightness and contrast of the images. CXR images can have variable contrast based on the technical calibration of the imaging acquisition equipment. Specifically, the energy of the primary beam and the possible application of techniques to reduce scatter radiation such as collimation, grids, or air gaps [107].

Histogram equalization processing has proven to be effective for normalization [111][90]. Therefore, in this study histogram equalization was used as a preprocessing step on all CXR sources. In Fig. 9.9, the result of equalizing shows consistent contrast between sources. All values above the 95% order statistic were clipped to 1.0 since some sources had bright annotations and other sources did not. This technique ensured that for every source the brightest CXR pixels were normalized to the brightest white.

In order to experimentally see the impact of histogram equalization, we compared training the baseline model with histogram equalized preprocessed inputs vs unprocessed images. We used the V2-COV19-NII for the COVID-19 positive class and Padchest for the COVID-19 negative class.

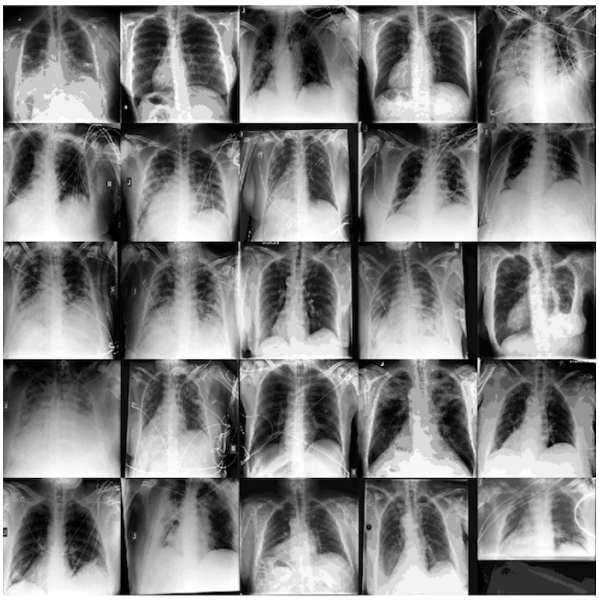


(a) V2-COV19-NII

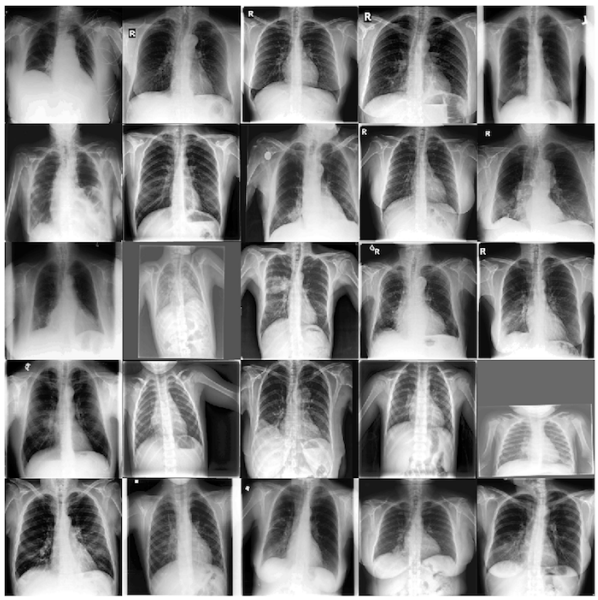


(b) Padchest

Figure 9.8: Before Histogram Equalization

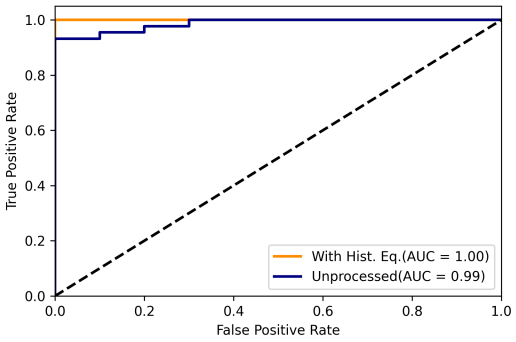


(a) V2-COV19-NII



(b) Padchest

Figure 9.9: After Histogram Equalization



(a) Internal Testing

	Precision	Recall	F1-score	Accuracy	AUC
With Hist. Equ.	0.54	0.5	0.52	0.61	0.66
Unprocessed	0.52	0.27	0.36	0.58	0.59

(b) External Testing

Figure 9.10: Performance Comparison of Models Trained with Histogram Equalized Pre-processed Images vs with Unprocessed Images

We tested the last model's snapshot (20th). Fig. 9.10a compares the internal performance of models trained with histogram equalized images vs with unprocessed images. While Fig. 9.10b compares the external performance of the two models. The results suggest that the model performs statistically significantly better when trained with histogram equalized images. Using the McNemar's test [96], we calculated a p-value of 1.7×10^{-4} which is lower than the significance threshold, $alpha = 0.01$. There is a significant difference between the model's performance on unseen sources before and after histogram equalization with 99% confidence.

However, a high testing accuracy gap between external and internal sources still exists, which suggests the persistence of shortcut learning.

Similar techniques such as quantile normalization and histogram matching were also explored but determined to be no more effective.

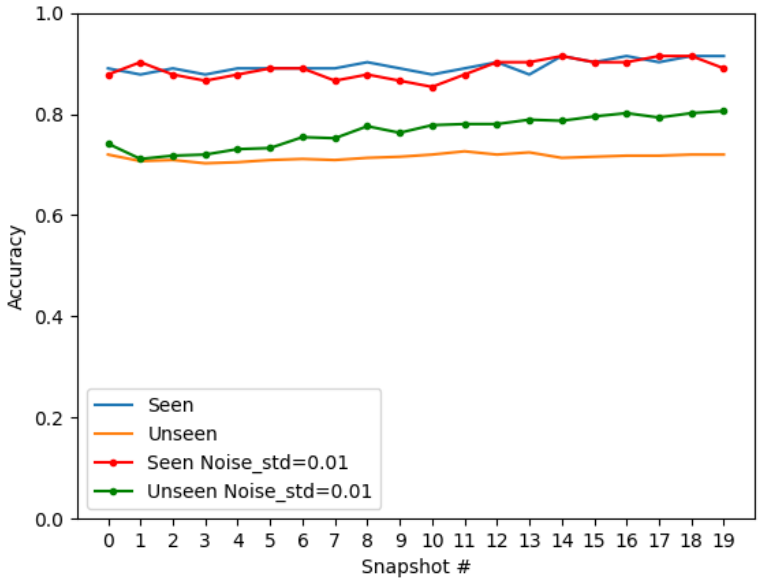


Figure 9.11: Test Accuracy of the Saved Snapshots on Seen vs Unseen Data Sources After the Gaussian Noise Based Training Data Augmentation

9.6.4 Noise-based Augmentation

In addition to image rotations, zero mean Gaussian noise with a standard deviation of 0.01 was applied to the original images. Over-fitting usually happens when the deep learning model tries to learn high-frequency features (patterns that frequently occur) which may not be useful. Gaussian noise, which has zero mean, essentially has data points in all frequencies, effectively distorting the high-frequency features and helping the model to look past them.

We applied this technique as a data augmentation procedure. Samples were augmented prior to training, while original images were also included during training.

The Euclidean distance between the original image and its noisy augmentation was 0.014. Fig. 9.11 shows the comparison of the performance gap between seen and unseen sources before adding noise-based augmentation (solid marker) and after (dotted marker).

To train the model BIMCV-COVID-19+ source was used for the COVID-19 positive class and Padchest for the COVID-19 negative class. For model testing a set of 480 images was used. There were 240 from the positive class including 77 from the COVID-19-AR source

Accuracy	baseline	std = 0.01	std=0.05	std=0.005
Seen	0.92	0.9	0.88	0.94
Unseen	0.72	0.81	0.78	0.78
Difference	0.2	0.09	0.1	0.16

Figure 9.12: Comparison of the Performance Gap When Testing the 20th Snapshot on Seen vs Unseen Data Sources for Multiple Gaussian Noise Standard Deviation Values

and 163 from the V2-COV19-NII source. The negative class consisted of 116 from the NIH source and 116 from the Chexpert source.

The test accuracy of the last snapshot (20th) when using only rotation and flip augmentation for seen sources was 92% and 72% for unseen sources. When adding Gaussian noise-based augmentation, the accuracy for seen was 90% and 81% for unseen sources. After adding the Gaussian noise-based augmentation, the accuracy gap was improved to only 9% instead of 20%.

The test accuracy on unseen sources is statistically significantly higher when adding Gaussian noise augmentation (81%) than when only using flip and rotate augmentation (72%). Using the McNemar's test [96], we calculated a p-value of 8.13×10^{-5} which is lower than the significance threshold, $alpha = 0.01$. There is a significant difference between the model's performance on unseen sources before and after adding Gaussian noise-based data augmentation with 99% confidence.

The results suggest that making features that are not medically relevant, but good discriminators, noisy can result in less focus on them. Medically relevant features are naturally expected to have some "noise" and be less affected. To further assess the effect of the amount of noise added, we experimented with multiple values of Gaussian noise standard deviation {0.005, 0.01, 0.05}. Fig. 9.12 shows the results of testing the 20th model snapshot for different values of Gaussian noise standard deviation. Based on the obtained results, it doesn't seem to hurt the accuracy much when testing data is from seen sources. However, it allowed significant improvement to the model's performance on data from unseen sources.

This suggests that the model has learned to ignore confounding features and focus on more medically meaningful differences between classes.

To answer the question of how much noise is good enough, based on our experimental results, adding more noise (std=0.05) has little to no effect on the accuracy performance for seen sources. However, it makes the accuracy of unseen sources slightly lower than a noise application of std=0.01. Less noise (std= 0.005) increased the accuracy performance on both the seen and unseen sources at the early stages but in the end, the performance gap becomes larger. Generally, we noticed that the application of noise-based data augmentation had no effect (or a slight improvement) on the model's performance on seen sources. However, it has significantly improved the model's performance on unseen sources.

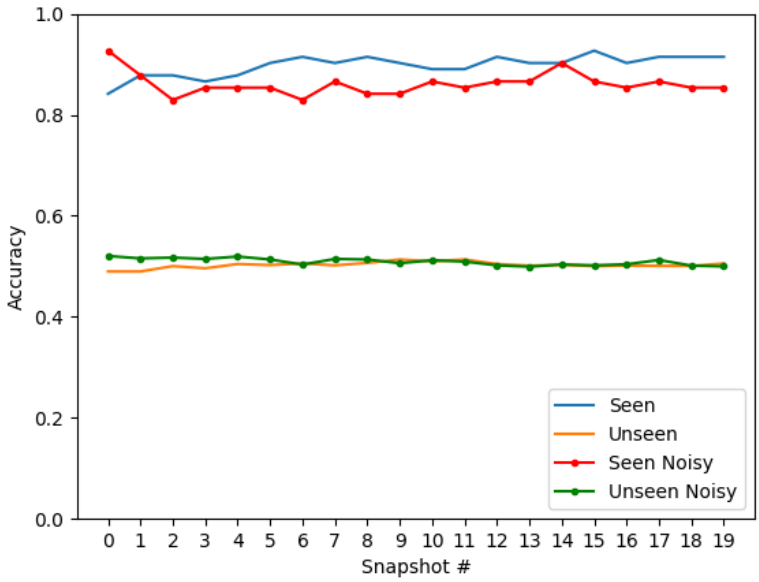


Figure 9.13: Validation of the Proposed Method on Three More Unseen Sources

For further validation, we tested the model on more unseen sources: 425 images from COVIDGR(+) and 500 images from CC-CXRI-P(+) for the positive class. For the negative class we used 967 images from CC-CXRI-P(+). Figure 9.13 shows the comparison of the

Table 9.3: Testing Accuracy Performance on Three More Unseen Sources Before vs After Noise-based Augmentation

Class	Data Source (Number of Samples)	Accuracy Pre-Aug.	Accuracy Post-Aug.
COVID-19(+)	COVIDGR(+)(425)	34 %	34 %
	CC-CXRI-P(+)(500)	74 %	55 %
COVID-19(-)	CC-CXRI-P(-)(967)	45 %	58 %

performance gap between seen and unseen sources before adding noise-based augmentation (solid marker) and after (dotted marker). It can be seen in Figure 9.13 that the performance on the unseen source did not improve much but the model performed slightly lower on seen sources. Table 9.3 shows the accuracy results of each of the unseen sources before and after the application of noise-based augmentation. It can be noticed from Table 9.3 that after applying the noise-based augmentation method the model is now performing 13% higher on CC-CXRI-P(-). However, the performance on CC-CXRI-P(+) decreased. Furthermore, the model's performance on COVIDGR(+) was unchanged We believe that the low performance on COVIDGR(+) is caused by the X-ray view of the images in this source, as explained in the data sources description above, all images in COVIDGR(+) were in PA view. If the model is relying on the X-ray view as one of the shortcuts then probably our method did not help to mitigate it here. Due to the fact that we don't usually know what type of pre-processing the images from different sources underwent, it's difficult to explain the result of our method on the other sources.

9.7 Discussion

As we mentioned before, in this study we used a customized Resnet-50 as a CNN model. However, the problem of shortcut learning is present in all deep learning based models and our solution is applicable to any CNN regardless of the architecture.

To experimentally show that shortcut learning exists in other architectures, we tested a COVID-19 diagnosis system called DeepCOVID-XR [149]. It is claimed that it is capable of

Table 9.4: Testing Accuracy Performance of DeepCOVID-XR on Unseen COVID-19 Sources

Data Source (Number of Samples)	Sensitivity
BIMCV COVID-19+ (38)	66%
V2-COV19-NII (163)	53%
COVID-19-AR (70)	63%
Overall (average)	58%

generalizing to a held-out test data set of 2214 images (1192 positive for COVID-19) from a single institution that the algorithm was not exposed to during model development. Their method achieved a sensitivity of 75% (898 of 1192). Authors claim that DeepCOVID-XR was trained and tested on, to their knowledge, the largest clinical data set of chest X-rays from the COVID-19 era. DeepCOVID-XR consists of an ensemble of six CNN architectures (DenseNet-121, ResNet-50, InceptionV3, Inception-ResNetV2, Xception, and EfficientNet-B2). The CNNs in this ensemble were pre-trained on the NIH X-rays dataset [148] and were then fine-tuned on a private clinical training set. Note that the external testing was done on data from an institution that is affiliated with the same health care system from which the training data was obtained. Therefore, there is a significant likelihood that the testing data have a very similar distribution and cover a very similar population as the training data. Thus, their good testing results (sensitivity of 75%) may not guarantee generalization. Their pre-trained models were made available on Github [65]. We tested their system on some external sources of COVID-19 positive to assess the robustness and generalization ability of their solution. Table. 9.4 shows the testing results with some significant drop-offs in accuracy. Overall the model had a sensitivity of 58% (156 of 271).

This suggests that having a large amount of training data from multiple sources and an ensemble of several advanced CNN architectures is not a complete solution to the problem of generalization to unseen sources. It may be impractical to have one set of data that covers the space of possible X-ray data acquisition parameters and protocols.

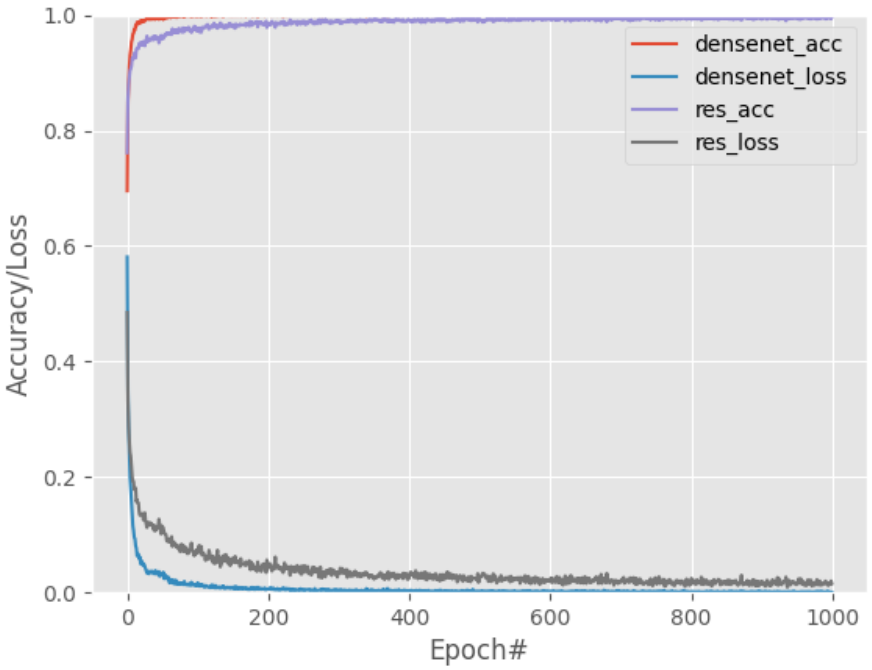


Figure 9.14: Comparison of Training Accuracy/Loss per Epoch for Resnet-50 vs Densenet-121 as Base Model

Additionally, we compared the modified Resnet-50 presented here against other CNN architectures. Instead of Resnet-50 as a pretrained base model, we used DenseNet-121. Fig. 9.14 shows a comparison of the training accuracy/loss per epoch of the compared models. For model training, as in Section 9.6.2, BIMCV-COVID-19+ data was used as the source for the COVID-19 positive class and Padchest for the COVID-19 negative class. It can be seen that both models have similar training accuracy graphs. Also, Densenet-121 was faster to reach 100% training accuracy than Resnet-50. The internal vs external testing results of Densenet-121 are presented in Table 9.5. The results (drop-off = 0.37) suggest that Densenet-121 suffers from shortcut learning as well. We can also notice that when a model has higher training accuracy, it achieves higher seen testing accuracy but has a larger drop-off when tested externally.

Table 9.5: Comparison of Internal Vs External Performance of Resnet-50 vs Densenet-121 as Base Model

Model	Num of params	Training Acc	Seen Acc	Unseen Acc	Drop-off
ResNet-50	184,449	0.9956	0.82	0.53	0.29
DenseNet-121	149,441	1.0	0.93	0.56	0.37

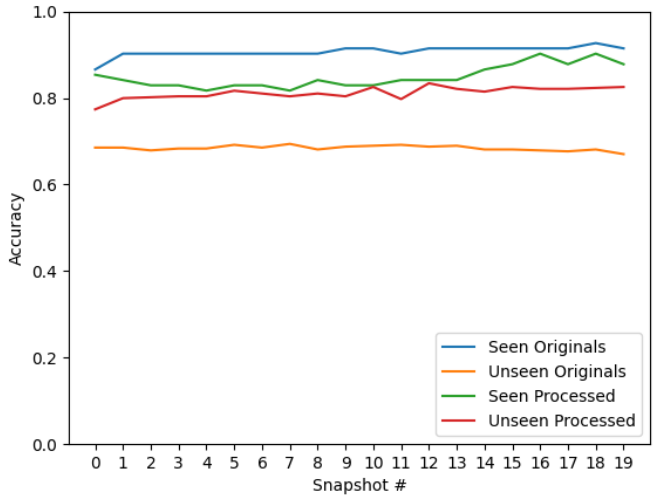


Figure 9.15: Test Accuracy of the Saved Densenet-121 Snapshots on Seen vs Unseen Data Sources after Application of the Proposed Approach

In order to determine if our method helps using DenseNet-121 as a baseline model, we applied the proposed solution and compared seen vs unseen testing performance before and after the application of the proposed method. We used the same train and test sets used in Section 9.6.4 with ResNet-50 based model. Fig. 9.15 shows results. It can be seen that the drop-off between internal and external testing has been reduced with the application of our proposed pipeline.

On the other hand, to our knowledge, there are no similar studies to compare with our approach. However, in a related paper, a study on congestive heart failure [66] observed in CXRs addressed shortcuts by pre-training a model originally trained on Imagenet on a related problem without shortcuts (pneumonia and other lung diseases which show up in CXR). They found this approach reduced the likelihood of the final tuning capturing

shortcuts. However, the approach requires choosing one particular known bias (age, BMI, sex, race, and presence of pacemaker) to precisely target and it needs careful data curation and extra steps in training. In our case, we are thinking of unknown, "concealed" shortcuts.

9.8 Summary

Deep learning based medical diagnostic systems cannot be considered clinically mature until they are also shown to be generalizable across source populations. The issues raised here apply to other image modalities and other diseases for which X-ray imaging is used. We point out shortcuts to look out for while training models and suggest steps for training data curation and preprocessing to avoid shortcut learning. The use of lung segmentation, histogram equalization, and added noise to images significantly reduced the gap in performance between cross-validation results and unseen source predictions. While overcoming shortcut learning entirely may require new neural network formulations, in this work we show the first steps towards mitigating it. Thus, leading to more fair, reliable, robust, and transparent deep learning-based clinical decision support systems.

Chapter 10: Conclusions and Future Work

10.1 Glioblastoma Prognosis

For Glioblastoma prognosis, we proposed multiple solutions to overcome the challenge of limited availability of public medical image datasets. To perform the task of overall survival analysis of Glioblastoma patients, we first explored the potential of using transfer learning and fine-tuning CNN models previously pre-trained on large-scale datasets like ImageNet. Then, we developed an ensemble-based training and testing approach with small CNNs trained from scratch. Finally, we built a snapshot learning-based system with a multi-branch CNN to merge multi-sequence MRIs for a higher predictive performance of overall survival.

This work has the limitation of using a pre-operative dataset to perform OS prediction. This can result in leaving out several prognostic factors that are available after histological confirmation. In research, the other most common prognostic factors for survival in glioblastoma patients were found to be: The age of patients, extent of resection, recursive partitioning analysis (RPA) class [131], and performance status (using Karnofsky Performance Scale (KPS) or the Eastern Cooperative Oncology Group (ECOG)/World Health Organization (WHO) performance status) [78]. Based on recent studies [73, 101, 13, 63, 128, 25, 122, 88, 26], improved communication and information exchange between radiology and histopathology experts is needed more than ever. Many papers have shown benefits in integrating several markers for more accurate OS prediction of patients with glioma. A study [63] used deep artificial neural networks on histopathologic images of gliomas and found an association of OS with several histopathologic features, such as pseudopalisading and geographic necrosis and inflammation. Recent work [73, 88, 122] has shown the integrated

potential of MR and histopathologic images, which has provided diagnostically relevant information for the prediction of OS in glioma. Similarly, it has been demonstrated that combining radiomics with multiomics can offer additional prognostic value [26]. Another study confirmed higher prediction accuracy when a combination of histopathologic images and genomic markers (isocitrate dehydrogenase [IDH], 1p/19q) was used [101]. Moreover, it has been shown that the addition of a radiomics model to clinical and genetic profiles improved survival prediction on glioblastoma when compared with models containing clinical and genetic profiles alone [13].

In the future, we intend to create a more advanced version of our deep learning model combining radiologic-based features from MR imaging with pathologic, clinical, and genetic markers to develop more informed and better predictors of overall survival of GBM patients.

In addition to IDH mutations [55] and 1p/19q codeletion [17], the methylation status of the O6-methyl guanine DNA methyltransferase (MGMT) gene promoter has been shown to be a strong predictor of the survival of glioblastoma patients [57]. There exist other prognostic molecular markers for which diagnostic evaluation is not routinely performed such as G-CIMP methylation, TERT promoter mutations, EGFR alterations, BRAF V600E mutations, Histone mutations, and H3K27 mutation, which can occur in histone H3.1 or H3.3 [11].

On the other hand, it may be possible to identify additional prognostic information using image analysis of the post-operative or post-treatment MRIs. A recent study has demonstrated an association between post-operative residual contrast-enhancing tumor volume in the post-surgical MRI and overall survival in newly diagnosed glioblastoma [42]. Furthermore, in our recent work [83], we showed that deep features extracted from post-treatment MRI scans, which were obtained during or after therapy, can entail relevant information for overall survival prediction. Therefore, to lend further validity to our model, we aim to combine data before and after therapy or resection, if available.

Recently, there has been increased interest in integrating non-invasive imaging techniques with clinical care for brain tumor patients. For instance, the use of amino acid PET has been shown to better identify the most biologically aggressive components of heterogeneous low and high-grade glioma [136, 12]. As another future work direction, we may explore the potential of non-invasive metabolic imaging as complementary to conventional MRI to predict survival outcomes of GBM patients using convolutional neural networks.

10.2 COVID-19 Diagnosis

For COVID-19 Diagnosis, we experimentally demonstrated that high-performing CNN-based models can be built to predict COVID-19 disease from X-rays. However, we discovered that the majority of these models suffer from drop-off in performance when tested on unseen data sources. We identified this problem as a shortcut learning issue. We showed how CNNs can exploit some unintended confounding features as shortcuts to make predictions, which, in medical image analysis, can lead to false diagnoses causing dangerous consequences to the examined patients. Finally, we point out multiple shortcuts, identified from our experiments, and practical solutions to mitigate them.

In this work, we show evidence that models, created using deep learning, which attain high accuracy/AUC on unseen data from seen sources, exhibit clear generalization gap issues and are unable to perform as well on data from external unseen sources. Unfortunately, we have too few data sources to conclude definitively that this inconsistency in performance is solely attributed to the differences in data sources or undisclosed preprocessing, or other unknown factors. CXRs of the same COVID-19 patient from two different sources would help as would full information on acquisition machines and parameters, which are not available to us at this time. Some of the data sources used in this work underwent partially or fully unknown pre-processing techniques that were not explained by the owners of the datasets. Such missing detail about the data limits our ability to be sure of providing a uniform normalization for all data sources. Due to the rapid and massive growth of the recent literature related to

COVID-19 diagnosis using AI methods from X-rays, we cannot be sure that we covered all papers. However, to our knowledge, none has proved its ability to generalize to external sites, which is the main focus of this study.

Our work here is concerned with mitigating shortcuts on the data level. In the future, we are planning to explore the possibility of solving shortcut learning at the model level. The black-box nature of CNNs continues to be further explored. More complex loss functions may be needed to avoid learning simple functions which capture shortcuts. New approaches may alleviate the problems encountered in cross-source generalization. Numerous alternate state-of-the-art techniques may provide significant improvements for medical imaging classification tasks on unseen sources and should be studied such as: multi-task learning [23], few-shot learning [45], semi-supervised learning [27], attention models [44], data synthesis (especially of confounders such as synthetic tubes/devices) [49], and federated learning. Each of these approaches improves generalization by ensuring that feature distillation is appropriately abstract, while also ensuring that the system is focused on classifying with the appropriate features. While federated learning has become a hot topic for the medical industry, it requires a high degree of coordination between facilities, and thus, the former techniques such as generative models are likely a more practical solution [67]. It must be noted that even if there is a broad variety of data there can still be shortcuts learned. Like a high likelihood of a person with a device attached having the disease. While a device indicates a suspected illness, no human doctor would diagnose based on its presence.

The medical industry, as well as any mission-critical application, requires trust in an algorithm, but this trust need not be implicit. Explainable AI (XAI) is another key factor for engineers and developers to successfully transition technology from the lab to the clinic. Decision support systems for the medical community can be hugely beneficial to the radiomics and pathology communities, but recommendations will need to be evidence-based. Technologies such as GradCAM are helping researchers, but other more sophisticated XAI solutions will be needed in the clinical space.

This work has resulted in the following publications:

- K. Ben Ahmed, L. O. Hall, D. B. Goldgof and R. Fogarty, "Achieving Multisite Generalization for CNN-Based Disease Diagnosis Models by Mitigating Shortcut Learning," in IEEE Access, vol. 10, pp. 78726-78738, 2022, doi: 10.1109/ACCESS.2022.3193700

- Ben Ahmed, K.; Hall, L.O.; Goldgof, D.B.; Gatenby, R., "Ensembles of Convolutional Neural Networks for Survival Time Estimation of High-Grade Glioma Patients from Multimodal MRI". Diagnostics 2022, 12, 345. https://doi.org/10.3390/diagnostics12020345

- K. B. Ahmed, G. M. Goldgof, R. Paul, D. B. Goldgof and L. O. Hall, "Discovery of a Generalization Gap of Convolutional Neural Networks on COVID-19 X-Rays Classification," in IEEE Access, vol. 9, pp. 72970-72979, 2021, doi: 10.1109/ACCESS.2021.3079716

- K. B. Ahmed, L. O. Hall, R. Liu, R. A. Gatenby and D. B. Goldgof, "Neuroimaging Based Survival Time Prediction of GBM Patients Using CNNs from Small Data," 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), 2019, pp. 1331-1335, doi: 10.1109/SMC.2019.8913929

- R. Liu, L. O. Hall, K. W. Bowyer, D. B. Goldgof, R. Gatenby and K. Ben Ahmed, "Synthetic minority image over-sampling technique: How to improve AUC for glioblastoma patient survival prediction," 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017, pp. 1357-1362, doi: 10.1109/SMC.2017.8122802

- Ahmed, Kaoutar B., Lawrence O. Hall, Dmitry B. Goldgof, Renhao Liu, and Robert A. Gatenby. "Fine-tuning convolutional deep features for MRI based brain tumor classification." In Medical Imaging 2017: Computer-Aided Diagnosis, vol. 10134, pp. 613-619. SPIE, 2017

- Renhao Liu, L. O. Hall, D. B. Goldgof, Mu Zhou, R. A. Gatenby and K. B. Ahmed, "Exploring deep features from brain tumor magnetic resonance images via transfer learning," 2016 International Joint Conference on Neural Networks (IJCNN), 2016, pp. 235-242, doi: 10.1109/IJCNN.2016.7727204

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. OSDI'16, page 265–283, USA, 2016. USENIX Association.

[2] Asmaa Abbas, Mohammed M. Abdelsamea, and Mohamed Medhat Gaber. Classification of COVID-19 in chest x-ray images using DeTraC deep convolutional neural network. *Applied Intelligence*, 51(2):854–864, Sep 2020. doi: 10.1007/s10489-020-01829-7.

[3] Parnian Afshar, Shahin Heidarian, Farnoosh Naderkhani, Anastasia Oikonomou, Konstantinos N. Plataniotis, and Arash Mohammadi. COVID-CAPS: A capsule network-based framework for identification of COVID-19 cases from x-ray images. *Pattern Recognition Letters*, 138:638–643, Oct 2020. doi: 10.1016/j.patrec.2020.09.010.

[4] Rupal Agravat and Mehul S Raval. Brain tumor segmentation and survival prediction. *arXiv preprint arXiv:1909.09399*, 2019.

[5] Kaoutar Ben Ahmed, Lawrence O Hall, Renhao Liu, Robert A Gatenby, and Dmitry B Goldgof. Neuroimaging based survival time prediction of gbm patients using cnns from small data. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 1331–1335. IEEE, 2019.

[6] Mahbubul Alam, Lasitha Vidyaratne, Zeina Shboul, L Pei, T Batchelder, and KM Iftekharuddin. Deep learning and radiomics for glioblastoma survival prediction. In *Pre-conference Proceedings of the 7th MICCAI BraTS Challenge*, pages 11–18, 2018.

[7] Mehdi Amian and Mohammadreza Soltaninejad. Multi-resolution 3d cnn for mri brain tumor segmentation and survival prediction. *arXiv preprint arXiv:1911.08388*, 2019.

[8] Javeria Amin, Muhammad Sharif, Mussarat Yasmin, Tanzila Saba, Muhammad Almas Anjum, and Steven Lawrence Fernandes. A new approach for brain tumor segmentation and classification based on score level fusion using transfer learning. *Journal of medical systems*, 43(11):1–16, 2019.

[9] Ioannis D. Apostolopoulos, Sokratis I. Aznaouridis, and Mpesiana A. Tzani. Extracting possibly representative COVID-19 biomarkers from x-ray images with deep learning approach and image data related to pulmonary diseases. *Journal of Medical and Biological Engineering*, 40(3):462–469, May 2020. doi: 10.1007/s40846-020-00529-4.

[10] Ioannis D. Apostolopoulos and Tzani A. Mpesiana. Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, Apr 2020. doi: 10.1007/s13246-020-00865-4.

[11] Elisa Aquilanti, Julie Miller, Sandro Santagata, Daniel P Cahill, and Priscilla K Brastianos. Updates in prognostic markers for gliomas. *Neuro-oncology*, 20(suppl_7):vii17–vii26, 2018.

[12] Javier Arbizu, S Tejada, JM Marti-Climent, R Diez-Valle, E Prieto, G Quincoces, C Vigil, MA Idoate, JL Zubieta, I Peñuelas, et al. Quantitative volumetric analysis of gliomas with sequential mri and 11 c-methionine pet assessment: patterns of integration in therapy planning. *European journal of nuclear medicine and molecular imaging*, 39(5):771–781, 2012.

[13] Sohi Bae, Yoon Seong Choi, Sung Soo Ahn, Jong Hee Chang, Seok-Gu Kang, Eui Hyun Kim, Se Hoon Kim, and Seung-Koo Lee. Radiomic MRI phenotyping of glioblastoma: Improving survival prediction. *Radiology*, 289(3):797–806, dec 2018.

[14] Chandan Ganesh Bangalore Yogananda, Bhavya R Shah, Maryam Vejdani-Jahromi, Sahil S Nalawade, Gowtham K Murugesan, Frank F Yu, Marco C Pinho, Benjamin C Wagner, Bruce Mickey, Toral R Patel, et al. A novel fully automated mri-based deep-learning method for classification of idh mutation status in brain gliomas. *Neuro-oncology*, 22(3):402–411, 2020.

[15] Pedro R. A. S. Bassi and Romis Attux. A deep convolutional neural network for COVID-19 detection using chest x-rays. *Research on Biomedical Engineering*, Apr 2021. doi: 10.1007/s42600-021-00132-9.

[16] Sanhita Basu, Sushmita Mitra, and Nilanjan Saha. Deep learning for screening covid-19 using chest x-ray images. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 2521–2527. IEEE, 2020.

[17] Brigitta G Baumert, Monika E Hegi, Martin J van den Bent, Andreas von Deimling, Thierry Gorlia, Khê Hoang-Xuan, Alba A Brandes, Guy Kantor, Martin JB Taphoorn, Mohamed Ben Hassel, et al. Temozolomide chemotherapy versus radiotherapy in high-risk low-grade glioma (eortc 22033-26033): a randomised, open-label, phase 3 intergroup study. *The lancet oncology*, 17(11):1521–1532, 2016.

[18] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473, 2018.

[19] Jose Bernal, Kaisar Kushibar, Daniel S. Asfaw, Sergi Valverde, Arnau Oliver, Robert Martí, and Xavier Lladó. Deep convolutional neural networks for brain image analysis on magnetic resonance imaging: a review. *Artificial Intelligence in Medicine*, 95:64–81, apr 2019.

[20] Stevo Bozinovski. Reminder of the first paper on transfer learning in neural networks, 1976. *Informatica*, 44(3), 2020.

[21] Keno K Bressem, Lisa C Adams, Christoph Erxleben, Bernd Hamm, Stefan M Niehues, and Janis L Vahldiek. Comparing different deep learning architectures for classification of chest radiographs. *Scientific reports*, 10(1):1–16, 2020.

[22] Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria de la Iglesia-Vayá. PadChest: A large chest x-ray image dataset with multi-label annotated reports. *Medical Image Analysis*, 66:101797, Dec 2020. doi: 10.1016/j.media.2020.101797.

[23] Rich Caruana. Multitask Learning. *Machine Learning*, (28):41–75, July 1997.

[24] Isabella Castiglioni, Davide Ippolito, Matteo Interlenghi, Caterina Beatrice Monti, Christian Salvatore, Simone Schiaffino, Annalisa Polidori, Davide Gandola, Cristina Messa, and Francesco Sardanelli. Artificial intelligence applied on chest x-ray can aid in the diagnosis of COVID-19 infection: a first experience from lombardy, italy, Apr 2020. doi: 10.1101/2020.04.08.20040907.

[25] Ahmad Chaddad, Paul Daniel, Christian Desrosiers, Matthew Toews, and Bassam Abdulkarim. Novel radiomic features based on joint intensity matrices for predicting glioblastoma patient survival time. *IEEE Journal of Biomedical and Health Informatics*, 23(2):795–804, mar 2019.

[26] Ahmad Chaddad, Paul Daniel, Siham Sabri, Christian Desrosiers, and Bassam Abdulkarim. Integration of radiomic and multi-omic analyses predicts survival of newly diagnosed idh1 wild-type glioblastoma. *Cancers*, 11(8):1148, 2019.

[27] O. Chapelle, B. Scholkopf, and A. Zien, Eds. Semi-supervised learning. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[28] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[29] Lina Chato and Shahram Latifi. Machine learning and deep learning techniques to predict overall survival of brain tumor patients using mri images. In *2017 IEEE 17th International Conference on Bioinformatics and Bioengineering (BIBE)*, pages 9–14. IEEE, 2017.

[30] Francois Chollet. Keras. https://github.com/fchollet/keras, 2015.

[31] Prakash Choudhary and Abhishek Hazra. Chest disease radiography in twofold: using convolutional neural networks and transfer learning. *Evolving Systems*, 12(2):567–579, 2021.

[32] Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Rashid Mazhar, Muhammad Abdul Kadir, Zaid Bin Mahbub, Khandakar Reajul Islam, Muhammad Salman Khan, Atif Iqbal, Nasser Al Emadi, Mamun Bin Ibne Reaz, and Mohammad Tariqul Islam. Can AI help in screening viral and COVID-19 pneumonia? *IEEE Access*, 8:132665–132676, 2020. doi: 10.1109/access.2020.3010287.

[33] Joseph Paul Cohen, Paul Morrison, and Lan Dao. Covid-19 image data collection, 2020.

[34] Ling Dai, Liang Wu, Huating Li, Chun Cai, Qiang Wu, Hongyu Kong, Ruhan Liu, Xiangning Wang, Xuhong Hou, Yuexing Liu, et al. A deep learning system for detecting diabetic retinopathy across the disease spectrum. *Nature communications*, 12(1):1–11, 2021.

[35] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

[36] Dipayan Das, K. C. Santosh, and Umapada Pal. Truncated inception net: COVID-19 outbreak screening using chest x-rays. *Physical and Engineering Sciences in Medicine*, 43(3):915–925, Jun 2020. doi: 10.1007/s13246-020-00888-x.

[37] Awwal Muhammad Dawud, Kamil Yurtkan, and Huseyin Oztoprak. Application of deep learning in neuroradiology: brain haemorrhage classification using transfer learning. *Computational Intelligence and Neuroscience*, 2019, 2019.

[38] Maria de la Iglesia Vayá, José Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients., 2020.

[39] Alex J. DeGrave, Joseph D. Janizek, and Su-In Lee. AI for radiographic COVID-19 detection selects shortcuts over signal, Sep 2020. doi: 10.1101/2020.09.13.20193565.

[40] S. Desai, A. Baghal, T. Wongsurawat, S. Al-Shukri, K. Gates, P. Farmer, M. Rutherford, G.D. Blake, T. Nolan, T. Powell, K. Sexton, W. Bennett, and F. Prior. Chest imaging with clinical and genomic correlates representing a rural covid-19 positive population [data set], 2020. doi: 10.7937/tcia.2020.py71-5978.

[41] Lars Egevad, Daniela Swanberg, Brett Delahunt, Peter Ström, Kimmo Kartasalo, Henrik Olsson, Dan M Berney, David G Bostwick, Andrew J Evans, Peter A Humphrey, et al. Identification of areas of grading difficulties in prostate cancer and comparison with artificial intelligence assisted grading. *Virchows Archiv*, 477(6):777–786, 2020.

[42] Benjamin M Ellingson, Lauren E Abrey, Sarah J Nelson, Timothy J Kaufmann, Josep Garcia, Olivier Chinot, Frank Saran, Ryo Nishikawa, Roger Henriksson, Warren P Mason, et al. Validation of postoperative residual contrast-enhancing tumor volume as an independent prognostic factor for overall survival in newly diagnosed glioblastoma. *Neuro-oncology*, 20(9):1240–1250, 2018.

[43] Muhammad Farooq and Abdul Hafeez. Covid-resnet: A deep learning framework for screening of covid19 from radiographs, 2020.

[44] Amir Hossein Farzaneh and Xiaojun Qi. Facial expression recognition in the wild via deep attentive center loss. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2401–2410, 2021.

[45] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1126–1135. JMLR.org, 2017.

[46] U.S. Centers for Disease Control and Prevention (CDC). Interim clinical guidance for management of patients with confirmed coronavirus disease (covid-19). Accessed May. 4, 2022.

[47] Adrian Galdran, Gustavo Carneiro, and Miguel A González Ballester. Balanced-mixup for highly imbalanced medical image classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 323–333. Springer, 2021.

[48] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[49] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[50] Omer Hadad, Ran Bakalo, Rami Ben-Ari, Sharbell Y. Hashoul, and Guy Amit. Classification of breast lesions using cross-modal deep learning. *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 109–112, 2017.

[51] Holger A Haenssle, Christine Fink, Roland Schneiderbauer, Ferdinand Toberer, Timo Buhl, Andreas Blum, A Kalloo, A Ben Hadj Hassen, Luc Thomas, A Enk, et al. Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists. *Annals of oncology*, 29(8):1836–1842, 2018.

[52] Lawrence O Hall, Rahul Paul, Dmitry B Goldgof, and Gregory M Goldgof. Finding covid-19 from chest x-rays using deep learning on a small dataset, 2020.

[53] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[54] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10):993–1001, 1990.

[55] Christian Hartmann, Bettina Hentschel, Wolfgang Wick, David Capper, Jörg Felsberg, Matthias Simon, Manfred Westphal, Gabriele Schackert, Richard Meyermann, Torsten Pietsch, et al. Patients with idh1 wild type anaplastic astrocytomas exhibit worse prognosis than idh1-mutated glioblastomas, and idh1 mutation status accounts for the unfavorable prognostic effect of higher age: implications for classification of gliomas. *Acta neuropathologica*, 120(6):707–718, 2010.

[56] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[57] Monika E Hegi, Annie-Claire Diserens, Thierry Gorlia, Marie-France Hamou, Nicolas De Tribolet, Michael Weller, Johan M Kros, Johannes A Hainfellner, Warren Mason, Luigi Mariani, et al. Mgmt gene silencing and benefit from temozolomide in glioblastoma. *New England Journal of Medicine*, 352(10):997–1003, 2005.

[58] Ezz El-Din Hemdan, Marwa A Shouman, and Mohamed Esmail Karar. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images, 2020.

[59] Mohammad Hesam Hesamian, Wenjing Jia, Xiangjian He, and Paul Kennedy. Deep learning techniques for medical image segmentation: achievements and challenges. *Journal of digital imaging*, 32(4):582–596, 2019.

[60] Michael Horry, Subrata Chakraborty, Biswajeet Pradhan, Manoranjan Paul, Jing Zhu, Hui Wen Loh, Prabal Datta Barua, and U Rajendra Arharya. Debiasing pipeline improves deep learning model generalization for x-ray based lung nodule detection. *arXiv preprint arXiv:2201.09563*, 2022.

[61] Rui Hua, Quan Huo, Yaozong Gao, He Sui, Bing Zhang, Yu Sun, Zhanhao Mo, and Feng Shi. Segmenting brain tumor using cascaded v-nets in multimodal mr images. *Frontiers in Computational Neuroscience*, 14, 2020.

[62] Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E Hopcroft, and Kilian Q Weinberger. Snapshot ensembles: Train 1, get m for free. *arXiv preprint arXiv:1704.00109*, 2017.

[63] Muhammad Iftikhar, Saima Rathore, and MacLean Nasrallah. Analysis of microscopic images via deep neural networks can predict outcome and idh and 1p/19q codeletion status in gliomas. In *Journal of Neuropathology and experimental neurology*, volume 78, pages 553–553. OXFORD UNIV PRESS INC JOURNALS DEPT, 2001 EVANS RD, CARY, NC 27513 USA, 2019.

[64] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 590–597, 2019.

[65] IVPLatNU. Deepcovidxr, 2021.

[66] Sarah Jabbour, David Fouhey, Ella Kazerooni, Michael W. Sjoding, and Jenna Wiens. Deep learning applied to chest x-rays: Exploiting and preventing shortcuts. In Finale Doshi-Velez, Jim Fackler, Ken Jung, David Kale, Rajesh Ranganath, Byron Wallace, and Jenna Wiens, editors, *Proceedings of the 5th Machine Learning for Healthcare Conference*, volume 126 of *Proceedings of Machine Learning Research*, pages 750–782. PMLR, 07–08 Aug 2020.

[67] Ali Jahanian, Xavier Puig, Yonglong Tian, and Phillip Isola. Generative models as a data source for multiview representation learning. *arXiv preprint arXiv:2106.05258*, 2021.

[68] Cheng Ju, Aurélien Bibaut, and Mark van der Laan. The relative performance of ensemble methods with deep convolutional neural networks for image classification. *Journal of Applied Statistics*, 45(15):2800–2818, 2018.

[69] Md. Rezaul Karim, Till Dohmen, Michael Cochez, Oya Beyan, Dietrich Rebholz-Schuhmann, and Stefan Decker. DeepCOVIDExplainer: Explainable COVID-19 diagnosis from chest x-ray images. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, Dec 2020. doi: 10.1109/bibm49941.2020.9313304.

[70] Asif Iqbal Khan, Junaid Latief Shah, and Mohammad Mudasir Bhat. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. *Computer Methods and Programs in Biomedicine*, 196:105581, Nov 2020. doi: 10.1016/j.cmpb.2020.105581.

[71] Muhammad Attique Khan, Tallha Akram, Yu-Dong Zhang, and Muhammad Sharif. Attributes based skin lesion detection and recognition: A mask rcnn and transfer learning-based deep learning framework. *Pattern Recognition Letters*, 143:58–66, 2021.

[72] Shahin Khobahi, Chirag Agarwal, and Mojtaba Soltanalian. CoroNet: A deep network architecture for semi-supervised task-based identification of COVID-19 from chest x-ray images, Apr 2020. doi: 10.1101/2020.04.14.20065722.

[73] Philipp Kickingereder, Sina Burth, Antje Wick, Michael Götz, Oliver Eidel, Heinz-Peter Schlemmer, Klaus H. Maier-Hein, Wolfgang Wick, Martin Bendszus, Alexander Radbruch, and David Bonekamp. Radiomic profiling of glioblastoma: Identifying an imaging predictor of patient survival with improved performance over established clinical and radiologic risk models. *Radiology*, 280(3):880–889, sep 2016.

[74] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[75] Sabitha Krishnamoorthy, Sudhakar Ramakrishnan, Lanson Brijesh Colaco, Akshay Dias, Indu K Gopi, Gautham AG Gowda, KC Aishwarya, Veena Ramanan, Manju Chandran, et al. Comparing a deep learning model's diagnostic performance to that of radiologists to detect covid-19 features on chest radiographs. *Indian Journal of Radiology and Imaging*, 31(5):53, 2021. doi: 10.4103/ijri.IJRI_914_20.

[76] Rahul Kumar, Ridhi Arora, Vipul Bansal, Vinodh J Sahayasheela, Himanshu Buckchash, Javed Imran, Narayanan Narayanan, Ganesh N Pandian, and Balasubramanian Raman. Accurate prediction of COVID-19 using chest x-ray images through deep feature learning model with SMOTE and machine learning classifiers, Apr 2020. doi: 10.1101/2020.04.13.20063461.

[77] Scarpace L., Mikkelsen T., Rao S. Cha S., Tekchandani S., Gutman D., Saltz J. H., Erickson B. J., Pedano N., Flanders A. E., Barnholtz-Sloan J., Ostrom Q., Barboriak D., and Pierce L. J. Radiology data from the cancer genome atlas glioblastoma multiforme [tcga-gbm] collection [data set]., 2016.

[78] Kathleen R Lamborn, Susan M Chang, and Michael D Prados. Prognostic factors for survival of patients with glioblastoma: recursive partitioning analysis. *Neuro-oncology*, 6(3):227–235, 2004.

[79] Jiangwei Lao, Yinsheng Chen, Zhi-Cheng Li, Qihua Li, Ji Zhang, Jing Liu, and Guangtao Zhai. A deep learning-based radiomics model for prediction of survival in glioblastoma multiforme. *Scientific reports*, 7(1):1–8, 2017.

[80] Feng Li, Zheng Liu, Hua Chen, Minshan Jiang, Xuedian Zhang, and Zhizheng Wu. Automatic detection of diabetic retinopathy in retinal fundus photographs based on deep learning algorithm. *Translational vision science & technology*, 8(6):4–4, 2019.

[81] Tianyang Li, Zhongyi Han, Benzheng Wei, Yuanjie Zheng, Yanfei Hong, and Jinyu Cong. Robust screening of covid-19 from chest x-ray via discriminative cost-sensitive learning, 2020.

[82] Yan Li and Liming Xia. Coronavirus disease 2019 (covid-19): Role of chest ct in diagnosis and management. *American Journal of Roentgenology*, 214(6):1280–1286, 2020. PMID: 32130038.

[83] Renhao Liu, Lawrence O Hall, Dmitry B Goldgof, Mu Zhou, Robert A Gatenby, and Kaoutar B Ahmed. Exploring deep features from brain tumor magnetic resonance images via transfer learning. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 235–242. IEEE, 2016.

[84] Yuan Liu, Ayush Jain, Clara Eng, David H Way, Kang Lee, Peggy Bui, Kimberly Kanada, Guilherme de Oliveira Marinho, Jessica Gallegos, Sara Gabriele, et al. A deep learning system for differential diagnosis of skin diseases. *Nature medicine*, 26(6):900–908, 2020.

[85] David N Louis, Arie Perry, Pieter Wesseling, Daniel J Brat, Ian A Cree, Dominique Figarella-Branger, Cynthia Hawkins, HK Ng, Stefan M Pfister, Guido Reifenberger, et al. The 2021 who classification of tumors of the central nervous system: a summary. *Neuro-oncology*, 23(8):1231–1251, 2021.

[86] Marit Lucas, Ilaria Jansen, C Dilara Savci-Heijink, Sybren L Meijer, Onno J de Boer, Ton G van Leeuwen, Daniel M de Bruin, and Henk A Marquering. Deep learning for automatic gleason pattern classification for grade group determination of prostate biopsies. *Virchows Archiv*, 475(1):77–83, 2019.

[87] Dailin Lv, Wuteng Qi, Yunxiang Li, Lingling Sun, and Yaqi Wang. A cascade network for detecting covid-19 using chest x-rays, 2020.

[88] Luke Macyszyn, Hamed Akbari, Jared M Pisapia, Xiao Da, Mark Attiah, Vadim Pigrish, Yingtao Bi, Sharmistha Pal, Ramana V Davuluri, Laura Roccograndi, et al. Imaging patterns predict patient survival and molecular subtype in glioblastoma via machine learning techniques. *Neuro-oncology*, 18(3):417–425, 2015.

[89] Amirreza Mahbod, Gerald Schaefer, Chunliang Wang, Georg Dorffner, Rupert Ecker, and Isabella Ellinger. Transfer learning using a multi-scale and multi-network ensemble for skin lesion classification. *Computer Methods and Programs in Biomedicine*, 193:105475, 2020.

[90] Shruti Atul Mali, Abdalla Ibrahim, Henry C. Woodruff, Vincent Andrearczyk, Henning Müller, Sergey Primakov, Zohaib Salahuddin, Avishek Chatterjee, and Philippe Lambin. Making Radiomics More Reproducible across Scanner and Imaging Protocol Variations: A Review of Harmonization Methods. *Journal of Personalized Medicine*, 11(9):842, August 2021.

[91] James Manyika, Sree Ramaswamy, Somesh Khanna, Hugo Sarrazin, Gary Pinkus, Guru Sethupathy, and Andrew Yaffe. Digital america: A tale of the haves and havemores. Accessed April. 5, 2022.

[92] Claudia Mazo, Jose Bernal, Maria Trujillo, and Enrique Alegre. Transfer learning for classification of cardiovascular tissues in histological images. *Computer Methods and Programs in Biomedicine*, 165:69–76, 2018.

[93] John McCarthy. What is artificial intelligence. *URL: http://www-formal. stanford. edu/jmc/whatisai. html*, 2004.

[94] Richard McKinley, Micheal Rebsamen, Katrin Daetwyler, Raphael Meier, Piotr Radojewski, and Roland Wiest. Uncertainty-driven refinement of tumor-core segmentation using 3d-to-2d networks with label uncertainty. *arXiv preprint arXiv:2012.06436*, 2020.

[95] Scott Mayer McKinney, Marcin Sieniek, Varun Godbole, Jonathan Godwin, Natasha Antropova, Hutan Ashrafian, Trevor Back, Mary Chesus, Greg S Corrado, Ara Darzi, et al. International evaluation of an ai system for breast cancer screening. *Nature*, 577(7788):89–94, 2020.

[96] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, Jun 1947. doi: 10.1007/bf02295996.

[97] Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014.

[98] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016.

[99] Shervin Minaee, Rahele Kafieh, Milan Sonka, Shakib Yazdani, and Ghazaleh Jamalipour Soufi. Deep-COVID: Predicting COVID-19 from chest x-ray images using deep transfer learning. *Medical Image Analysis*, 65:101794, Oct 2020. doi: 10.1016/j.media.2020.101794.

[100] Makoto Miyagawa, Marly Guimaraes Fernandes Costa, Marco Antonio Gutierrez, João Pedro Guimarães Fernandes Costa, and Cicero Ferreira Fernandes Costa Filho. Detecting vascular bifurcation in ivoct images using convolutional neural networks with transfer learning. *IEEE Access*, 7:66167–66175, 2019.

[101] Pooya Mobadersany, Safoora Yousefi, Mohamed Amgad, David A Gutman, Jill S Barnholtz-Sloan, Jose Enrique Velazquez Vega, Daniel J Brat, and Lee AD Cooper. Predicting cancer outcomes from histology and genomics using convolutional networks. oct 2017.

[102] Joaquim De Moura, Lucia Ramos Garcia, Placido Francisco Lizancos Vidal, Milena Cruz, Laura Abelairas Lopez, Eva Castro Lopez, Jorge Novo, and Marcos Ortega. Deep convolutional approaches for the analysis of COVID-19 using chest x-ray images from portable devices. *IEEE Access*, 8:195594–195607, 2020. doi: 10.1109/access.2020.3033762.

[103] Pierre GB Moutounet-Cartan. Deep convolutional neural networks to diagnose covid-19 and other pneumonia diseases from posteroanterior chest x-rays, 2020.

[104] Himadri Mukherjee, Subhankar Ghosh, Ankita Dhar, Sk Md Obaidullah, K. C. Santosh, and Kaushik Roy. Deep neural network to detect COVID-19: one architecture for both CT scans and chest x-rays. *Applied Intelligence*, Nov 2020. doi: 10.1007/s10489-020-01943-6.

[105] Himadri Mukherjee, Subhankar Ghosh, Ankita Dhar, Sk Md Obaidullah, K. C. Santosh, and Kaushik Roy. Shallow convolutional neural network for COVID-19 outbreak screening using chest x-rays. *Cognitive Computation*, Feb 2021. doi: 10.1007/s12559-020-09775-9.

[106] Dominik Müller, Iñaki Soto-Rey, and Frank Kramer. Multi-disease detection in retinal imaging based on ensembling heterogeneous deep learning models. In *German Medical Data Sciences 2021: Digital Medicine: Recognize–Understand–Heal*, pages 23–31. IOS Press, 2021.

[107] Andrew Murphy. Radiographic contrast, 2022.

[108] Jakub Nalepa, Pablo Ribalta Lorenzo, Michal Marcinkiewicz, Barbara Bobek-Billewicz, Pawel Wawrzyniak, Maksym Walczak, Michal Kawulok, Wojciech Dudzik, Krzysztof Kotowski, Izabela Burda, Bartosz Machura, Grzegorz Mrukwa, Pawel Ulrych, and Michael P. Hayball. Fully-automated deep learning-powered system for DCE-MRI analysis of brain tumors. *Artificial Intelligence in Medicine*, 102:101769, jan 2020.

[109] nilearn. Nilearn. https://github.com/nilearn/nilearn, 2022.

[110] Hugo Oliveira and Jefersson dos Santos. Deep transfer learning for segmentation of anatomical structures in chest radiographs. In *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 204–211, 2018.

[111] John A. Onofrey, Dana I. Casetti-Dinescu, Andreas D. Lauritzen, Saradwata Sarkar, Rajesh Venkataraman, Richard E. Fan, Geoffrey A. Sonn, Preston C. Sprenkle, Lawrence H. Staib, and Xenophon Papademetris. Generalizable multi-site training and testing of deep neural networks using image normalization. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 348–351, 2019.

[112] Quinn T Ostrom, Haley Gittleman, Gabrielle Truitt, Alexander Boscia, Carol Kruchko, and Jill S Barnholtz-Sloan. CBTRUS statistical report: Primary brain and other central nervous system tumors diagnosed in the united states in 2011–2015. *Neuro-Oncology*, 20(suppl_4):iv1–iv86, oct 2018.

[113] Şaban Öztürk, Umut Özkaya, and Mücahid Barstuğan. Classification of coronavirus (COVID -19) from x-ray and CT images using shrunken features. *International Journal of Imaging Systems and Technology*, 31(1):5–15, Aug 2020. doi: 10.1002/ima.22469.

[114] Tulin Ozturk, Muhammed Talo, Eylul Azra Yildirim, Ulas Baran Baloglu, Ozal Yildirim, and U. Rajendra Acharya. Automated detection of COVID-19 cases using deep neural networks with x-ray images. *Computers in Biology and Medicine*, 121:103792, Jun 2020. doi: 10.1016/j.compbiomed.2020.103792.

[115] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[116] Radiopaedia. *Radiopaedia Blog RSS*. (accessed Dec. 10, 2020).

[117] Asra Rafi, Junaid Ali, Tahir Akram, Kiran Fiaz, Ahmad Raza Shahid, Basit Raza, and Tahir Mustafa Madni. U-net based glioblastoma segmentation with patient's overall survival prediction. In *International Symposium on Intelligent Computing Systems*, pages 22–32. Springer, 2020.

[118] Md Mamunur Rahaman, Chen Li, Yudong Yao, Frank Kulwa, Mohammad Asadur Rahman, Qian Wang, Shouliang Qi, Fanjie Kong, Xuemin Zhu, and Xin Zhao. Identification of COVID-19 samples from chest x-ray images using deep learning: A comparison of transfer learning approaches. *Journal of X-Ray Science and Technology*, 28(5):821–839, Sep 2020. doi: 10.3233/xst-200715.

[119] Mohammad Rahimzadeh and Abolfazl Attar. A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest x-ray images based on the concatenation of xception and ResNet50v2. *Informatics in Medicine Unlocked*, 19:100360, 2020. doi: 10.1016/j.imu.2020.100360.

[120] Sivaramakrishnan Rajaraman, Jenifer Siegelman, Philip O. Alderson, Lucas S. Folio, Les R. Folio, and Sameer K. Antani. Iteratively pruned deep learning ensembles for COVID-19 detection in chest x-rays. *IEEE Access*, 8:115041–115050, 2020. doi: 10.1109/access.2020.3003810.

[121] Sivaramakrishnan Rajaraman, Ghada Zamzmi, and Sameer K Antani. Novel loss functions for ensemble-based medical image classification. *Plos one*, 16(12):e0261307, 2021.

[122] Saima Rathore, Ahmad Chaddad, Muhammad A. Iftikhar, Michel Bilello, and Ahmed Abdulkadir. Combining MRI and histologic imaging features for predicting overall survival in patients with glioma. *Radiology: Imaging Cancer*, 3(4):e200108, jul 2021.

[123] Arshia Rehman, Saeeda Naz, Ahmed Khan, Ahmad Zaib, and Imran Razzak. Improving coronavirus (COVID-19) diagnosis using deep transfer learning, Apr 2020. doi: 10.1101/2020.04.11.20054643.

[124] Grad View Research. Artificial intelligence in healthcare market report, 2022-2030. Accessed April. 10, 2022.

[125] Michael Roberts, , Derek Driggs, Matthew Thorpe, Julian Gilbey, Michael Yeung, Stephan Ursprung, Angelica I. Aviles-Rivero, Christian Etmann, Cathal McCague, Lucian Beer, Jonathan R. Weir-McCall, Zhongzhao Teng, Effrossyni Gkrania-Klotsas, James H. F. Rudd, Evis Sala, and Carola-Bibiane Schönlieb. Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans. *Nature Machine Intelligence*, 3(3):199–217, Mar 2021. doi: 10.1038/s42256-021-00307-0.

[126] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[127] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[128] G Shukla, S Bakas, S Rathore, H Akbari, A Sotiras, and C Davatzikos. Radiomic features from multi-institutional glioblastoma mri offer additive prognostic value to clinical and genomic markers: focus on tcga-gbm collection. *International Journal of Radiation Oncology, Biology, Physics*, 99(2):E107–E108, 2017.

[129] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[130] SIRM. *COVID-19 DATABASE — SIRM*. (accessed Feb. 20, 2021).

[131] Roger Stupp, Monika E Hegi, Warren P Mason, Martin J Van Den Bent, Martin JB Taphoorn, Robert C Janzer, Samuel K Ludwin, Anouk Allgeier, Barbara Fisher, Karl Belanger, et al. Effects of radiotherapy with concomitant and adjuvant temozolomide versus radiotherapy alone on survival in glioblastoma in a randomised phase iii study: 5-year analysis of the eortc-ncic trial. *The lancet oncology*, 10(5):459–466, 2009.

[132] Roger Stupp, Sophie Taillibert, Andrew Kanner, William Read, David M. Steinberg, Benoit Lhermitte, Steven Toms, Ahmed Idbaih, Manmeet S. Ahluwalia, Karen Fink, Francesco Di Meco, Frank Lieberman, Jay-Jiguang Zhu, Giuseppe Stragliotto, David D. Tran, Steven Brem, Andreas F. Hottinger, Eilon D. Kirson, Gitit Lavy-Shahaf, Uri Weinberg, Chae-Yong Kim, Sun-Ha Paek, Garth Nicholas, Jordi Bruna, Hal Hirte, Michael Weller, Yoram Palti, Monika E. Hegi, and Zvi Ram. Effect of tumor-treating fields plus maintenance temozolomide vs maintenance temozolomide alone on survival in patients with glioblastoma. *JAMA*, 318(23):2306, dec 2017.

[133] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[134] Siham Tabik, Anabel Gómez-Ríos, José Luis Martín-Rodríguez, Iván Sevillano-García, Manuel Rey-Area, David Charte, Emilio Guirado, Juan-Luis Suárez, Julián Luengo, MA Valero-González, et al. Covidgr dataset and covid-sdnet methodology for predicting covid-19 based on chest x-ray images. *IEEE journal of biomedical and health informatics*, 24(12):3595–3605, 2020.

[135] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[136] Yoji Tanaka, Tadashi Nariai, Toshiya Momose, Masaru Aoyagi, Taketoshi Maehara, Toshiki Tomori, Yoshikazu Yoshino, Tsukasa Nagaoka, Kiichi Ishiwata, Kenji Ishii, et al. Glioma surgery using a multimodal navigation system with integrated metabolic images. *Journal of neurosurgery*, 110(1):163–172, 2009.

[137] Enzo Tartaglione, Carlo Alberto Barbano, Claudio Berzovini, Marco Calandri, and Marco Grangetto. Unveiling COVID-19 from CHEST x-ray with deep learning: A hurdles race with small data. *International Journal of Environmental Research and Public Health*, 17(18):6933, Sep 2020. doi: 10.3390/ijerph17186933.

[138] Lucas O Teixeira, Rodolfo M Pereira, Diego Bertolini, Luiz S Oliveira, Loris Nanni, George DC Cavalcanti, and Yandre MG Costa. Impact of lung segmentation on the diagnosis and explanation of covid-19 in chest x-ray images. *Sensors*, 21(21):7116, 2021.

[139] Joseph J Titano, Marcus Badgeley, Javin Schefflein, Margaret Pain, Andres Su, Michael Cai, Nathaniel Swinburne, John Zech, Jun Kim, Joshua Bederson, et al. Automated deep-neural-network surveillance of cranial images for acute neurologic events. *Nature medicine*, 24(9):1337–1341, 2018.

[140] Lucas M. Valério, Daniel H. A. Alves, Luigi F. Cruz, Pedro H. Bugatti, Claiton de Oliveira, and Priscila T. M. Saito. Deepmammo: Deep transfer learning for lesion classification of mammographic images. In *2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)*, pages 447–452, 2019.

[141] Andrea Vedaldi and Karel Lenc. Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 689–692, 2015.

[142] Mohammad Walid. Prognostic factors for long-term survival after glioblastoma. *The Permanente Journal*, 12(4), sep 2008.

[143] Guangyu Wang, Xiaohong Liu, Jun Shen, Chengdi Wang, Zhihuan Li, Linsen Ye, Xingwang Wu, Ting Chen, Kai Wang, Xuan Zhang, et al. A deep-learning pipeline for the diagnosis and discrimination of viral, non-viral and covid-19 pneumonia from chest x-ray images. *Nature biomedical engineering*, 5(6):509–521, 2021.

[144] Guotai Wang, Wenqi Li, Sébastien Ourselin, and Tom Vercauteren. Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks. In *International MICCAI brainlesion workshop*, pages 178–190. Springer, 2017.

[145] Shuo Wang, Chengliang Dai, Yuanhan Mo, Elsa Angelini, Yike Guo, and Wenjia Bai. Automatic brain tumour segmentation and biophysics-guided survival prediction. *arXiv preprint arXiv:1911.08483*, 2019.

[146] Wei Wang and Jianxun Gang. Application of convolutional neural network in natural language processing. In *2018 International Conference on Information Systems and Computer Aided Education (ICISCAE)*, pages 64–70. IEEE, 2018.

[147] Wenling Wang, Yanli Xu, Ruqin Gao, Roujian Lu, Kai Han, Guizhen Wu, and Wenjie Tan. Detection of SARS-CoV-2 in Different Types of Clinical Specimens. *JAMA*, Mar 2020. doi: 10.1001/jama.2020.3786.

[148] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

[149] Ramsey M. Wehbe, Jiayue Sheng, Shinjan Dutta, Siyuan Chai, Amil Dravid, Semih Barutcu, Yunan Wu, Donald R. Cantrell, Nicholas Xiao, Bradley D. Allen, Gregory A. MacNealy, Hatice Savas, Rishi Agrawal, Nishant Parekh, and Aggelos K. Katsaggelos. DeepCOVID-XR: An artificial intelligence algorithm to detect COVID-19 on chest radiographs trained and tested on a large u.s. clinical data set. *Radiology*, 299(1):E167–E176, apr 2021.

[150] Ralph Weissleder, Hakho Lee, Jina Ko, and Mikael J. Pittet. COVID-19 diagnostics in context. *Science Translational Medicine*, 12(546):eabc1931, Jun 2020. doi: 10.1126/scitranslmed.abc1931.

[151] Hinrich B. Winther, Hans Laser, Svetlana Gerbel, Sabine K. Maschke, Jan B. Hinrichs, Jens Vogel-Claussen, Frank K. Wacker, Marius M. Höper, and Bernhard C. Meyer. Covid-19 image repository, May 2020. doi: 10.6084/m9.figshare.12275009.v1.

[152] Worldometers. *Coronavirus Cases.* (accessed Dec. 22, 2020).

[153] Xiaowei Xu, Jiancheng Lin, Ye Tao, and Xiaodong Wang. An improved densenet method based on transfer learning for fundus medical images. In *2018 7th International Conference on Digital Home (ICDH)*, pages 137–140. IEEE, 2018.

[154] Chun-Fu Yeh, Hsien-Tzu Cheng, Andy Wei, Keng-Chi Liu, Mong-Chi Ko, Po-Chen Kuo, Ray-Jade Chen, Po-Chang Lee, Jen-Hsiang Chuang, Chi-Mai Chen, et al. A cascaded learning strategy for robust covid-19 pneumonia chest x-ray screening, 2020.

[155] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 856–863, 2003.

[156] Yading Yuan, Ming Chao, and Yeh-Chi Lo. Automatic skin lesion segmentation using deep fully convolutional networks with jaccard distance. *IEEE transactions on medical imaging*, 36(9):1876–1886, 2017.

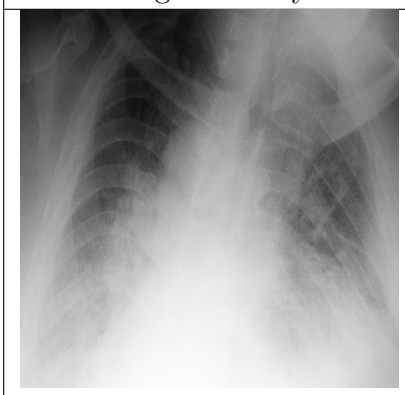
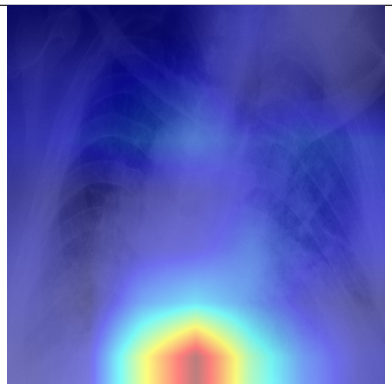
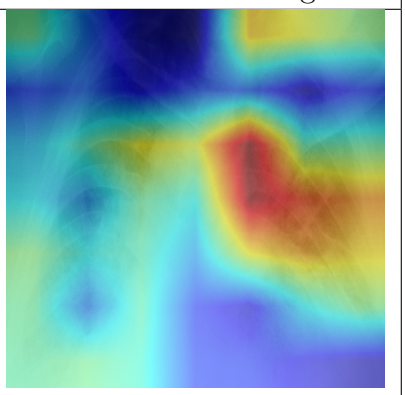
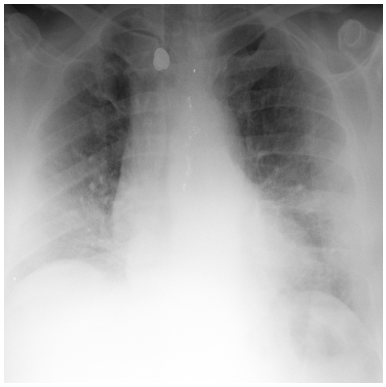
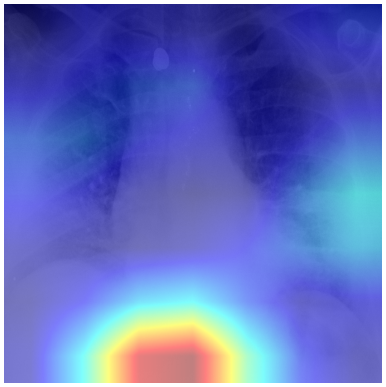
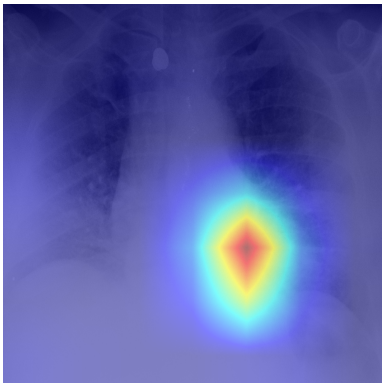
[157] John R Zech, Marcus A Badgeley, Manway Liu, Anthony B Costa, Joseph J Titano, and Eric Karl Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study. *PLoS medicine*, 15(11):e1002683, 2018.

[158] Mu Zhou, Jacob Scott, Baishali Chaudhury, Lawrence Hall, Dmitry Goldgof, Kristen W Yeom, Michael Iv, Yangming Ou, Jayashree Kalpathy-Cramer, Sandy Napel, et al. Radiomics in brain tumor: image assessment, quantitative feature descriptors, and machine-learning approaches. *American Journal of Neuroradiology*, 39(2):208–216, 2018.

Appendix A: Grad-Cam After Fine-Tuning

Table A.1 shows Grad-Cam visualization of two test samples before and after fine-tuning the model. As seen in the images, it is hard to confirm that fine-tuning has succeeded in making the model focus on lung area, though the focus there is increased. We do observe that some seemingly random locations outside the lungs are highlighted.

Table A.1: Grad-Cam Visualization of Two Test Samples Before and After Fine-tuning

Original X-ray	Before Fine-tuning	After Fine-tuning
		
		

Appendix B: The 68 Cases Used in Chapter 4

Class	Cases
Long-Term	TT020006_0011,TT020011_0006,TT020034_0011,TT020047_0014,TT020064_0003,TT020068_0003,TT020075_0002,TT020085_0006,TT020102_0003,TT020116_0022,TT020125_0005,TT020137_0003,TT020147_0036,TT020164_0001,TT020168_0001,TT020179_0033,TT020189_0004,TT020214_0003,TT020216_0017,TT020221_0033,TT020380_0007,TT020616_0009,TT020769_0009,TT020772_0012,TT020773_0015,TT020829_0007,TT020963_0006,TT021092_0102,TT021097_0062,TT021454_0004,TT021600_0062,TT023646_0056,TT023648_0051,TT023649_0031,TT026282_0002
Short-Term	TT020003_0014,TT020027_0002,TT020033_0002,TT020037_0020,TT020046_0013,TT020048_0012,TT020054_0013,TT020059_0006,TT020086_0014,TT020106_0006,TT020122_0022,TT020141_0041,TT020145_0006,TT020149_0028,TT020156_0003,TT020160_0008,TT020162_0025,TT020175_0003,TT020210_0013,TT020213_0003,TT020620_0071,TT020646_0062,TT020648_0030,TT020649_0007,TT021098_0057,TT021459_0003,TT021794_0005,TT021795_0003,TT022624_0004,TT024926_0006,TT024934_0004,TT025412_0045,TT026656_0010

Figure B.1: The 68 Cases Used in Chapter 4

Appendix C: Copyright Permissions

The permissions below are for material in chapters 5, 6, 7,8 and 9 respectively.

5/5/22, 1:53 PM Rightslink® by Copyright Clearance Center

CCC RightsLink®

Home | Help | Live Chat | Sign in | Create Account

IEEE Requesting permission to reuse content from an IEEE publication

Exploring deep features from brain tumor magnetic resonance images via transfer learning

Conference Proceedings: 2016 International Joint Conference on Neural Networks (IJCNN)
Author: Renhao Liu
Publisher: IEEE
Date: July 2016

Copyright © 2016, IEEE

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

© 2022 Copyright - All Rights Reserved | Copyright Clearance Center, Inc. | Privacy statement | Terms and Conditions
Comments? We would like to hear from you. E-mail us at customercare@copyright.com

RE: Copyright request

Karleena Burdick <karleenab@spie.org>
Fri 5/6/2022 4:29 PM

To: Kaoutar Ben Ahmed <kbenahmed@usf.edu>

Dear Kaoutar,

Thank you for seeking permission from SPIE to reprint material from our publications. SPIE shares the copyright with you, so as author you retain the right to reproduce your paper in part or in whole.

Publisher's permission is hereby granted under the following conditions:
(1) the material to be used has appeared in our publication without credit or acknowledgment to another source; and
(2) you credit the original SPIE publication. Include the authors' names, title of paper, volume title, SPIE volume number, and year of publication in your credit statement.

Best,
Karleena Burdick
Editorial Assistant, Publications
SPIE – the international society for optics and photonics
karleenab@spie.org
1 360 685 5515
SPIE.

From: Kaoutar Ben Ahmed <kbenahmed@usf.edu>
Sent: Friday, May 6, 2022 9:30 AM
To: reprint_permission <reprint_permission@spie.org>
Subject: Copyright request

Dear Sir/Madam,

I am requesting a copyright reprint permission for my article and I want to use some figures and tables in my PhD dissertation. Please check the following information.

Title and authors:
"Fine-tuning convolutional deep features for MRI based brain tumor classification"
Kaoutar B. Ahmed, Lawrence O. Hall, Dmitry B. Goldgof, Renhao Liu, Robert A. Gatenby

Proceedings Volume 10134, Medical Imaging 2017: Computer-Aided Diagnosis; 101342E (2017) https://doi.org/10.1117/12.2253982
Event: SPIE Medical Imaging, 2017, Orlando, Florida, United States

What you would like to reproduce:
Tables, Figures and descriptions of the proposed approach

Where you will republish the requested material:
Will going to add in my PhD dissertation.

I would be glad if you can send me the copyright reprint permission for my article.

RightsLink®

Home | Help | Live Chat | Sign in | Create Account

IEEE Requesting permission to reuse content from an IEEE publication

Neuroimaging Based Survival Time Prediction of GBM Patients Using CNNs from Small Data

Conference Proceedings:
2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)

Author: Kaoutar Ben Ahmed

Publisher: IEEE

Date: Oct. 2019

Copyright © 2019, IEEE

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

© 2022 Copyright - All Rights Reserved | Copyright Clearance Center, Inc. | Privacy statement | Terms and Conditions
Comments? We would like to hear from you. E-mail us at customercare@copyright.com

About the Author

Kaoutar Ben Ahmed is a PhD candidate in the Department of Computer Science and Engineering, University of South Florida. Her research interests include artificial intelligence, machine learning, medical imaging, bioinformatics, and deep learning. Kaoutar is graduating in Fall 2022 and plans to continue her research studies to deliver AI-powered solutions for healthcare.