Understanding Deep Learning Techniques for Image Segmentation

Swarnendu Ghosh, Nibaran Das, Ishita Das, Ujjwal Maulik

November 15, 2019

Abstract

The machine learning community has been overwhelmed by a plethora of deep learning based approaches. Many challenging computer vision tasks such as detection, localization, recognition and segmentation of objects in unconstrained environment are being efficiently addressed by various types of deep neural networks like convolutional neural networks, recurrent networks, adversarial networks, autoencoders and so on. While there have been plenty of analytical studies regarding the object detection or recognition domain, many new deep learning techniques have surfaced with respect to image segmentation techniques. This paper approaches these various deep learning techniques of image segmentation from an analytical perspective. The main goal of this work is to provide an intuitive understanding of the major techniques that has made significant contribution to the image segmentation domain. Starting from some of the traditional image segmentation approaches, the paper progresses describing the effect deep learning had on the image segmentation domain. Thereafter, most of the major segmentation algorithms have been logically categorized with paragraphs dedicated to their unique contribution. With an ample amount of intuitive explanations, the reader is expected to have an improved ability to visualize the internal dynamics of these processes.

1 Introduction

Image segmentation can be defined as a specific image processing technique which is used to divide an image into two or more meaningful regions. Image segmentation can also be seen as a process of defining boundaries between separate semantic entities in an image. From a more technical perspective, image segmentation is a process of assigning a label to each pixel in the image such that pixels with the same label are connected with respect to some visual or semantic property (Fig. 1).

Image segmentation subsumes a large class of finely related problems in computer vision. The most classic version is semantic segmentation [66]. In semantic segmentation, each pixel is classified into one of the predefined set

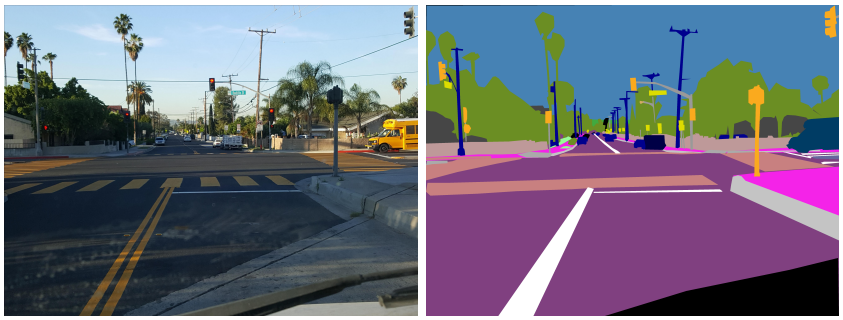

Figure 1: Semantic Image Segmentation(Samples from the Mapillary Vistas Dataset [155])

of classes such that pixels belonging to the same class belongs to an unique semantic entity in the image. It is also worthy to note that the semantics in question depends not only on the data but also the problem that needs to be addressed. For example, for a pedestrian detection system, the whole body of person should belong to the same segment, however for a action recognition system, it might be necessary to segment different body parts into different classes. Other forms of image segmentation can focus on the most important object in a scene. A particular class of problem called saliency detection [19] is born from this. Other variants of this domain can be foreground background separation problems. In many systems like, image retrieval or visual question answering it is often necessary to count the number of objects. Instance specific segmentation addresses that issue. Instance specific segmentation is often coupled with object detection systems to detect and segment multiple instances of the same object[43] in a scene. Segmentation in the temporal space is also a challenging domain and has various application. In object tracking scenarios, pixel level classification is not only performed in the spatial domain but also across time. Other applications in traffic analysis or surveillance needs to perform motion segmentation to analyze paths of moving objects. In the field of segmentation with lower semantic level, over-segmentation is also a common approach where images are divided into extremely small regions to ensure boundary adherence, at the cost of creating a lot of spurious edges. Over-segmentation algorithms are often combined with region merging techniques to perform image segmentation. Even simple color or texture segmentation also finds its use in various scenarios. Another important distinction between segmentation algorithms is the need of interactions from the user. While it is desirable to have fully automated systems, a little bit of interaction from the user can improve the quality of segmentation to a large extent. This is especially applicable where we are dealing with complex scenes or we do not posses an ample amount of data to train the system.

Segmentation algorithms has several applications in the real world. In medical image processing [123] as well we need to localize various abnormalities

like aneurysms [48], tumors [145], cancerous elements like melanoma detection [189], or specific organs during surgeries [206]. Another domain where segmentation is important is surveillance. Many problems such as pedestrian detection [113], traffic surveillance [60] require the segmentation of specific objects e.g. persons or cars. Other domains include satellite imagery [11, 17], guidance systems in defense [119], forensics such as face [5], iris [51] and fingerprint [144] recognition. Generally traditional methods such as histogram thresholding [195], hybridization [193, 87] feature space clustering [40], region-based approaches [59], edge detection approaches [184], fuzzy approaches [39], entropy-based approaches [47], neural networks (Hopfield neural network [35], self-organizing maps [27]), physics-based approaches [158] etc. are used popularly in this purpose. However, such feature-based approaches have a common bottleneck that they are dependent on the quality of feature extracted by the domain experts. Generally, humans are bound to miss latent or abstract features for image segmentation. On the other hand, deep learning in general addresses this issue of automated feature learning. In this regard one of the most common technique in computer vision was introduced soon by the name of convolutional neural networks [110] that learned a cascaded set of convolutional kernels through backpropagation [182]. Since then, it has been improved significantly with features like layer-wise training [13], rectified linear activations [153], batch normalization [84], auxiliary classifiers [52], atrous convolutions [211], skip connections [78], better optimization techniques [97] and so on. With all these there was a large number of new types of image segmentation techniques as well. Various such techniques drew inspiration from popular networks such as AlexNet [104], convolutional autoencoders [141], recurrent neural networks [143], residual networks [78] and so on.

2 Motivation

There have been many reviews and surveys regarding the traditional technologies associated with image segmentation [61, 160]. While some of them specialized in application areas [107, 123, 185], while other focused on specific types of algorithms [20, 19, 59]. With arrival of deep learning techniques many new classes of image segmentation algorithms have surfaced. Earlier studies [219] have shown the potential of deep learning based approaches. There have been more recent studies [68] which cover a number of methods and compare them on the basis of their reported performance. The work of Garcia et al. [66] lists a variety of deep learning based segmentation techniques. They have tabulated the performance of various state of the art networks on several modern challenges. The resources are incredibly useful for understanding the current state-of-the-art in this domain. While knowing the available methods is quite useful to develop products, however, to contribute to this domain as a researcher, one needs to understand the underlying mechanics of the methods that make them confident. In the present work, our main motivation is to answer the question why the methods are designed in a way they are. Understanding the mechanics

of modern techniques would make it easier to tackle new challenges and develop better algorithms. Our approach carefully analyses each method to understand why they succeed at what they do and also why they fail for certain problems. Being aware of pros and cons of such method new designs can be initiated that reaps the benefits of the pros and overcomes the cons. We recommend the works of Alberto Garcia-Garcia [66] to get an overview of some of the best image segmentation techniques using deep learning while our focus would be to understand why, when and how these techniques perform on various challenges.

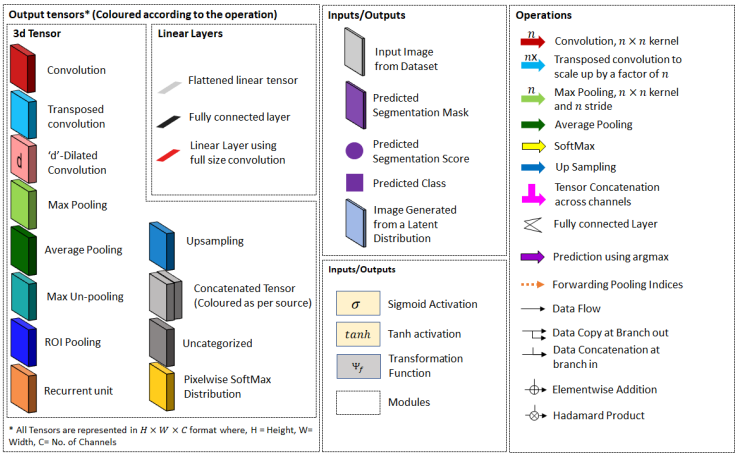

Figure 2: Legends for subsequent diagrams of popular deep learning architectures

2.1 Contribution

The paper has been designed in a way such that new researchers reap the most benefits. Initially some of the traditional techniques have been discussed to uphold the frameworks before the deep learning era. Gradually the various factors governing the onset of deep learning has been discussed so that readers have a good idea of the current direction in which machine learning is progressing. In the subsequent sections the major deep learning algorithms have been briefly described in a generic way to establish a clearer concept of the procedures in the mind of the readers. The image segmentation algorithms discussed thereafter have been categorized into the major families of algorithms that governed the last few years in this domain. The concepts behind all the major approaches have been explained through a very simple language with minimum amount of complicated mathematics. Almost all the diagrams corresponding to major networks have been drawn using a common representational format as shown in fig. 2. The various approaches that have been discussed comes with different representations for architectures. The unified representation scheme allows the

user to understand the fundamental similarities and differences between networks. Finally, the major application areas have been discussed to help new researchers pursue a field of their choice.

3 Impact of Deep Learning on Image Segmentation

The development of deep learning algorithms like convolutional neural networks or deep autoencoders not only affected typical tasks like object classification but are also efficient in other related tasks like object detection, localization, tracking, or as in this case image segmentation.

3.1 Effectiveness of convolutions for segmentation

As an operation convolution can be simply defined as the function that performs a sum-of-product between kernel weights and input values while convoluting the smaller kernel over a larger image. For a typical image with k channels we can convolute a smaller sized kernel with k channels along the x and y direction to obtain an output in the format of a 2 dimensional matrix. It has been observed that after training a typical CNN the convolutional kernels tend to generate activation maps with respect to certain features of the objects [214]. Given the nature of activations, it can be seen as segmentation masks of object specific features. Hence the key to generating requirement specific segmentation is already embedded within this output activation matrices. Most of the image segmentation algorithm uses this property of CNNs to somehow generate the segmentation masks as required to solve the problem. As shown below in fig. 3, the earlier layers capture local features like the contour or a small part of an object. In the later layers more global features are activated such as field, people or sky. It can also be noted from this figure that the earlier layers show sharper activations as compared to the later ones.

3.2 Impact of larger and more complex datasets

The second impact that deep learning brought to the world of image segmentation is the plethora of datasets, challenges and competitions. These factors encouraged researchers across the world to come up with various state-of-the-art technologies to implement segmentation across various domains. A list of many such datasets have been provided in table 1

4 Image Segmentation using Deep Learning

As explained before, convolutions are quite effective in generating semantic activation maps that has components which inherently constitute various semantic

Table 1: A list of various datasets in the image segmentation domain

Category	**Dataset**
Natural Scenes	Berkeley Segmentation Dataset [140]
	PASCAL VOC [54]
	Stanford Background Dataset [72]
	Microsoft COCO [122]
	MIT Scene parsing data(ADE20K) [222, 223]
	Semantic Boundaries Dataset [75]
	Microsoft Research Cambridge Object Recognition Image Database (MSRC) [188]
Video Segmentation Dataset	Densely Annotated Video Segmentation(DAVIS) [168]
	Video Segmentation Benchmark(VSB100) [64]
	YouTube-Video object Segmentation [209]
Autonomous Driving	Cambridge-driving Labeled Video Database (CamVid) [23]
	Cityscapes: Semantic Urban Scene Understanding [41]
	Mapillary Vistas Dataset [155]
	SYNTHIA: Synthetic collection of Imagery and Annotations [178]
	KITTI Vision Benchmark Suite [67]
	Berkeley Deep Drive [212]
	India Driving Dataset(IDD) [202]
Aerial Imaging	Inria Aerial Image Labeling Dataset [134]
	Aerial Image Segmentation Dataset [213]
	ISPRS Dataset collection [57]
	Google Open Street Map [8]
	DeepGlobe [49]
Medical Imaging	DRIVE:Digital Retinal Images for Vessel Extraction [191]
	Sunnybrook Cardiac Data [171]
	Multiple Sclerosis Database [129, 25]
	IMT: Intima Media Thickness Segmentation Dataset [148]
	SCR: Segmentation in Chest Radiographs [201]
	BRATS: Brain Tumor Segmentation [146]
	LITS: Liver Tumour Segmentation [74]
	BACH: Breast Cancer Histology [6]
	IDRiD: Indian Diabetic Retinopathy Image Dataset [169]
	ISLES: Ischemic Stroke Lesion Segmentation [135]
Saliency Detection	MSRA Salient Object Database [37]
	ECSSD: Extended Complex Scene Saliency Dataset [187]
	PASCAL-S DATASET [117]
	THUR15K: Group Saliency in Image [36]
	JuddDB: MIT saliency benchmark [18]
	DUT-OMRON Image Dataset [210]
Scene Text Segmentation	KAIST Scene Text Database [112]
	COCO-Text [203]
	SVT: Street View Text Dataset [205]

Figure 3: Input image and sample activation maps from a typical CNN. (Top row) Input image and two activation maps from earlier layers showing part objects like t-shirts and features like contours. (Bottom row) shows activation maps from later layers with more meaningful activations like fields, people and sky respectively

segments. Various methods have been implemented to make use of these internal activations to segment the images. A summary of major deep learning based segmentation algorithms are provided in table 2 along with brief description of their major contribution.

4.1 Convolutional Neural Networks

Convolutional neural networks being one of the most commonly used methods in computer vision has adopted many simple modifications to perform well in segmentation tasks as well.

4.1.1 Fully convolutional layers

Classification tasks generally require a linear output in the form of a probability distribution over the number of classes. To convert volumes of 2 dimensional activation maps into linear layers they were often flattened. The flattened shape allowed the execution of fully connected networks to obtain the probability distribution. However, this kind of reshaping loses the spatial relations among the pixels in the image. In a fully convolutional neural network(FCN) [130] the output of the last convolutional block is directly used for a pixel level classification. FCNs were first implemented on the PASCAL VOC 2011 segmentation dataset[54] and achieved a pixel accuracy of 90.3% and a mean IOU of 62.7%. Another way to avoid fully connected linear layers is the use of a full size average pooling to convert a set of 2 dimensional activation maps to a set of scalar

Table 2: A summary of major deep learning based segmentation algorithms. Abbreviations: S: Supervised, W: Weakly supervised, U: Unsupervised, I: Interactive, P: Partially Supervised, SO: Single objective optimization, MO: Multi objective optimization, AD: Adversarial Learning, SM: Semantic Segmentation, CL: Class specific Segmentation, IN: Instance Segmentation, RNN: Recurrent Modules, E-D: Encoder Decoder Architecture

Method	Year	Supervision					Learning			Type			Modules		Description
		S	W	U	I	P	SO	MO	AD	SM	CL	IN	RNN	E-D	
Global Average Pooling	2013		✓				✓				✓				Object specific soft segmentation
DenseCRF	2014					✓	✓			✓					Using CRF to boost segmentation
FCN	2015	✓					✓			✓					Fully convolutional layers
DeepMask	2015	✓						✓			✓				Simultaneous learning for segmentation and classification
U-Net	2015	✓					✓			✓				✓	Encoder-Decoder with multiscale feature concatenation
SegNet	2015	✓					✓			✓				✓	Encoder-Decoder with forwarding pooling indices
CRF as RNN	2015	✓						✓		✓			✓		Simulating CRFs as trainable RNN modules
Deep Parsing Network	2015	✓						✓							Using unshared kernels to incorporate higher order dependency
BoxSup	2015		✓							✓					Using bounding box for weak supervision
SharpMask	2016	✓						✓			✓			✓	Refined Deepmask with multi layer feature fusion
Attention to Scale	2016	✓					✓			✓					Fusing features from multi scale inputs
Semantic Segmentation	2016	✓							✓	✓					Adversarial training for image segmentation
Conv LSTM and Spatial Inhibition	2016	✓						✓				✓	✓		Using spatial inhibition for instance segmentation
JULE	2016			✓				✓		✓			✓		Joint unsupervised learning for segmentation
ENet	2016	✓					✓			✓					Compact network for realtime segmentation
Instance aware segmentation	2016	✓						✓				✓			Multi task approach for instance segmentation
Mask RCNN	2017	✓						✓		✓					Using region propsal network for segmentation
Large Kernel Matters	2017	✓					✓			✓				✓	Using larger kernels for learning complex features
RefineNet	2017	✓					✓			✓				✓	Multi path refinement module for fine segmentation
PSPNet	2017	✓					✓			✓					Multi scale pooling for scale agnostic segmentation
Tiramisu	2017	✓					✓			✓				✓	DenseNet 121 feature extractor
Image to Image Translation	2017	✓							✓	✓				✓	Conditional GAN for translation image to segment maps
Instance Segmentation with attention	2017	✓						✓				✓	✓		Attention modules for image segmentatoin
W-Net	2017			✓				✓		✓				✓	Unsupervised segmentation using normalized cut loss
PolygonRNN	2017				✓		✓			✓			✓		Generating contours by RNN
Deep Layer Cascade	2017	✓					✓			✓					Multi level approach to handle pixels of different complexity
Spatial Propagation Network	2017	✓					✓			✓					Refinement using linear label propagation
DeepLab	2018	✓					✓			✓					Atrous convolution, Spatial pooling pyramid, DenseCRF
SegCaps	2018	✓						✓							Capsule Networks for Segmentation
Adversarial Collaboration	2018			✓				✓							Adversarial collaboration between multiple networks
Superpixel Supervision	2018			✓				✓							Using superpixel refinement as supervisory signals
Deep Extreme Cut	2018				✓		✓			✓					Using extreme points for interactive segmentation
Two Stream Fusion	2019				✓		✓			✓					Using image stream and interaction stream simultaneously
SegFast	2019	✓					✓			✓				✓	Using depth-wise separable convolution in SqueezeNet encoder

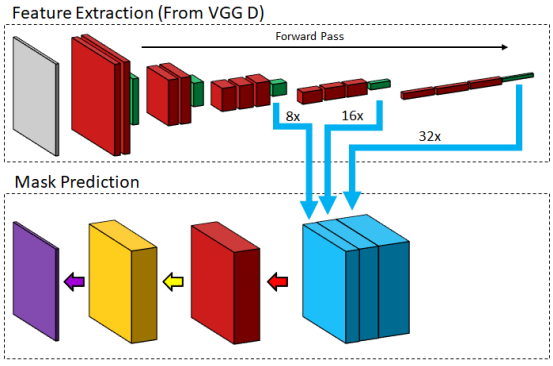

Figure 4: A Fully convolutional network with image segmentation with concatenated multi-scale features

values. As these pooled scalars are connected to the output layers, the weights corresponding to each class may be used to perform weighted summation of the corresponding activation maps in the previous layers. This process called Global Average Pooling(GAP) [121] can be directly used on various trained networks like residual network to find object specific activation zones which can be used for pixel level segmentation. The major issues with algorithm such as this is the loss of sharpness due to the intermediate sub-sampling operations. Sub-sampling is a common operation in convolutional neural networks to increase the sensory area of kernels. What it means is that as the activations maps reduces in size in the subsequent layers, the kernels convoluting over them actually corresponds to a larger area in the original image. However, it reduces the image size in the process, which when up-sampled to original size loses sharpness. Many approaches have been implemented to handle this issue. For fully convolutional models, skip connections from preceding layers can be used to obtain sharper versions of the activations from which finer segments can be chalked out (Refer fig. 4). Another work showed how the usage of high dimensional kernels to capture global information with FCN models created better segmentation masks [165]. Segmentation algorithms can also be treated as boundary detection technique. convolutional features are also very useful from that perspective [139]. While earlier layers can provide fine details, later layers focus more on the coarser boundaries.

DeepMask and SharpMask DeepMask [166] was a name given to a project at Facebook AI Research (FAIR) related to image segmentation. It exhibited the same school of thought as FCN models except that the model was capable of multi-tasking (Refer fig. 5). It had two main branches coming out of a

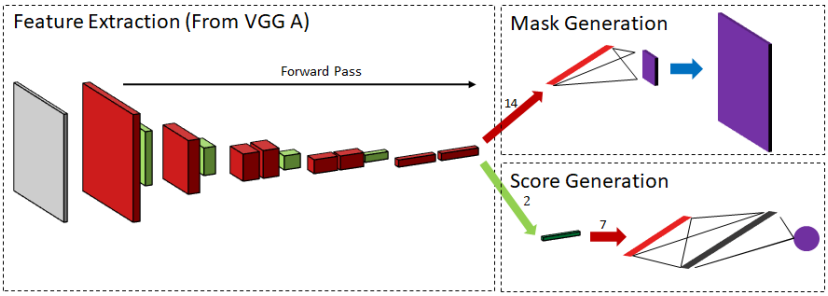

Figure 5: The Deepmask Network

shared feature representation. One of them created a pixel level classification of or a probabilistic mask for the central object and the second branch generated a score corresponding to the object recognition accuracy. The network coupled with sliding windows of sixteen strides to create segments of objects at various locations of the image, whereas the score helped in identifying which of the segments were good. The network was further upgraded in SharpMask [167], where probabilistic masks from each layer were combined in top-down fashion

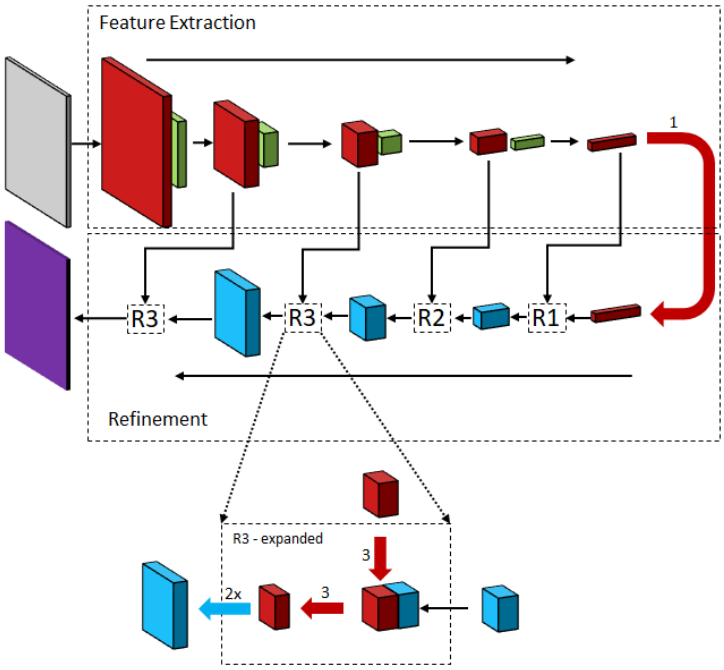

Figure 6: The Sharpmask Network

using convolutional refinements at every steps to generate high resolution masks (Refer fig. 6). The sharpmask scored an average recall of 39.3 which beats deepmask, which scored 36.6 on the MS COCO Segmentation Dataset.

4.1.2 Region proposal networks

Another similar wing that started developing with image segmentation was object localization. Task such as this involved locating specific objects in images. Expected outputs for such problems is normally a set of bounding boxes corresponding to the queried objects. Though strictly stating, some of these algorithms do not address image segmentation problems, however their approaches are of relevance to this domain.

RCNN (Region-based Convolutional Neural Networks) The introduction of the CNNs raised many new questions in the domain of computer vision. One of them primarily being whether a network like AlexNet can be extended to detect the presence of more than one object. Region-based-CNN [70] or more commonly known as R-CNN used selective search technique to propose probable object regions and performed classification on the cropped window to verify sensible localization based on the output probability distribution. Selective search technique [198, 200] analyses various aspects like texture, color, or

intensities to cluster the pixels into objects. The bounding boxes corresponding to these segments are passed through classifying networks to short-list some of the most sensible boxes. Finally, with a simple linear regression network tighter co-ordinate can be obtained. The main downside of the technique is its computational cost. The network needs to compute a forward pass for every bounding box proposition. The problem with sharing computation across all boxes was that the boxes were of different sizes and hence uniform sized features were not achievable. In the upgraded Fast R-CNN [69], ROI (Region of Interest) Pooling was proposed in which region of interests were dynamically pooled to obtain a fixed size feature output. Henceforth, the network was mainly bottlenecked by the selective search technique for candidate region proposal. In Faster-RCNN [175], instead of depending on external features, the intermediate activation maps were used to propose bounding boxes, thus speeding up the feature extraction process. Bounding boxes are representative of the location of the object, however they do not provide pixel-level segments. The Faster R-CNN network was extended as Mask R-CNN [76] with a parallel branch that performed pixel level object specific binary classification to provide accurate segments. With Mask-RCNN an average precision of 35.7 was attained in the COCO[122] test images. The family of RCNN algorithms have been depicted in fig.7. Region proposal networks have often been combined with other networks [118, 44] to give instance level segmentations. RCNN was further improved under the name of HyperNet [99] by using features from multiple layers of the feature extractor. Region proposal networks have also been implemented for instance specific segmentation as well. As mentioned before object detection capabilities of approaches like RCNN are often coupled with segmentation models to generate different masks for different instances of the same object[43].

4.1.3 DeepLab

While pixel level segmentation was effective, two complementing issues were still affecting the performance. Firstly, smaller kernel sizes failed to capture contextual information. In classification problems, this is handled using pooling layers that increases the sensory area of the kernels with respect to the original image. But in segmentation that reduces the sharpness of the segmented output. Alternative usage of larger kernels tend to be slower due to significanty larger number of trainable parameters. To handle this issue the DeepLab [30, 32] family of algorithms demonstrated the usage of various methodologies like atrous convolutions [211], spatial pooling pyramids [77] and fully connected conditional random fields [100] to perform image segmentation with great efficiency. The DeepLab algorithm was able to attain a meanIOU of 79.7 on the PASCAL VOC 2012 dataset[54].

Atrous/Dilated Convolution The size of the convolution kernels in any layer determine the sensory response area of the network. While smaller kernels extract local information, larger kernels try to focus on more contextual information. However, larger kernels normally comes with more number of parameters.

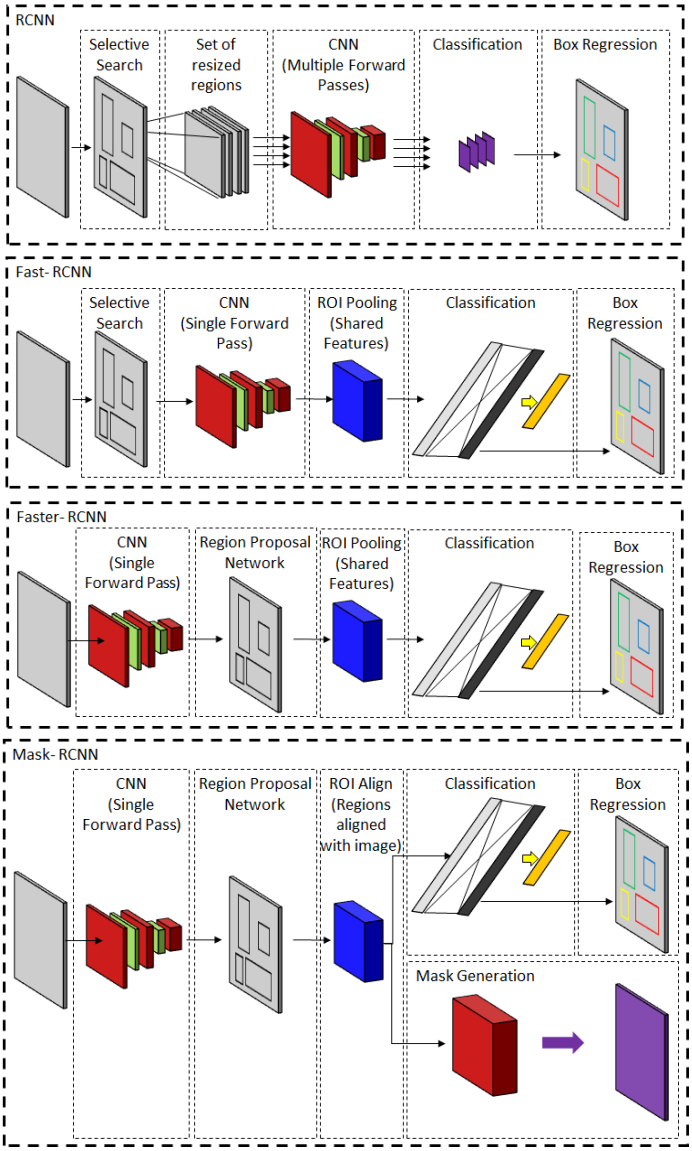

Figure 7: The RCNN Family of localization and segmentation networks

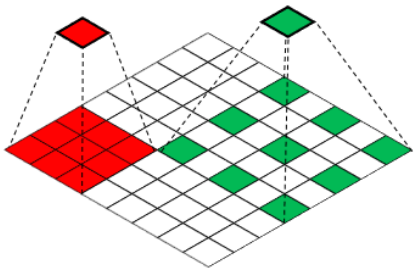

Figure 8: Normal convolution(red) vs. Atrous or Dilated convolution(green)

For example to have a sensory region of 6×6, one must have 36 neurons. To reduce the number of parameters in the CNN, the sensory area is increased in higher layers through techniques like pooling. Pooling layers reduce the size of the image. When an image is pooled by a 2×2 kernel with a stride of two, the size of the image reduces by 25%. A kernel with an area of 3×3 corresponds to a larger sensory area of 6×6 in the original image. However, unlike before now only 18 neurons (9 for each layer) are needed in the convolution kernel. In case of segmentation, pooling creates new problems. The reduction in the image size results in loss of sharpness in generated segments as the reduced maps are scaled up to image size. To deal with these two issues simultaneously, dilated or atrous convolutions play a key role. Atrous/Dilated convolutions increase the field of view without increasing the number of parameters. As shown in fig.8 a 3×3 kernel with a dilation factor of 1 can act upon an area of 5×5 in the image. Each row and column of the kernel has three neurons which is multiplied with intensity values in the image which separated by the dilation factor of 1. In this way the kernels can span over larger areas while keeping the number of neurons low and also preserving the sharpness of the image. Besides the DeepLab algorithms, atrous convolutions [34] have also been used with auto encoder based architectures.

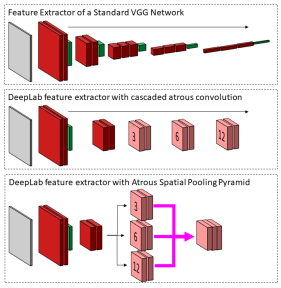

Figure 9: DeepLab Architecture as compared to a standard VGG net(top) along with cascaded atrous convolutions (middle) and atrous spatial pooling pyramid(bottom)

Spatial Pyramid Pooling Spatial pyramid pooling [77] was introduced in R-CNN where ROI pooling showed the benefit of using multi-scale regions for object localization. However, in DeepLab, atrous convolutions were preferred over pooling layers for changing field of view or sensory area. To imitate the effect of ROI pooling, multiple branches with atrous convolutions of different dilations were combined together to utilize multi-scale properties for image segmentation.

Fully connected conditional random field Conditional random field is a undirected discriminative probabilistic graphical model that is often used for various sequence learning problems. Unlike discrete classifiers, while classifying a sample it takes into account the labels of other neighboring samples. Image segmentation can be treated as a sequence of pixel classifications. The label of a pixel is not only dependent on its own intensity values but also the values of neighboring pixels. The use of such probabilistic graphical models is often used in the field of image segmentation and hence it deserves a dedicated section (section 4.1.4).

4.1.4 Using inter pixel correlation to improve CNN based segmentation

The use of probabilistic graphical models such as markov random fields (MRF) or conditional random fields (CRF) for image segmentation thrived on its own even without the inclusion of CNN based feature extractors. The CRF or MRF is mainly characterized by an energy function with a unary and a pairwise component.

$$E(x) = \underbrace{\sum_{i} \theta_i(x_i)}_{\text{unary potential}} + \underbrace{\sum_{ij} \theta_{ij}(x_i, x_j)}_{\text{pairwise potential}} \tag{1}$$

While non-deep learning approaches focused on building efficient pair-wise potentials like exploiting long-range dependencies, designing higher-order potentials and exploring contexts of semantic labels, deep learning based approaches focused on generating a strong unary potentials and using simple pairwise components to boost the performance. CRFs have usually been coupled with deep learning based methods in two ways. One as a separate post-processing module and the other as an trainable module in an end-to-end network like deep parsing networks[128] or spatial propagation networks[126].

Using CRFs to improve Fully convolutional networks One of the earliest implementations that kick-started this paradigm of boundary refinements was the works of [101] With the introduction of fully convolutional networks for image segmentation it was quite possible to draw coarse segments for objects in images. However, getting sharper segments was still a problem. In the works of [29], the output pixel level prediction was used as a unary potential for a

fully connected CRF. For each pair of pixels i and j in the image the pairwise potential was defined as

$$\theta_{ij}(x_i, x_j) = \mu(x_i, x_j) \left[w_1 \exp\left(-\frac{||p_i - p_j||^2}{2\sigma_\alpha^2} - \frac{||I_i - I_j||^2}{2\sigma_\beta^2} \right) + w_2 \exp\left(-\frac{||p_i - p_j||^2}{2\sigma_\gamma^2} \right) \right] \quad (2)$$

Here, $\mu(x_i, x_j) = 1$ if $x_i \neq x_j, 0$ otherwise and w_1, w_2 are the weights given to the kernels. The expression uses two gaussian kernels. The first one is a bilateral kernel that depends on both pixel positions(p_i, p_j) and their corresponding intensities in the RGB channels. The second kernel is only dependent on the the pixel positions. $\sigma_\alpha, \sigma_\beta$ and σ_γ controls the scale of the Gaussian kernels. The intuition behind the design of such a pairwise potential energy function is to ensure that nearby pixels of similar intensities in the RGB channels are classified under the same class. This model has also been later included in the popular network called DeepLab (refer section 4.1.3). In the various versions of the DeepLab algorithm, the use of CRF was able to boost the mean IOU on the Pascal 2012 Dataset by significant amount(upto 4% in some cases).

CRF as RNN While CRF is an useful post-processing module[101] for any deep learning based semantic image segmentation architecture, yet one of the main drawbacks was that it could not be used as a part of an end-to-end architecture. In the standard CRF model the pairwise potentials can be represented in terms of a sum of weighted Gaussians. However since the exact minimization is intractable a mean-field approximation of the CRF distribution is considered to represent the distribution with a simpler version which is simply a product of independent marginal distributions. This mean-field approximation in its native form isn't suitable for back-propagation. In the works of [221], this step was replaced by a set of convolutional operation that is iterated over a recurrent pipeline until convergence is reached. As reported in their work, with the proposed approach an mIOU of 74.7 was obtained as compared to 71.0 by BoxSup and 72.7 by DeepLab. The sequence of operations can be most easily explained as follows.

1. Initialization : A SoftMax operations over the unary potentials can give us the intial distribution to work with.
2. Message Passing : Convoluting using two Gaussian kernels, one spatial and one bilateral kernel. Similar to the actual implementation of CRF, the splatting and slicing also occurs while building the permutohedral lattice for efficient computation of the fully connected CRF
3. Weighting Filter Outputs : Convoluting with 1×1 kernels with the required number of channels the filter outputs can be weighted and summed. The weights can be easily learnt through backpropagation.

4. Compatibility Transform : Considering a compatibility function to keep a track of uncertainty between various labels, a simple 1×1 convolution with the same number of input and output channel is enough to simulate that. Unlike the potts model that assigns the same penalty, here the compatibility function can be learnt and hence a much better alternative.

5. Adding the unary potentials : This can be performed by a simple element wise subtraction of the penalty from the compatibility transform from the unary potentials

6. Normalization : The outputs can be normalized with another simple softmax function.

Incorporating higher order dependencies Another end-to-end network inspired from CRFs, incorporate higher order relations into a deep network . With a deep parsing network [128] pixel-wise prediction from a standard VGG-like feature extractor (but with lesser pooling operations) is boosted using a sequence of special convolution and pooling operations. Firstly , by using local convolutions that implement large unshared convolutional kernels across the different positions of the feature map, to obtain translation dependent features that model long-distance dependencies. Similar to standard CRFs a spatial convolution penalizes probabilistic maps based on local label contexts. Finally, with block min pooling that does a pixel-wise min-pooling across the depth to accept the prediction with the lowest penalty. Similarly, in the works of [126], a row/columnwise propagation model was proposed the calculated the global pairwise relationship across an image. With a dense affinity matrix drawn from a sparse transformation matrix, coarsely predicted labels were reclassified based on the affinity of pixels.

4.1.5 Multi-scale networks

One of the main problems with image segmentation for natural scene images is that the size of the object of interest is very unpredictable, as in real world objects may be of different sizes and objects may look bigger or smaller depending on the position of the object and the camera. The nature of a CNN dictates that delicate small scale features are captured in early layers whereas as one moves across the depth of the network the features become more specific for larger objects. For example a tiny car in a scene has much lesser chance of being captured in the higher layers due to operations like pooling or down-sampling. It is often beneficial to extract information from feature maps of various scales to create segmentations that are agnostic of the size of the object in the image. Multi-scale auto-encoder models [33] consider activations of different resolutions to provide image segmentation output.

PSPNet The pyramid scene parsing network [220] was built upon the FCN based pixel level classification network. The feature maps from a ResNet-101

network are converted to activations of different resolutions thorough multi-scale pooling layers which are later upsampled and concatenated with the orginal feature map to perform segmentation(Refer fig.10). The learning process in deep networks like ResNet was further optimized by using auxiliary classifiers. The different types of pooling modules focus on different areas of the activation map. Pooling kernels of various sizes like 1×1, 2×2, 3×3, 6×6 look into different areas of the activation map to create the spatial pooling pyramid. One the ImageNet scene parsing challenge the PSPNet was able to score an mean IoU of 57.21 with respect to 44.80 of FCN and 40.79 of SegNet.

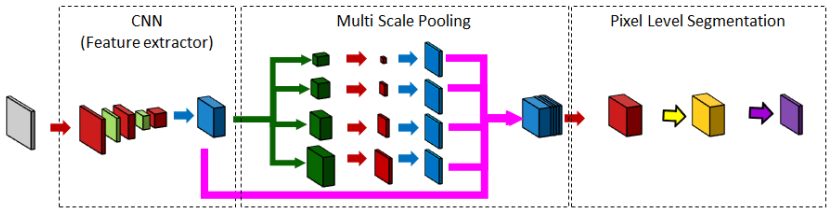

Figure 10: A schematic representation of the PSPNet

RefineNet Working with features from last layer of a CNN produces soft boundaries for the object segments. This issue was avoided in DeepLab algorithms with atrous convolutions. RefineNet [120] takes an alternative approach by refining intermediate activation maps and hierarchically concatenating it to combine multi-scale activations and prevent loss of sharpness simultaneously. The network consisted of separate RefineNet modules for each block of the ResNet. Each RefineNet module were made up of three main blocks, namely, Residual convolution unit(RCU), multi-resolution fusion(MRF) and chained residual pooling(CRP)(Refer fig.11). The RCU block consists of an adaptive convolution set that fine-tunes the pre-trained weights of the ResNet weights for the segmentation problem. The MRF layer fuses activations of different resolutions using convolutions and upsampling layers to create a higher resolution map. Finally in CRP layer pooling kernels of multiple sizes are used on the activations to capture background context from large image areas. The RefineNet was tested on the Person-Part Dataset where it obtained an IOU of 68.6 as compared to 64.9 by DeepLab-v2 both of which used the ResNet-101 as a feature extractor.

4.2 Convolutional autoencoders

The last subsection deals with discriminative models that are used to perform pixel level classification to deal with image segmentation problems. Another line of thought gets its inspiration from autoencoders. Autoencoders have been traditionally used for feature extraction from input samples while trying to retain most of the original information. An autoencoder is basically composed of

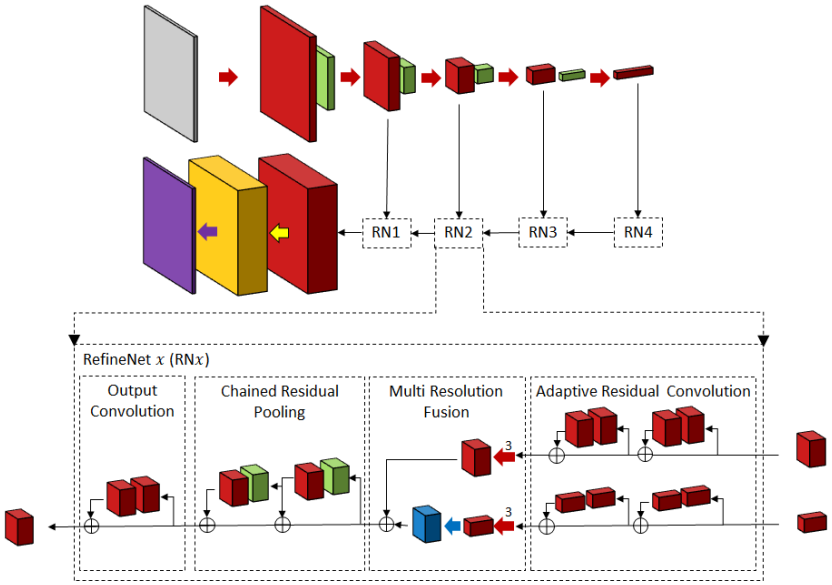

Figure 11: A schematic representation of the RefineNet

an encoder that encodes the input representations from a raw input to a possibly lower dimensional intermediate representation and a decoder that attempts to reconstruct the original input from the intermediate representation. The loss is computed in terms of the difference between the raw input images and the reconstructed output image. The generative nature of the decoder part has often been modified and used for image segmentation purposes. Unlike the traditional autoencoders, during segmentation the loss is computed in terms of the difference between the reconstructed pixel level class distribution and the desired pixel level class distribution. This kind of segmentation approach is more of a generative procedure as compared to the classification approach of RCNN or DeepLab algorithms. The problem with approaches such as this is to prevent over-abstraction of images during the encoding process. The primary benefit of such approaches is the ability to generate sharper boundaries with much lesser complication. Unlike the classification approaches, the generative nature of the decoder can learn to create delicate boundaries based on extracted features. The major issue that affects these algorithm is the level of abstraction. It has been seen that without proper modification the reduction in the size of the feature map created inconsistencies during the reconstruction. in the paradigm of convolutional neural networks the encoding is basically a series of convolution and pooling layers or strided convolutions. The reconstruction however can be tricky. The commonly used techniques for decoding from a lower dimensional feature are transposed convolution or a unpooling layers. One of the main advantages of using autoencoder based approach over normal convolutional feature extractor is the freedom of choosing input size. With a clever use of

down-sampling and up-sampling operation it is possible to output a pixel-level probability that is of the same resolution as the input image. This benefit has made encoder-decoder architectures with multi-scale feature forwarding has become ubiquitous for networks where input size is not predetermined and an output of same size as the input is needed.

Transposed Convolution Transposed convolution also known as convolution with fractional strides has been introduced to reverse the effects of a traditional convolution operation [156, 53]. It is often referred to as deconvolution. However deconvolution, as defined in signal processing, is different than transposed convolution in terms of the basic formulation, although they effectively address the same problem. In a convolution operation there is a change in size of the input based on the amount of padding and stride of the kernels. As shown in fig. 12 a stride of 2 will create half the number of activations as that of a stride of 1. For a transposed convolution to work padding and stride should be controlled in a way that the size change is reversed. This is achieved by dilating the input space. Note that unlike atrous convolutions, where the kernels were dilated, here the input spaces are dilated.

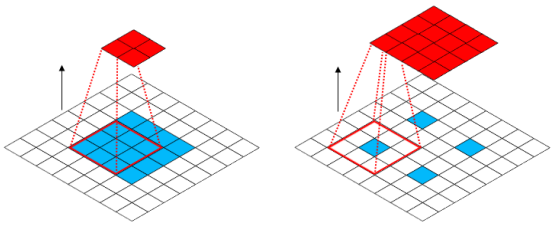

Figure 12: (Left) Normal Convolution with unit stride. (Right) Transposed convolution with fractional strides.

Unpooling Another approach to reduce the size of the activations is through pooling layers. a 2×2 pooling layer with a stride of two reduces the height and width of the image by a factor of 2. In such a pooling layer, a 2×2 neighborhood of pixel is compressed to a single pixel. Different types of pooling performs the compression in different ways. Max-pooling considers the maximum activation value among 4 pixels while average pooling takes an average of the same. A corresponding unpooling layer decompresses a single pixel to a neighborhood of 2×2 pixels to double the height and width of the image.

4.2.1 Skip Connections

Linear skip connections has often been used in convolutional neural networks to improve gradient flow across a large number of layers [78]. As depth increases in a network the activations maps tend to focus on more and more abstract

concepts. Skip connections has proved to be very effective to combine different levels of abstractions from different layers to generate crisp segmentation maps.

U-NET The U-Net architecture, proposed in 2015, proved to be quite efficient for a variety of problems such as segmentation of neuronal structures, radiography, and cell tracking challenges [177]. The network is characterized by an encoder with a series of convolution and max pooling layers. The decoding layer contains a mirrored sequence of convolutions and transposed convolutions. As described till now it behaves as a traditional auto-encoder. Previously it has been mentioned how the level of abstraction plays an important role in the quality of image segmentation. To consider various levels of abstraction U-Net implements skip connections to copy the uncompressed activations from encoding blocks to their mirrored counterparts among the decoding blocks as shown in the fig. 13. The feature extractor of the U-Net can also be upgraded to provide better segmentation maps. The network nicknamed "The one hundred layers Tiramisu" [88] applied the concept of U-Net using a dense-net based feature extractor. Other modern variations involve the use of capsule networks [183] along with locally constrained routing [108]. U-Net was selected as a winner for an ISBI cell tracking challenge. In the PhC-U373 dataset it scored a mean IoU of 0.9203 whereas the second best was at 0.83. In the DIC-HeLa dataset, it scored a mean IoU of 0.7756 which was significantly better than the second best approach which scored only 0.46.

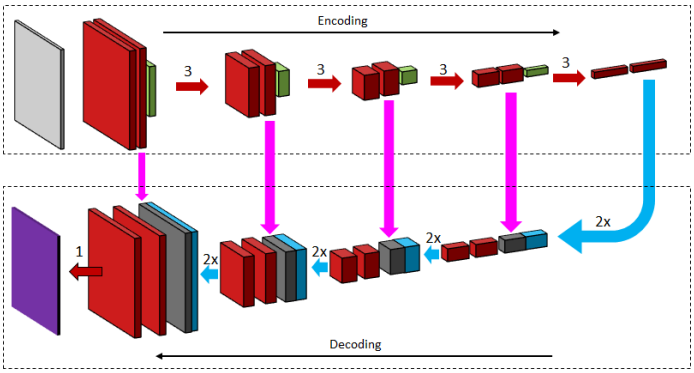

Figure 13: Architecture of U-Net

4.2.2 Forwarding pooling indices

Max-pooling has been the most commonly used technique for reducing the size of the activation maps for various reasons. The activations represent of the response of the region of an image to a specific kernel. In max pooling, a region of pixels is compressed to single value by considering only the maximum response obtained within that region. If a typical autoencoder compresses a 2×2 neighborhood of pixels to a single pixel in the encoding phase, the decoder must

decompress the pixel to a similar dimension of 2×2. By forwarding pooling indices the network basically remembers the location of the maximum value among the 4 pixels while performing max-pooling. The index corresponding to the maximum value is forwarded to the decoder(Refer fig.14) so that while the un-pooling operation the value from the single pixel can be copied to the corresponding location in 2×2 region in the next layer [215]. The values in rest of the three positions are computed in the subsequent convolutional layers. If the value was copied to random location without the knowledge of the pooling indices, there would be inconsistencies in classification especially in the boundary regions.

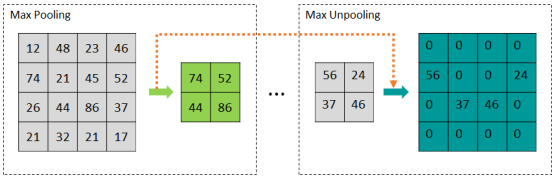

Figure 14: Forwarding pooling indices to maintain spatial relationship during unpooling.

SegNet The SegNet algorithm [9] was launched in 2015 to compete with the FCN network on complex indoor and outdoor images. The architecture was composed of 5 encoding blocks and and 5 decoding blocks. The encoding blocks followed the architecture of the feature extractor in VGG-16 network. Each block is a sequence of multiple convolution, batch normalization and ReLU layers. Each encoding block ends with a max-pooling layer where the indices are stored. Each decoding block begins with a unpooling layer where the saved pooling indices are used (Refer fig.15). The indices from the max-pooling layer of the ith block in the encoder is forwarded to the max-unpooling layer in the $(L-i+1)th$ block in the decoder where L is the total number of blocks in each of the encoder and decoder. The SegNet architecture scored an mIoU of 60.10 as compared to 53.88 by DeepLab-LargeFOV[31] or 49.83 by FCN[130] or 59.77 by Deconvnet[156] on the CamVid Dataset.

4.3 Adversarial Models

Until now, we have seen purely discriminative models like FCN, DeepMask, DeepLab that primarily generates a probability distribution for every pixel across the number of classes. Furthermore, autoencoder treated segmentation as a generative process however the last layer is generally connected to a pixel-wise soft-max classifier. The adversarial learning framework approaches the optimization problem from a different perspective. Generative Adversarial Networks (GANs) gained a lot of popularity due to there remarkable performance as a generative network. The adversarial learning framework mainly consists of

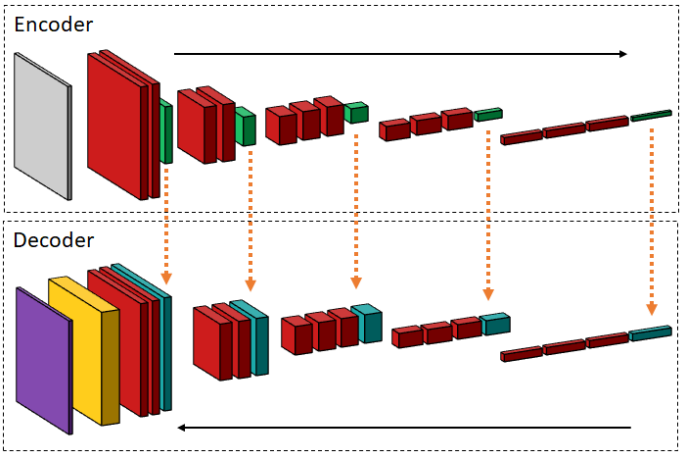

Figure 15: Architecture of SegNet

two networks a generative network and a discriminator network. The generator G tries to generate images,,./ like the ones from the training dataset using a noisy input prior distribution called $p_z(z)$. The network $G(z;\theta_g)$ represents a differentiable function represented by a neural network with weights θ_g. A discriminator network tries to correctly guess whether an input data is from the training data distribution ($p_{data}(x)$) or generated by the generator G. The goal of the discriminator is to get better at catching a fake image, while the generator tries to get better at fooling the discriminator, thus in the process generating better outputs. The entire optimization process can be written as a min-max problem as follows:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \quad (3)$$

The segmentation problem has also been approached from a adversarial learning perspective. The segmentation network is treated as a generator that generates the segmenation masks for each class, whereas a discriminator network tries to predict whether a set of masks is from the ground truth or from the output of the generator [133]. A schematic diagram of the process is shown in fig.20. Furthermore, conditional GANs have been used to perform image to image translation[86]. This framework can be used for image segmentation problems where the semantic boundaries of the image and output segmentation map do not necessarily coincide, for example, in case of creating a schematic diagram of a façade of a building.

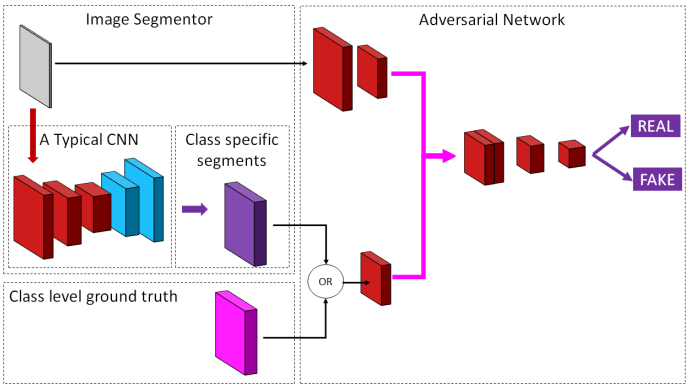

Figure 16: Adversarial learning model for image segmentation

4.4 Sequential Models

Till now almost all the techniques discussed deal with semantic image segmentation. Another class of segmentation problem, namely, instance level segmentation needs slightly different approach. Unlike semantic image segmentation, here all instances of the same object are segmented into different classes. This type of segmentation problem is mostly handled as a learning to give a sequence of object segments as outputs. Hence sequential models come into play in such problems. Some of the main architectures commonly used are convolutional LSTMs, Reccurent Networks, Attention-based models and so on.

4.4.1 Recurrent Models

Traditional LSTM networks employ fully connected weights to model long and short term memories accross sequential inputs. But they fail to capture spatial information of images. Moreover, fully connected weights for images increases the cost of computation by a great extent. In convolutional LSTM [176] these weights are replaced by convolutional layers (Refer fig. **??**). Convolutional LSTMs have been used in several works to perform instance level segmentation. Normally they are used as a suffix to a object segmentation network. The purpose of the recurrent model like LSTM is to select each instance of the object in different timestamps of the sequential output. The approach has been implemented with object segmentation frameworks like FCN and U-NET [28].

4.4.2 Attention Models

While convolutional LSTMs can select different instance of objects at different timestamps, attention models are designed to have more control over this process of localizing individual instances. One simple method to control attention is by spatial inhibition [176]. Spatial inhibition network is designed to learn a bias parameter that cuts off previously detected segments from future activa-

tions. Attention models have been further developed with the introduction of dedicated attention module and an external memory to keep track of segments. In the works of [174], the instance segmentation network was divided into 4 modules. First, and external memory provides object boundary details from all previous steps. Second, a box network attempts to predict the location of the next instance of the object and outputs a sub-region of the image for the third module that is the segmentation module. The segmentation module is similar to a convolutional auto-encoder model discussed previously. The fourth module scores the predicted segments based on whether they qualify as a proper instance of the object. The network terminates when the score goes below a user-defined threshold.

4.5 Weakly Supervised or Unsupervised Models

Neural Networks in general are trained with algorithms like back-propagation, where the parameters **w** are updated based on their local partial derivative with respect to a error value E obtained using a loss function f.

$$w = w + \Delta w = w - \eta \frac{\delta E}{\delta w} \tag{4}$$

The loss function is generally expressed in terms of a distance between a target value and the predicted value. But in many scenarios image segmentation requires the use of data without annotations with ground truth. This leads to the development of unsupervised image segmentation techniques. One of the straight forward ways to achieve this is to use networks pre-trained on other larger datasets with similar kinds of samples and ground truths and use clustering algorithms like K-means on the feature maps. However this kind of semi-supervised technique is inefficient for data samples that have a unique distribution of sample space. Another cons is that the network is trained to perform on a input distribution which is still different from the test data. That does not allow the network to perform to it with full potential. The key problem in fully unsupervised segmentation algorithm is the development of a loss function capable of measuring the quality of segments or cluster of pixels. With all these limitations the amount of literature is comparatively much lighter when it comes to weakly supervised or unsupervised approaches.

4.5.1 Weakly supervised algorithms

Even in the lack of proper pixel level annotations, segmentation algorithms can exploit coarser annotations like bounding boxes or even image level labels[161, 116] for performing pixel level segmentation.

Exploiting bounding boxes From the angle of data annotation, defining bounding boxes is a much less expensive task as compared to pixel level segmentation. The availability of datasets with bounding boxes is also much larger than those with pixel level segmentations. The bounding box can be used as

a weak supervision to generate pixel level segmentation maps. In the works of [42], titled BoxSup, segmentation proposals were generated using region proposal methods like selective search. After that multi-scale combinatorial grouping is used to combine candidate masks and the objective is to select the optimal combination that has the highest IOU with the box. This segmentation map is used to tune a traditional image segmentation network like FCN. BoxSup was able to attain an mIOU of 75.1 in the pascal VOC 2012 test set as compared to 62.2 of FCN or 66.4 of DeepLab-CRF.

4.5.2 Unsupervised Segmentation

Unlike supervised or weakly supervised segmentation, the success of unsupervised image segmentation algorithms are mostly dependent on the learning mechanism. Some of the common approaches are described below.

Learning multiple of objectives One of the most generic approach in unsupervised learning is considering multiple objectives designed in a way to perform well for segmentation when ground-truths are not available.

A common variant called JULE or joint unsupervised learning of deep representation have been used in several applications where there is a lack of samples with ground-truth. The basis of JULE lies in training a sequential model along with a deep feature extraction model. The learning methodology was primarily introduced as an image clustering algorithm. However it has been extended to other applications like image segmentation. The key objective for being able to perform these kinds of segmentation is the development of a proper objective function. In JULE, the objective function considers the affinity between samples in the clusters and also the negative affinity between the clusters and its neighbors. Agglomerative clustering is performed across the timestamps of a recurrent network. In the works of [149], JULE was attempted on image patches rather than entire samples of images. With a similar objective function, it was able to classify the patches into a predefined number of classes. JULE was used in this case to provide the agglomerative clustering information as supervisory signals for the next iteration.

Another variant of learning multiple objectives has been demonstrated through adversarial collaboration[172] among neural networks with independent jobs like monocular depth prediction, estimating camera motion, detecting optical flow and segmentation of a video into the static scene and moving regions. Through competitive collaboration, each network competes to explain the same pixel that either belongs to static or moving class which in turn shares their learned concepts with a moderator to perform the motion segmentation.

Using refinement modules for self supervision : Using other unsupervised over-segmentation techniques can be used to provide supervision to deep feature extractors [92]. By enforcing multiple constraints like similarity between features, spatial continuity, intra-axis normalization. All these objectives are optimized through back propagation. The spatial continuity is achieved by

extracting superpixel from the image using standard algorithms like the SLIC[1] and all pixels within a superpixel are forced to have the same label. The difference between the two segmentation map is used as a supervisory signal to update the weights.

Other relevant unsupervised techniques for extracting semantic information: Learning without annotations is always challenging and hence have a variety of literature where many interesting solutions have been proposed. Using CNNs to solve jigsaw puzzles [157] derived from images can be used to learn semantic connections between various parts of the objects. The proposed context free network takes a set of image tiles as input and tries to establish the correct spatial relations between them. During the process it simultaneously learns features specific to parts of an object as well as their semantic relationships. These patch based self supervision techniques can be further improved using contextual information [152]. Using context-encoders [164] can also derive spatial and semantic relationship among various parts of an image. Context encoders are basically, CNNs trained to generate arbitrary regions of an image which is conditioned by its surrounding information. Another demonstration of extracting semantic information can be found in the process of image colorization [218]. The process of colorization requires pixel level understanding of the semantic boundaries corresponding to objects. Other self supervision techniques can leverage motion cues in videos to segment various parts of an object for better semantic understanding [216]. This method can learn several structural and coherent features for tasks like semantic segmentation, human parsing, instance segmentation and so on.

4.5.3 W-Net

W-Net [207] derived its inspiration from the previously discussed U-Net. The W-Net architecture consists of a two cascaded U-Nets. The first U-Net acts as a encoder that converts an image to its segmented version while the second U-Net tries to reconstruct the original image from the output of the first U-Net that is the segmented image. Two loss functions are minimized simultaneously. One of them being the Mean square error between the input image and the reconstructed image that is given by the second U-Net. The second loss function is derived from the Normalized-Cut [186]. The hard normalized cut is formulated as,

$$Ncut_K(V) = \sum_{k=1}^{K} \frac{\sum_{u \in A_k, v \in V - A_k} w(u,v)}{\sum_{u \in A_k, t \in V} w(u,t)} \tag{5}$$

where, A_k is set of pixels in the $k-th$ segment, V is the set of all the pixels, and w measures the weight between two pixels.

However, this function is not differentiable and hence backpropagation is not

possible. Hence a soft version of function was proposed.

$$J_{soft-Ncut}(V,K) = K - \sum_{k=1}^{K} \frac{\sum_{u\in V} p(u=A_k) \sum_{v\in V} w(u,v) p(v=A_k)}{\sum_{u\in V} p(u=A_k) \sum_{t\in V} w(u,t)} \quad (6)$$

where, $p(u = A_k)$ represents the probability of a node u belonging to a class A_k. The output segmentation maps were further refined using fully connected conditional random fields. Remaining insignificant segments was further merged using hierarchical clustering.

4.6 Interactive Segmentation

Image segmentation is one of the most difficult challenges in the field of computer vision. In many scenarios where the images are too complex, noisy or subjected to poor illumination conditions, a little bit of interaction and guidance from users can significantly improve the performance of segmentation algorithms. Interactive segmentation have been flourishing even outside the deep learning paradigm. However with powerful feature extraction of convolutional neural nets the amount of interaction can be reduced to get extent.

4.6.1 Two stream fusion

One of the most straight forward implementations of interactive segmentation is to have two parallel branches, one from the image another from the an image representing the interactive stream, and fuse them to perform the segmentation[83]. The interaction inputs are taken in form of different coloured dots representing positive and negative classes. With a bit of post processing where intensities of the interaction maps are calculated based on the euclidean distance from the points we get two sets of maps(one for each class) that looks like fuzzy voronoi cells with the points at the centre. The maps are multiplied element-wise to obtain the image for the interaction stream. The two branches are composed of sequence of convolutions and pooling layers. The Hadamard product of features obtained at the end of each branches are sent to the fusion network where a low resolution segmentation map is generated. An alternative approach follows the footsteps of FCN to fuse features of multiple scales to obtain a sharper resolution image.

4.6.2 Deep Extreme Cut

Contrary to the two stream approach, deep extreme cut[138] takes a single pipeline to create segmentation maps from RGB Images. This method expects 4 points from the user denoting the four extreme regions in the boundary of the object (leftmost, rightmost, topmost,bottommost). By creating heatmaps from the points, a 4 channel input is fed into a DenseNet101 network. The final feature map of the network is passed into a pyramid scene parsing module for analyzing global contexts to perform the final segmentation. This method was able to attain an mIOU of 80.3 on the PASCAL test set.

4.6.3 Polygon-RNN

Polygon-RNN[26] takes a different approach to the other methods. Multi scale features are extracted from different layers of a typical VGG Network and concatenated to create a feature block for a recurrent network. The RNN in turn is supposed to provide a sequence of points as an output that represents the contour of the object. The system is primarily designed as an interactive image annotation tool. The users can interact in two different ways. Firstly the users must provide a tight bounding box for the object of interest. Secondly after the polygon is built the users were allowed to edit any point in the polygon. However this editing is not used for any further training of the system and hence presents a small avenue for improvement of the system.

4.7 Building more efficient networks

While many complicated networks with lots of fancy modules can give a very decent quality of semantic segmentation, embedding such algorithms in real-life systems is a different story. Many other factors like cost of hardware, real-time response and so on poses a new degree of challenge. Efficiency is also key for creating consumer level systems.

4.7.1 ENet

The ENet[163] brought forward a couple of interesting design choices to create a quite shallow segmentation network with a small number of parameters (0.37 Million). Instead of a symmetric encoder decoder architecture like SegNet or U-Net, it has a deeper encoder and a shallower decoder. Instead of increasing channel sizes after pooling, parallel pooling operations were performed along with convolutions of stride 2 to reduce overall features. To increase the learning capability PReLU were used as compared to ReLU so that the transfer functions remains dynamic so that it can simulate the jobs of a ReLU as well as an identity function as required. This is normally an important factor in ResNet however because the network is shallow, using PReLU is a smarter choice. Above that using factorized filters also allowed for a smaller number of parameters.

4.7.2 Deep Layer Cascade

Deep Layer Cascade [116] tackles several challenges and makes two significant contributions. Firstly, it analyzed the level of difficulty of pixel level segmentation for various classes. With a cascaded network, easier segments are discovered in the earlier stage while the latter layers focus on regions that need more delicate segments. Secondly, the proposed layer cascading can be used with common networks like Inception-ResNet-V2(IRNet) to improve the speed and even the performance to some extent. The basic principle of IRNet is to create a multi-stage pipeline where in each stage a certain amount of pixels would be classified into one of the segments. In the earlier stages the easiest pixels will be classified and the harder pixels with more uncertainty will move forward to latter stages.

In the consequent stages the convolutions will only take place on those pixels which could not be classified in the previous stage while forwarding yet harder pixels to the next stage. Typically the proposed model comes with three stages each adding more convolutional modules to the network. With layer cascading an mIOU of 82.7 was reached on the VOC12 test set with DeepLabV2 and Deep Parsing Network being the nearest competitors with mIOU of 79.7 and 77.5 respectively. In terms of speed, 23.6 frames were processed per second as compared to 14.6 fps by SegNet or 7.1 fps by DeepLab-V2.

4.7.3 SegFast

Another recent implementation titled SegFast [159] was able to build a network with only 0.6 Million parameters that resulted in a network that can do a forward pass in around 0.38 seconds without a GPU. The approach combined the concept of depth-wise separable convolutions with the fire modules of SqueezeNet. SqueezeNet introduced the concepts of fire modules to reduce the number of convolutional weights. With depth-wise separable convolutions, the number of parameters went further down. They also proposed the use of depthwise separable transposed convolutions for decoding. Even with so many approaches of feature reductions the performance was quite comparable to other popular networks like SegNet.

4.7.4 Segmentation using superpixels

Over-segmentation algorithms [1] have flourished well to divide images into small patches based on local information. With patch classification algorithms these superpixels can be converted to semantic segments. However since the process of over-segmentation do not consider neighborhood relations, it is necessary to include that in the patch classification algorithm. It is much faster to perform patch classification as compared to pixel level classification simply because of the fact that number of superpixels is much lesser than the number of pixels in an image. One of the first works of using CNNs for superpixel level classification was carried out by Farabet et al. [55]. However just considering superpixels without context can result in errorneous classification. In the works of [46], multiple levels of contexts were captured by considering neighborhood super-pixels of different levels during the patch classification. By fusing patch level probabilities from different levels of contexts using methods like weighted averaging, max-voting or uncertainty based approaches like dempster-shafer theory, a very efficient segmentation algorithm was proposed. The works of Das et al. was able to obtain a pixel level accuracy of 77.14% [46] with respect to 74.56% Farabet et al. [55]. While superpixels can be exploited to build efficient semantic segmentation models, the reverse is also true. Conversely, semantic segmentation ground-truths can be used to train networks to perform over-segmentation [197]. Pixel affinities can be calculated using convolutional features to perform over-segmentation while paying special attention to semantic boundaries.

5 Applications

Image segmentation is one of the most commonly addressed problems in the domain of computer vision. It is often augmented with other related tasks like object detection, object recognition, scene parsing, image description generation. Hence this branch of study finds extensive use in various real-life scenarios.

5.1 Content-based image retrieval (CBIR)

With the ever increasing amount of structured and unstructured data on the internet, development of efficient information retrieval systems is of the utmost importance. CBIR systems have hence been a lucrative area of research. Many other related problems like visual question answering, interactive query based image processing, description generation. Image segmentation is useful in many cases as they are representative of spatial relations among various objects [12, 127]. Instance level segmentation is essential for handling numeric queries [217]. Unsupervised approaches [90] are particularly useful for handling bulk amount of non-annotated data which is very common in this field of work.

5.2 Medical imaging

Another major application area for image segmentation is in the domain of health care. Many kinds of diagnostic procedures involve working with images corresponding to different types of imaging source and various parts of the body. Some of the most common types of tasks are segmentation of organic elements like, vessels [58], tissues [91], nerves [132], and so on. Other kinds of problems include localization of abnormalities like tumors [224, 145], aneurysms [48, 131] and so on. Microscopic images [85] also need various kinds of segmentations like cell or nuclei detection, counting number of cells, cell structure analysis for cancer detection and so on. The primary challenges with this domain is the lack of bulk amount of data for challenging diseases, variety in the quality of images due to the different types of imaging device involved. Medical procedures are not only involved to human beings, but also other animals as well as plants.

5.3 Object Detection

With the success of deep learning algorithms there has also been a surge in research areas related to automatic object detection. Many application like robotic maneuverability [114], autonomous driving [196], intelligent motion detection [192], tracking systems [204] and so on. Extremely remote regions such as deep sea [190, 106], or space [181] can be efficiently explored with the help of intelligent robots making autonomous decisions. In sectors like defense, unmanned aerial vehicles or UAVs [154] are used to detect anomalies or threats in remote regions [119]. Segmentation algorithms have significant usage in satellite images for various geo-statistical analysis [109]. In fields like image or video

post-production it is often essential to perform segmentation for various tasks like image matting[115, 115], compositing[24] and rotoscoping[2].

5.4 Forensics

Biometric verification systems like iris [125, 65], fingerprint [94], finger vein [170], dental records [93], involve segmentation of various informative regions for efficient analysis.

5.5 Surveillance

Surveillance systems [147, 95, 89] are associated with various issues like occlusion, lighting or weather conditions. Moreover surveillance system can also involve analysis of images from hyper-spectral sources [4]. Surveillance system can also be extended to various applications such as object tracking [82], searching [3], anomaly detection [173], threat detection [137], traffic control [208] and so on. Image segmentation plays a vital role to segregate objects of interest from the clutter present in natural scenes.

6 Discussion and Future Scope

Throughout the paper various methods have been discussed with an effort to highlight their key contributions, pros and cons. With so many different options it is still hard to choose the right approach for a problem. The most optimal way to choose a correct algorithm is to first analyze the variables that affect the choice.

One of the most important aspect that affects the performance of deep learning based approaches is the availability of datasets and annotations. In that regard a concise list of datasets belonging to various domains has been provided in table 1. When working on other small scale datasets it is a common practice to pre-train the network on a larger dataset of a similar domain. Sometimes may be ample amount of samples are available yet pixel level segmentation labels may not be available as creating them is a taxing problem. Even in those cases pre-training parts of networks on other related problems like classification or localization can also help in the process of learning a better set of weights.

A related decision that one must take in this regard is to choose among supervised, unsupervised or weakly supervised algorithms. In the current scenario there exists a large number of supervised approaches, however unsupervised and weakly supervised algorithms are still far from reaching a level of saturation. This is a legitimate concern in the field of image segmentation because data collection can be carried out through many automated processes but annotating them perfectly requires manual labor. It is one of the most prominent areas where a researcher can contribute in terms of building end-to-end scalable systems that can model data distribution, decide on the optimal number of classes and create accurate pixel-level segmentation maps in a completely unsupervised

domain. Weakly supervised algorithms is also a highly demanding area. It is much easier to collect annotations corresponding to problems like classification or localization. Using those annotations to guide image segmentation problem is also a promising domain.

The next important aspect of building deep learning models for image segmentation is the selection of the appropriate approaches. Pre-trained classifiers can be used for various fully convolutional approaches. Most of the time some kind of multi-scale feature fusion can be carried out by combining information from different depths of the network. Pre-trained classifiers like VGGNet or ResNet or DenseNet are also often used for the encoder part of a encoder-decoder architecture. Here also information can be passed from various layers of encoders to corresponding similar sized layers of the decoder to obtain multi-scale information. Another major benefit of encoder-decoder architectures are that if the down-sampling and up-sampling operations are designed carefully, outputs can be generated which are of the same size as that of the input. It is a major benefit over simple convolutional approaches like FCN or DeepMask. This removes the dependency on the input size and hence makes the system more scalable. These two approaches are the most common in case of semantic segmentation problem. However, if finer level of instance specific segments are required it is often necessary to couple with other methods corresponding to object detection. Utilizing bounding box information is one way to address these problems, while other approaches use attention based models or recurrent models to provide output as sequence of segments for each instance of the object.

There can be two aspects to consider while measuring the performance of the system. One is speed and the other is accuracy. Conditional random field is one of the most commonly used post-processing module for refining outputs from other networks. CRFs can be simulated as an RNN to create end-to-end trainable modules to provide very precise segmentation maps. Other refinement strategies include the use of over-segmentation algorithms like superpixels, or using human interactions to guide segmentation algorithms. In terms of gain in speed, networks can be highly compressed using strategies like depth-wise separable convolutions, kernel factorizations, reducing number of spatial convolutions and so on. These tactics can reduce number of parameters to a great extent without reducing the performance too much. Lately, generative adversarial networks have seen a tremendous rise in popularity. However, their use in the field of segmentation is still pretty thin with only a handful of approaches addressing the avenue. Given the success they have gained it certainly has potential to improve existing systems by a great margin.

The future of image segmentation largely depends on the quality and quantity of available data. While there is an abundance of unstructured data in the internet, the lack of accurate annotations is a legitimate concern. Especially pixel level annotations can be incredibly difficult to obtain without manual intervention. The most ideal scenario would be to exploit the data distribution itself to analyze and extract meaningful segments that represent concepts rather than content. This is an incredibly challenging task especially if we are working

with a huge amount of unstructured data. The key is to map a representation of the data distribution to the intent of the problem statement such that the derived segments are meaningful in some way and contributes to the overall purpose of the system.

7 Conclusion

Image segmentation has seen a new rush of deep learning based algorithms. Starting with the evolution of deep learning based algorithms we have thoroughly explained the pros and cons of the various state of the art algorithms associated with image segmentation based on deep learning. The simple explanations allow the reader to grasp the most basic concepts that contribute to the success of deep learning based image segmentation algorithms. The unified representation scheme followed in the diagrams can highlight the similarities and differences of various algorithms. In the future this theoretical survey work can be accompanied by empirical analysis of the discussed methods.

Acknowledgement

This work is partially supported by the project order no. SB/S3/EECE/054/2016, dated 25/11/2016, sponsored by SERB (Government of India) and carried out at the Centre for Microprocessor Application for Training Education and Research, CSE Department, Jadavpur University. The authors would also like to thank the reviewers for their valuable suggestions which gelp to improve the quality of the manuscript.

Supplementary Information

Image Segmenation before deep learning : A refresher

Thresholding : The most straight forward image segmentation can be assigning class labels to each image based on a threshold-point with respect to the intensity values. One of the earliest algorithm popularly known as Otsu's [199] method chooses a threshold point with respect to maximum variance. Many modern approaches have been applied which involving fuzzy logic [39] or non-linear thresholds [193]. While early approaches were mainly focused on binary thresholding, multi-class segmentation have also come up in subsequent years.

Clustering methods : Clustering methods like K-means [194] can cluster images into more than one class. It is quite a simple process which can yield excellent results for images where objects of interests are in a high contrast with respect to the background. Other clustering approaches are combined with fuzzy logic [16, 150] or even multi-objective optimizations [10].

Histogram-based methods : Histogram-based methods [195, 193] provide a more global perspective when it comes to semantic segmentation. By analyzing the peaks and troughs of the histogram the image can be appropriately segmented into an optimal number of segments. Unlike clustering algorithms like k-means number of clusters need not be known beforehand.

Edge detection : Another angle to look at the problem of image segmentation is to consider the semantic boundaries [184] between objects. Semantic image segmentation is quite related to edge detection algorithms for many reasons such as individual objects tend to be separated in an image by an edge that exhibits a sharp change in the intensity gradient. A common method that utilizes concepts of edge-detection and semantic segmentations are super-pixel based processes [45, 38].

Region-growing methods : While intensity based methods are quite potent in clustering images, they do not consider the factor of locality. Region growing methods rely on the assumption that neighboring pixels within common segment share some common properties. These kind of methods generally start from seed points and slowly grow while staying within semantic boundaries [59] The regions are grown by merging adjacent smaller regions based on some intra-region variance or energy. Many common algorithms such as Mumford-Shah [151] or Snakes algorithm [96] Other variants of such methods depend on the lambda connectedness and grown on the basis of pixel intensities

Graph based approaches : Graph partitioning algorithms can be used to consider context of locality by treating pixels or groups of pixels as nodes thus converting an image to a weighted undirected graphs. Graph cutting algorithms

[186, 21, 22, 180] may be efficiently used to obtain the segments. Probabilistic graphical models such as Markov random fields(MRF) [80] can be used to create a pixel labeling scheme based on prior probability distribution. MRF tries to maximize the probability of correctly classifying pixels or regions based on a set of features.Probabilistic graphical models like MRF or other similar graph based approaches [56] can also be seen as energy minimization problems [47]. Simulated annealing [124] can be an apt example in this regard as well. These approaches can choose to partition graphs based on their energy.

Watershed transformations : Watershed algorithms [73] assumes the gradients of an image as a topographic surface. Analogous to the water flow lines in such a surface, the pixels with the highest gradients act as contours for segmentation

Feature based techniques : Various features such as colors, textures, shape, gradients, and so on can be used to train machine learning algorithms such as neural networks [35, 27, 105] or support vector machines to perform pixel level classification. Before the onset of deep learning fully connected neural networks were efficiently used to perform semantic segmentation. However fully connected networks can incur huge memory challenges for larger images as each layer is accompanied by trainable weight in the order of $O(n^2)$, where n is the height and weight of the final activation map of each input image.

A brief history of neural networks

With the initial proposition of artificial neurons by McCulloch and Pitts [142] and the follow up model of perceptron [179], one could simulate linear models with learnable weights. But the limitation of linear systems soon crept up as they were found unable to learn cases with dependent variables like in case of the XOR problem [162]. It took almost a decade before multi-layer models started to arrive since the introduction of the Neocognitron [63, 62], a hierarchical self-organizing neural architecture. The problem was however to extended the idea of stochastic gradient based learning over multiple layers. That is when the idea of back-propagation [182] surfaced. With back-propagation the error from the visible layers could be propagated as a chain of partial derivatives to the weights in the intermediate layers. Introduction of non-linear activation functions such as sigmoid units allowed these intermediate gradients to flow without break as well as solve the XOR problem. While it was clear that deeper networks provided better learning capability but they also induce vanishing or exploding gradients [14]. This was a difficult problem to deal with specially in sequential networks until long short-term memory cells [81] were proposed to replace tradition recurrent neural networks [143]. Also the convolutional neural networks [110] were introduced as an end-to-end classifier which became one of the most important contribution of the modern computer vision domain. Further down a decade deep learning started flourishing started with Hinton proposing the restricted boltzmann machines [13] and the deep belief networks [79].

Advancement in Hardwares : Another important factor that triggered onset of deep learning was the availability of parallel computation capabilities. Earlier dependence on CPU based architecture was creating a bottleneck for the large amount of floating point operations. Clusters were something that was costly and hence research was quite localized at organizations with substantial funding. But with the onset of graphics processing units(GPUs) [7] developed by Nvidia and their underlying CUDA api for accessing the parallel compute cores neural learning systems got a significant boost. These graphics cards provided hundreds and thousands of computational cores at a much cheaper rate which were efficiently built to handle matrix based operation which is ideal for neural networks.

Larger Datasets : With the availibilty of GPUs neural network based research became much more widespread. But another important factor to boost it was the influx of new and challenging datasets and challenges. Starting with the MNIST dataset [111], digit recognition became one of the vanilla challenges for new systems. In the late 90's Yann LeCun, Yoshua Bengio and Geoffrey Hinton kept deep learning alive at the Canadian institute of advanced research by launching the CIFAR datasets [103] of natural objects for object classification. The challenge was further extended to new heights with the 1000 class Imagenet database [50] along with the competition called Imagenet large scale visual recognition challenge [15]. Till this date this has been one of the main challenges which acts as a benchmark for any new object classification algorithms. As people moved on to more complex datasets, image segmentation became challenging as well. Many datasets and competition such as PASCAL VOC Image segmentation [54], ILSVRC Scene parsing [15], DAVIS Challenge [168], Microsoft COCO [122] and so on came into the picture that boosted research in image segmentation as well.

Broad categories of typical deep learning models

With the advent of deep learning many interesting neural networks were proposed that solved various challenges

Sequential Models : The earliest problem with deep networks were seen with recurrent neural networks [143]. Recurrent networks can be characterized by the feedback loop that allows them to accept a sequence of inputs while carrying the learned information over every time-step. However, over long chains of inputs it was seen that information got lost over time due to vanishing gradients. Vanishing or exploding gradients primarily occurs due to the long multiplication chains of partial derivatives which can push resultant value to almost zero or a huge value, which in turn results in either an insignificant or too large weight update [14]. The first attempt to solve this was proposed by Long short term memory [81] architectures where relevant information can be propagated through long distances through a highway channel which was affected only my

addition or subtraction, hence, preserving the gradient values. Sequential models can have various applications in computer vision such as video processing, instance segmentation and so on. Fig. 17 shows a sample of a generic and unrolled version of a typical recurrent network. .

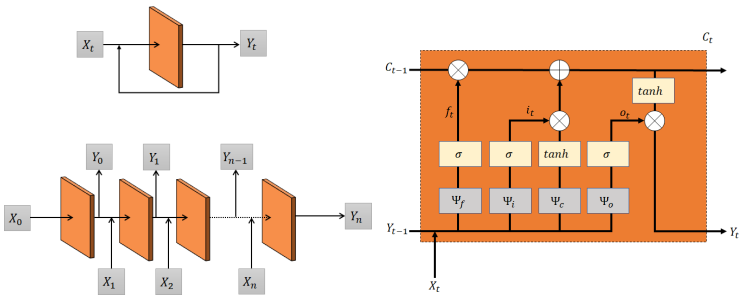

Figure 17: Sequential Models: (topleft) Generic Representation for t-th input, (bottomleft) Unfolded network along a sequence of n inputs, (Right) A generic LSTM Module. The Ψ function represents a linear layer in tradition LSTM and a convolutional layer in convolutional LSTM

Autoencoders : Autoencoders have been around since the introduction of auto-associative networks [102] for multi-layer perceptron. The principle behind autoencoders is to encode raw inputs into a latent representation. A decoder network attempts to reconstruct the input from the encoded representation. The minimization of a loss function based on the difference of the input and the reconstructed output ensures minimum loss of information in the intermediate representation. This hidden representation is a compressed form of the actual input. As it preserves most of the defining properties of the input image it is often used as features for further processing. An autoencoder consists of two main phases, namely, encoding and decoding phase. After training the encoder can be easily used as a feature extractor. The decoder part can be used for generative purposes. Many works use the generative property of the decoder for various image segmentation application [98]. The figure below(18) shows a representation of an autoencoder with fully connected linear layers.

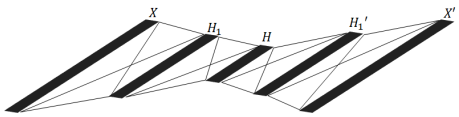

Figure 18: Generic representation of autoencoder with fully connected linear layers

Convolutional Neural Networks : Convolutional neural networks [110, 104] are probably one of the most significant inventions under the wing of deep learning for computer vision. Convolutional kernels have been often used for feature extraction from complex images. However designing kernels was not an easy task especially for complex data like natural images. With convolutional neural networks kernels can be randomly initialized and updated iteratively through back propagation based on a error function like cross entropy or mean square error. Many other operations are commonly found in CNNs such as pooling, batch normalization, activations, residual connections and so on. Pooling layer increase the receptive fields of convolutional kernels. Batch normalization [84] refers to a generalization process that involves normalization of activations across the batch. Activation functions have been an integral part of perceptron based learning. Since the introduction of AlexNet [104], rectified linear units (ReLU) [153] have been the activation function of choice. ReLU(Rectified Linear Unit) provides a gradient of either 0 or 1, thus, preventing vanishing or exploding gradients and also inducing sparsity in the activations. Lately another interesting method for gradient boosting was seen in the application residual connection. Residual connections [78] provided an alternate path for gradients to flow which is devoid of operations that inhibit gradients. Residual connections have also been applied in many cases to improve the quality of segmented images. Fig. 19 shows a convolutional feature extractor along with fully connected classifier.

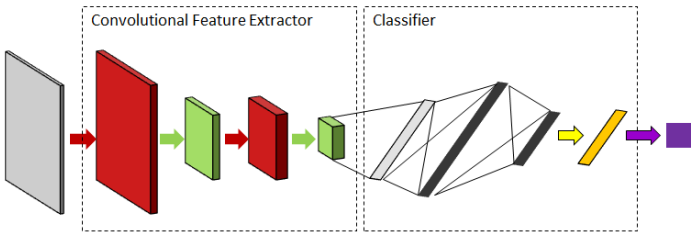

Figure 19: A typical convolutional neural network

Generative Models : Generative models are probably one of the latest attractions of deep learning in computer vision. While sequential models like long short term memory or gated recurrent units are able to generate sequence of vectorized elements, in computer vision it is much more difficult due to the spatial complexities. Lately various methodologies like variational autoencoders [98], or adversarial learning [136, 71] has become extremely efficient in generating complex images. The generative properties can be used quite efficiently in tasks like generation of segmentation masks. An typical example of a generative network is shown in fig. 20 which learns by adversarial learning.

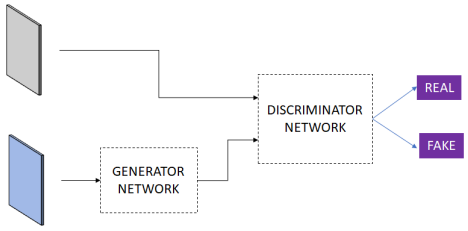

Figure 20: A block diagram of generative adversarial network

References

[1] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., et al. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence 34*, 11 (2012), 2274–2282.

[2] Agarwala, A., Hertzmann, A., Salesin, D. H., and Seitz, S. M. Keyframe-based tracking for rotoscoping and animation. In *ACM Transactions on Graphics (ToG)* (2004), vol. 23, ACM, pp. 584–591.

[3] Ahmad, J., Mehmood, I., and Baik, S. W. Efficient object-based surveillance image search using spatial pooling of convolutional features. *Journal of Visual Communication and Image Representation 45* (2017), 62–76.

[4] Alam, F. I., Zhou, J., Liew, A. W.-C., and Jia, X. Crf learning with cnn features for hyperspectral image segmentation. In *Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International* (2016), IEEE, pp. 6890–6893.

[5] Albiol, A., Torres, L., and Delp, E. J. An unsupervised color image segmentation algorithm for face detection applications. In *Image Processing, 2001. Proceedings. 2001 International Conference on* (2001), vol. 2, IEEE, pp. 681–684.

[6] Araújo, T., Aresta, G., Castro, E., Rouco, J., Aguiar, P., Eloy, C., Polónia, A., and Campilho, A. Classification of breast cancer histology images using convolutional neural networks. *PloS one 12*, 6 (2017), e0177544.

[7] Asano, S., Maruyama, T., and Yamaguchi, Y. Performance comparison of fpga, gpu and cpu in image processing. In *Field programmable logic and applications, 2009. fpl 2009. international conference on* (2009), IEEE, pp. 126–131.

[8] Ather, A. A quality analysis of openstreetmap data. *ME Thesis, University College London, London, UK 22* (2009).

[9] Badrinarayanan, V., Kendall, A., and Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence 39*, 12 (2017), 2481–2495.

[10] Bandyopadhyay, S., Maulik, U., and Mukhopadhyay, A. Multiobjective genetic clustering for pixel classification in remote sensing imagery. *IEEE transactions on Geoscience and Remote Sensing 45*, 5 (2007), 1506–1511.

[11] Barlow, J., Franklin, S., and Martin, Y. High spatial resolution satellite imagery, dem derivatives, and image segmentation for the detection of mass wasting processes. *Photogrammetric Engineering and Remote Sensing 72*, 6 (2006), 687–692.

[12] Belongie, S., Carson, C., Greenspan, H., and Malik, J. Color-and texture-based image segmentation using em and its application to content-based image retrieval. In *Computer Vision, 1998. Sixth International Conference on* (1998), IEEE, pp. 675–682.

[13] Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. Greedy layer-wise training of deep networks. In *Advances in neural information processing systems* (2007), pp. 153–160.

[14] Bengio, Y., Simard, P., and Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks 5*, 2 (1994), 157–166.

[15] Berg, A., Deng, J., and Fei-Fei, L. Large scale visual recognition challenge (ilsvrc), 2010. *URL http://www. image-net. org/challenges/LSVRC 3* (2010).

[16] Bezdek, J. C., Ehrlich, R., and Full, W. Fcm: The fuzzy c-means clustering algorithm. *Computers and Geosciences 10*, 2-3 (1984), 191–203.

[17] Bins, L. S., Fonseca, L. G., Erthal, G. J., and Ii, F. M. Satellite imagery segmentation: a region growing approach. *Simpósio Brasileiro de Sensoriamento Remoto 8*, 1996 (1996), 677–680.

[18] Borji, A. What is a salient object? a dataset and a baseline model for salient object detection. *IEEE Transactions on Image Processing 24*, 2 (2015), 742–756.

[19] Borji, A., Cheng, M.-M., Hou, Q., Jiang, H., and Li, J. Salient object detection: A survey. *arXiv preprint arXiv:1411.5878* (2014).

[20] Borji, A., Cheng, M.-M., Jiang, H., and Li, J. Salient object detection: A benchmark. *IEEE Transactions on Image Processing 24*, 12 (2015), 5706–5722.

[21] Boykov, Y., Veksler, O., and Zabih, R. Fast approximate energy minimization via graph cuts. *IEEE Transactions on pattern analysis and machine intelligence 23*, 11 (2001), 1222–1239.

[22] Boykov, Y. Y., and Jolly, M.-P. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on* (2001), vol. 1, IEEE, pp. 105–112.

[23] Brostow, G. J., Fauqueur, J., and Cipolla, R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters 30*, 2 (2009), 88–97.

[24] Cahill, N. D., and Ray, L. A. Method and system for compositing images to produce a cropped image, Jan. 9 2007. US Patent 7,162,102.

[25] Carass, A., Roy, S., Jog, A., Cuzzocreo, J. L., Magrath, E., Gherman, A., Button, J., Nguyen, J., Bazin, P.-L., Calabresi, P. A., et al. Longitudinal multiple sclerosis lesion segmentation data resource. *Data in brief 12* (2017), 346–350.

[26] Castrejon, L., Kundu, K., Urtasun, R., and Fidler, S. Annotating object instances with a polygon-rnn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 5230–5238.

[27] Chang, P.-L., and Teng, W.-G. Exploiting the self-organizing map for medical image segmentation. In *Computer-Based Medical Systems, 2007. CBMS'07. Twentieth IEEE International Symposium on* (2007), IEEE, pp. 281–288.

[28] Chen, J., Yang, L., Zhang, Y., Alber, M., and Chen, D. Z. Combining fully convolutional and recurrent neural networks for 3d biomedical image segmentation. In *Advances in Neural Information Processing Systems* (2016), pp. 3036–3044.

[29] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062* (2014).

[30] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence 40*, 4 (2018), 834–848.

[31] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence 40*, 4 (2018), 834–848.

[32] Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587* (2017).

[33] Chen, L.-C., Yang, Y., Wang, J., Xu, W., and Yuille, A. L. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 3640–3649.

[34] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv preprint arXiv:1802.02611* (2018).

[35] Cheng, K.-S., Lin, J.-S., and Mao, C.-W. The application of competitive hopfield neural network to medical image segmentation. *IEEE transactions on medical imaging 15*, 4 (1996), 560–567.

[36] Cheng, M.-M., Mitra, N. J., Huang, X., and Hu, S.-M. Salientshape: Group saliency in image collections. *The Visual Computer 30*, 4 (2014), 443–453.

[37] Cheng, M.-M., Mitra, N. J., Huang, X., Torr, P. H., and Hu, S.-M. Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence 37*, 3 (2015), 569–582.

[38] Choudhuri, S., Das, N., Ghosh, S., and Nasipuri, M. A multi-cue information based approach to contour detection by utilizing superpixel segmentation. In *Advances in Computing, Communications and Informatics (ICACCI), 2016 International Conference on* (2016), IEEE, pp. 1057–1063.

[39] Chuang, K.-S., Tzeng, H.-L., Chen, S., Wu, J., and Chen, T.-J. Fuzzy c-means clustering with spatial information for image segmentation. *Computerized MedicalImaging and Graphics 30*, 1 (2006), 9–15.

[40] Comaniciu, D., and Meer, P. Robust analysis of feature spaces: color image segmentation. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on* (1997), IEEE, pp. 750–755.

[41] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 3213–3223.

[42] Dai, J., He, K., and Sun, J. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1635–1643.

[43] Dai, J., He, K., and Sun, J. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 3150–3158.

[44] Dai, J., Li, Y., He, K., and Sun, J. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems* (2016), pp. 379–387.

[45] Das, A., Ghosh, S., Sarkhel, R., Choudhuri, S., Das, N., and Nasipuri, M. Combining multi-level contexts of superpixel using convolutional neural networks to perform natural scene labeling. *arXiv preprint arXiv:1803.05200* (2018).

[46] Das, A., Ghosh, S., Sarkhel, R., Choudhuri, S., Das, N., and Nasipuri, M. Combining multilevel contexts of superpixel using convolutional neural networks to perform natural scene labeling. In *Recent Developments in Machine Learning and Data Analytics*. Springer, 2019, pp. 297–306.

[47] De Albuquerque, M. P., Esquef, I. A., and Mello, A. G. Image thresholding using tsallis entropy. *Pattern Recognition Letters 25*, 9 (2004), 1059–1065.

[48] de Bruijne, M., van Ginneken, B., Viergever, M. A., and Niessen, W. J. Interactive segmentation of abdominal aortic aneurysms in cta images. *Medical Image Analysis 8*, 2 (2004), 127–138.

[49] Demir, I., Koperski, K., Lindenbaum, D., Pang, G., Huang, J., Basu, S., Hughes, F., Tuia, D., and Raskar, R. Deepglobe 2018: A challenge to parse the earth through satellite images. *arXiv preprint arXiv:1805.06561* (2018).

[50] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (2009), IEEE, pp. 248–255.

[51] Du, Y., Arslanturk, E., Zhou, Z., and Belcher, C. Video-based noncooperative iris image segmentation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 41*, 1 (2011), 64–74.

[52] Duan, L., Tsang, I. W., Xu, D., and Chua, T.-S. Domain adaptation from multiple sources via auxiliary classifiers. In *Proceedings of the 26th Annual International Conference on Machine Learning* (2009), ACM, pp. 289–296.

[53] Dumoulin, V., and Visin, F. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285* (2016).

[54] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. The pascal visual object classes (voc) challenge. *International journal of computer vision 88*, 2 (2010), 303–338.

[55] Farabet, C., Couprie, C., Najman, L., and LeCun, Y. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence 35*, 8 (2013), 1915–1929.

[56] Felzenszwalb, P. F., and Huttenlocher, D. P. Efficient graph-based image segmentation. *International journal of computer vision 59*, 2 (2004), 167–181.

[57] for Photogrammetry, I. S., and Sensing, R. Isprs 2d semantic labeling contest.

[58] Fraz, M. M., Remagnino, P., Hoppe, A., Uyyanonvara, B., Rudnicka, A. R., Owen, C. G., and Barman, S. A. Blood vessel segmentation methodologies in retinal images–a survey. *Computer methods and programs in biomedicine 108*, 1 (2012), 407–433.

[59] Freixenet, J., Muñoz, X., Raba, D., Martí, J., and Cufí, X. Yet another survey on image segmentation: Region and boundary information integration. In *European conference on computer vision* (2002), Springer, pp. 408–422.

[60] Friedman, N., and Russell, S. Image segmentation in video sequences: A probabilistic approach. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence* (1997), Morgan Kaufmann Publishers Inc., pp. 175–181.

[61] Fu, K.-S., and Mui, J. A survey on image segmentation. *Pattern recognition 13*, 1 (1981), 3–16.

[62] Fukushima, K. Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural networks 1*, 2 (1988), 119–130.

[63] Fukushima, K., and Miyake, S. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*. Springer, 1982, pp. 267–285.

[64] Galasso, F., Shankar Nagaraja, N., Jimenez Cardenas, T., Brox, T., and Schiele, B. A unified video segmentation benchmark: Annotation, metrics and analysis. In *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 3527–3534.

[65] Gangwar, A., and Joshi, A. Deepirisnet: Deep iris representation with applications in iris recognition and cross-sensor iris recognition. In *Image Processing (ICIP), 2016 IEEE International Conference on* (2016), IEEE, pp. 2301–2305.

[66] Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., and Garcia-Rodriguez, J. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857* (2017).

[67] Geiger, A., Lenz, P., and Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (2012), IEEE, pp. 3354–3361.

[68] Geng, Q., Zhou, Z., and Cao, X. Survey of recent progress in semantic image segmentation with cnns. *Science China Information Sciences 61*, 5 (2018), 051101.

[69] Girshick, R. Fast r-cnn. *arXiv preprint arXiv:1504.08083* (2015).

[70] Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 580–587.

[71] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems* (2014), pp. 2672–2680.

[72] Gould, S., Fulton, R., and Koller, D. Decomposing a scene into geometric and semantically consistent regions. In *Computer Vision, 2009 IEEE 12th International Conference on* (2009), IEEE, pp. 1–8.

[73] Grau, V., Mewes, A., Alcaniz, M., Kikinis, R., and Warfield, S. K. Improved watershed transform for medical image segmentation using prior information. *IEEE transactions on medical imaging 23*, 4 (2004), 447–458.

[74] Han, X. Automatic liver lesion segmentation using a deep convolutional neural network method. *arXiv preprint arXiv:1704.07239* (2017).

[75] Hariharan, B., Arbelaez, P., Bourdev, L., Maji, S., and Malik, J. Semantic contours from inverse detectors. In *International Conference on Computer Vision (ICCV)* (2011).

[76] He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on* (2017), IEEE, pp. 2980–2988.

[77] He, K., Zhang, X., Ren, S., and Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence 37*, 9 (2015), 1904–1916.

[78] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 770–778.

[79] Hinton, G. E., Osindero, S., and Teh, Y.-W. A fast learning algorithm for deep belief nets. *Neural computation 18*, 7 (2006), 1527–1554.

[80] Hochbaum, D. S. An efficient algorithm for image segmentation, markov random fields and related problems. *Journal of the ACM (JACM) 48*, 4 (2001), 686–701.

[81] Hochreiter, S., and Schmidhuber, J. Long short-term memory. *Neural computation 9*, 8 (1997), 1735–1780.

[82] Hong, S., You, T., Kwak, S., and Han, B. Online tracking by learning discriminative saliency map with convolutional neural network. In *International Conference on Machine Learning* (2015), pp. 597–606.

[83] Hu, Y., Soltoggio, A., Lock, R., and Carter, S. A fully convolutional two-stream fusion network for interactive image segmentation. *Neural Networks 109* (2019), 31–42.

[84] Ioffe, S., and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* (2015).

[85] Irshad, H., Veillard, A., Roux, L., and Racoceanu, D. Methods for nuclei detection, segmentation, and classification in digital histopathology: a reviewcurrent status and future potential. *IEEE reviews in biomedical engineering 7* (2014), 97–114.

[86] Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. Image-to-image translation with conditional adversarial networks. *arXiv preprint* (2017).

[87] Jassim, F. A., and Altaani, F. H. Hybridization of otsu method and median filter for color image segmentation. *arXiv preprint arXiv:1305.1052* (2013).

[88] Jégou, S., Drozdzal, M., Vazquez, D., Romero, A., and Bengio, Y. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on* (2017), IEEE, pp. 1175–1183.

[89] Jin, C.-B., Li, S., Do, T. D., and Kim, H. Real-time human action recognition using cnn over temporal images for static video surveillance cameras. In *Pacific Rim Conference on Multimedia* (2015), Springer, pp. 330–339.

[90] Kam, A., Ng, T., Kingsbury, N., and Fitzgerald, W. Content based image retrieval through object extraction and querying. In *cbaivl* (2000), IEEE, p. 91.

[91] Kamnitsas, K., Ledig, C., Newcombe, V. F., Simpson, J. P., Kane, A. D., Menon, D. K., Rueckert, D., and Glocker, B. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical image analysis 36* (2017), 61–78.

[92] Kanezaki, A. Unsupervised image segmentation by backpropagation. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018), IEEE, pp. 1543–1547.

[93] Kang, J., Li, X., Luan, Q., Liu, J., and Min, L. Dental plaque quantification using cellular neural network-based image segmentation. In *Intelligent computing in signal processing and pattern recognition*. Springer, 2006, pp. 797–802.

[94] Kang, J., and Zhang, W. Fingerprint segmentation using cellular neural network. In *Computational Intelligence and Natural Computing, 2009. CINC'09. International Conference on* (2009), vol. 2, IEEE, pp. 11–14.

[95] Kang, K., and Wang, X. Fully convolutional neural networks for crowd segmentation. *arXiv preprint arXiv:1411.4464* (2014).

[96] Kass, M., Witkin, A., and Terzopoulos, D. Snakes: Active contour models. *International journal of computer vision 1*, 4 (1988), 321–331.

[97] Kingma, D. P., and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[98] Kingma, D. P., and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[99] Kong, T., Yao, A., Chen, Y., and Sun, F. Hypernet: Towards accurate region proposal generation and joint object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 845–853.

[100] Krähenbühl, P., and Koltun, V. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems* (2011), pp. 109–117.

[101] Krähenbühl, P., and Koltun, V. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems* (2011), pp. 109–117.

[102] Kramer, M. A. Autoassociative neural networks. *Computers and chemical engineering 16*, 4 (1992), 313–328.

[103] Krizhevsky, A., Nair, V., and Hinton, G. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html* (2014).

[104] Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105.

[105] Kuntimad, G., and Ranganath, H. S. Perfect image segmentation using pulse coupled neural networks. *IEEE Transactions on Neural networks 10*, 3 (1999), 591–598.

[106] Labao, A. B., and Naval, P. C. Weakly-labelled semantic segmentation of fish objects in underwater videos using a deep residual network. In *Asian Conference on Intelligent Information and Database Systems* (2017), Springer, pp. 255–265.

[107] ladys law Skarbek, W., and Koschan, A. Colour image segmentation a survey. *IEEE Transactions on circuits and systems for Video Technology 14*, 7 (1994).

[108] LaLonde, R., and Bagci, U. Capsules for object segmentation. *arXiv preprint arXiv:1804.04241* (2018).

[109] Längkvist, M., Kiselev, A., Alirezaie, M., and Loutfi, A. Classification and segmentation of satellite orthoimagery using convolutional neural networks. *Remote Sensing 8*, 4 (2016), 329.

[110] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*, 11 (1998), 2278–2324.

[111] LeCun, Y., Cortes, C., and Burges, C. Mnist handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist 2* (2010).

[112] Lee, S., Cho, M. S., Jung, K., and Kim, J. H. Scene text extraction with edge constraint and text collinearity. In *2010 International Conference on Pattern Recognition* (2010), IEEE, pp. 3983–3986.

[113] Leibe, B., Seemann, E., and Schiele, B. Pedestrian detection in crowded scenes. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (2005), vol. 1, IEEE, pp. 878–885.

[114] Levi, D., Garnett, N., Fetaya, E., and Herzlyia, I. Stixelnet: A deep convolutional network for obstacle detection and road segmentation. In *BMVC* (2015), pp. 109–1.

[115] Levin, A., Lischinski, D., and Weiss, Y. A closed-form solution to natural image matting. *IEEE transactions on pattern analysis and machine intelligence 30*, 2 (2008), 228–242.

[116] Li, X., Liu, Z., Luo, P., Change Loy, C., and Tang, X. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 3193–3202.

[117] Li, Y., Hou, X., Koch, C., Rehg, J. M., and Yuille, A. L. The secrets of salient object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 280–287.

[118] Li, Y., Qi, H., Dai, J., Ji, X., and Wei, Y. Fully convolutional instance-aware semantic segmentation. *arXiv preprint arXiv:1611.07709* (2016).

[119] Lie, W.-N. Automatic target segmentation by locally adaptive image thresholding. *IEEE Transactions on Image Processing 4*, 7 (1995), 1036–1041.

[120] Lin, G., Milan, A., Shen, C., and Reid, I. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[121] Lin, M., Chen, Q., and Yan, S. Network in network. *arXiv preprint arXiv:1312.4400* (2013).

[122] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision* (2014), Springer, pp. 740–755.

[123] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Sánchez, C. I. A survey on deep learning in medical image analysis. *Medical image analysis 42* (2017), 60–88.

[124] Liu, J., and Yang, Y.-H. Multiresolution color image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 16*, 7 (1994), 689–700.

[125] Liu, N., Li, H., Zhang, M., Liu, J., Sun, Z., and Tan, T. Accurate iris segmentation in non-cooperative environments using fully convolutional networks. In *Biometrics (ICB), 2016 International Conference on* (2016), IEEE, pp. 1–8.

[126] Liu, S., De Mello, S., Gu, J., Zhong, G., Yang, M.-H., and Kautz, J. Learning affinity via spatial propagation networks. In *Advances in Neural Information Processing Systems* (2017), pp. 1520–1530.

[127] Liu, Y., Zhang, D., Lu, G., and Ma, W.-Y. A survey of content-based image retrieval with high-level semantics. *Pattern recognition 40*, 1 (2007), 262–282.

[128] Liu, Z., Li, X., Luo, P., Loy, C.-C., and Tang, X. Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1377–1385.

[129] Loizou, C. P., Murray, V., Pattichis, M. S., Seimenis, I., Pantziaris, M., and Pattichis, C. S. Multiscale amplitude-modulation frequency-modulation (am–fm) texture analysis of multiple sclerosis in brain mri images. *IEEE Transactions on Information Technology in Biomedicine 15*, 1 (2011), 119–129.

[130] Long, J., Shelhamer, E., and Darrell, T. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 3431–3440.

[131] López-Linares, K., Lete, N., Kabongo, L., Ceresa, M., Maclair, G., García-Familiar, A., Macía, I., and Ballester, M. Á. G. Comparison of regularization techniques for dcnn-based abdominal aortic aneurysm segmentation. In *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on* (2018), IEEE, pp. 864–867.

[132] Lu, P., Barazzetti, L., Chandran, V., Gavaghan, K., Weber, S., Gerber, N., and Reyes, M. Highly accurate facial nerve segmentation refinement from cbct/ct imaging using a super-resolution classification approach. *IEEE transactions on biomedical engineering 65*, 1 (2018), 178–188.

[133] Luc, P., Couprie, C., Chintala, S., and Verbeek, J. Semantic segmentation using adversarial networks. *arXiv preprint arXiv:1611.08408* (2016).

[134] Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. In *IEEE International Symposium on Geoscience and Remote Sensing (IGARSS)* (2017).

[135] Maier, O., Menze, B. H., von der Gablentz, J., Häni, L., Heinrich, M. P., Liebrand, M., Winzeck, S., Basit, A., Bentley, P., Chen, L., et al. Isles 2015-a public evaluation benchmark for ischemic stroke lesion segmentation from multispectral mri. *Medical image analysis 35* (2017), 250–269.

[136] Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., and Frey, B. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644* (2015).

[137] Mandal, R., and Choudhury, N. Automatic video surveillance for theft detection in atm machines: An enhanced approach. In *Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on* (2016), IEEE, pp. 2821–2826.

[138] Maninis, K.-K., Caelles, S., Pont-Tuset, J., and Van Gool, L. Deep extreme cut: From extreme points to object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 616–625.

[139] Maninis, K.-K., Pont-Tuset, J., Arbeláez, P., and Van Gool, L. Convolutional oriented boundaries. In *European Conference on Computer Vision* (2016), Springer, pp. 580–596.

[140] Martin, D., Fowlkes, C., Tal, D., and Malik, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision* (July 2001), vol. 2, pp. 416–423.

[141] Masci, J., Meier, U., Cireşan, D., and Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In *International Conference on Artificial Neural Networks* (2011), Springer, pp. 52–59.

[142] McCulloch, W. S., and Pitts, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics 5*, 4 (1943), 115–133.

[143] Medsker, L., and Jain, L. Recurrent neural networks. *Design and Applications 5* (2001).

[144] Mehtre, B. M., Murthy, N. N., Kapoor, S., and Chatterjee, B. Segmentation of fingerprint images using the directional image. *Pattern recognition 20*, 4 (1987), 429–435.

[145] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging 34*, 10 (2015), 1993–2024.

[146] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging 34*, 10 (2015), 1993–2024.

[147] Merlino, A., Morey, D., and Maybury, M. Broadcast news navigation using story segmentation. In *Proceedings of the fifth ACM international conference on Multimedia* (1997), ACM, pp. 381–391.

[148] Molinari, F., Zeng, G., and Suri, J. S. A state of the art review on intima-media thickness (imt) measurement and wall segmentation techniques for carotid ultrasound. *Computer methods and programs in biomedicine 100*, 3 (2010), 201–221.

[149] Moriya, T., Roth, H. R., Nakamura, S., Oda, H., Nagara, K., Oda, M., and Mori, K. Unsupervised segmentation of 3d medical images based on clustering and deep representation learning. In *Medical Imaging 2018: Biomedical Applications in Molecular, Structural, and Functional Imaging* (2018), vol. 10578, International Society for Optics and Photonics, p. 1057820.

[150] Mukhopadhyay, A., Maulik, U., and Bandyopadhyay, S. Multiobjective genetic algorithm-based fuzzy clustering of categorical attributes. *IEEE transactions on evolutionary computation 13*, 5 (2009), 991–1005.

[151] Mumford, D., and Shah, J. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics 42*, 5 (1989), 577–685.

[152] Mundhenk, T. N., Ho, D., and Chen, B. Y. Improvements to context based self-supervised learning. In *Computer Vision and Pattern Recognition (CVPR)* (2018).

[153] Nair, V., and Hinton, G. E. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (2010), pp. 807–814.

[154] Nassar, A., Amer, K., ElHakim, R., and ElHelw, M. A deep cnn-based framework for enhanced aerial imagery registration with applications to uav geolocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2018), pp. 1513–1523.

[155] Neuhold, G., Ollmann, T., Bulo, S. R., and Kontschieder, P. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy* (2017), pp. 22–29.

[156] Noh, H., Hong, S., and Han, B. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1520–1528.

[157] Noroozi, M., and Favaro, P. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision* (2016), Springer, pp. 69–84.

[158] Onyango, C. M., and Marchant, J. A. Physics-based colour image segmentation for scenes containing vegetation and soil. *Image and vision computing 19*, 8 (2001), 523–538.

[159] Pal, A., Jaiswal, S., Ghosh, S., Das, N., and Nasipuri, M. Segfast : A faster squeezenet based semantic image segmentation technique using depth-wise separable convolutions. In *11th Indian Conference on*

Computer Vision, Graphics and Image Processing (ICVGIP 2018), ACM, p. 7.

[160] Pal, N. R., and Pal, S. K. A review on image segmentation techniques. *Pattern recognition 26*, 9 (1993), 1277–1294.

[161] Papandreou, G., Chen, L.-C., Murphy, K. P., and Yuille, A. L. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 1742–1750.

[162] Papert, S. Linearly unrecognizable patterns. *Mathematical aspects of computer science 19* (1967), 176.

[163] Paszke, A., Chaurasia, A., Kim, S., and Culurciello, E. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147* (2016).

[164] Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., and Efros, A. A. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 2536–2544.

[165] Peng, C., Zhang, X., Yu, G., Luo, G., and Sun, J. Large kernel matters–improve semantic segmentation by global convolutional network. *arXiv preprint arXiv:1703.02719* (2017).

[166] Pinheiro, P. O., Collobert, R., and Dollár, P. Learning to segment object candidates. In *Advances in Neural Information Processing Systems* (2015), pp. 1990–1998.

[167] Pinheiro, P. O., Lin, T.-Y., Collobert, R., and Dollár, P. Learning to refine object segments. In *European Conference on Computer Vision* (2016), Springer, pp. 75–91.

[168] Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., and Van Gool, L. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675* (2017).

[169] Porwal, P., Pachade, S., Kamble, R., Kokare, M., Deshmukh, G., Sahasrabuddhe, V., Meriaudeau, F., Quellec, G., MacGillivray, T., Giancardo, L., and Sidib, D. Diabetic retinopathy: Segmentation and grading challenge workshop. *IEEE International Symposium on Biomedical Imaging(ISBI-2018)* (2018).

[170] Qin, H., and El-Yacoubi, M. A. Deep representation-based feature extraction and recovering for finger-vein verification. *IEEE Transactions on Information Forensics and Security 12*, 8 (2017), 1816–1829.

[171] Radau, P., Lu, Y., Connelly, K., Paul, G., Dick, A., and Wright, G. Evaluation framework for algorithms segmenting short axis cardiac mri. *The MIDAS Journal-Cardiac MR Left Ventricle Segmentation Challenge 49* (2009).

[172] Ranjan, A., Jampani, V., Kim, K., Sun, D., Wulff, J., and Black, M. J. Adversarial collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. *arXiv preprint arXiv:1805.09806* (2018).

[173] Ravanbakhsh, M., Nabi, M., Mousavi, H., Sangineto, E., and Sebe, N. Plug-and-play cnn for crowd motion analysis: An application in abnormal event detection. *arXiv preprint arXiv:1610.00307* (2016).

[174] Ren, M., and Zemel, R. S. End-to-end instance segmentation with recurrent attention. *arXiv preprint arXiv:1605.09410* (2017).

[175] Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (2015), pp. 91–99.

[176] Romera-Paredes, B., and Torr, P. H. S. Recurrent instance segmentation. In *European Conference on Computer Vision* (2016), Springer, pp. 312–329.

[177] Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (2015), Springer, pp. 234–241.

[178] Ros, G., Sellart, L., Materzynska, J., Vazquez, D., and Lopez, A. M. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 3234–3243.

[179] Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review 65*, 6 (1958), 386.

[180] Rother, C., Kolmogorov, V., and Blake, A. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)* (2004), vol. 23, ACM, pp. 309–314.

[181] Rothrock, B., Kennedy, R., Cunningham, C., Papon, J., Heverly, M., and Ono, M. Spoc: Deep learning-based terrain classification for mars rover missions. In *AIAA SPACE 2016* (2016), p. 5539.

[182] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning representations by back-propagating errors. *Nature 323*, 6088 (1986), 533.

[183] Sabour, S., Frosst, N., and Hinton, G. E. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems* (2017), pp. 3856–3866.

[184] Senthilkumaran, N., and Rajesh, R. Edge detection techniques for image segmentation–a survey of soft computing approaches. *International Journal of Recent Trends in Engineering 1*, 2 (2009), 250–254.

[185] Sharma, N., and Aggarwal, L. M. Automated medical image segmentation techniques. *Journal of medical physics/Association of Medical Physicists of India 35*, 1 (2010), 3.

[186] Shi, J., and Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence 22*, 8 (2000), 888–905.

[187] Shi, J., Yan, Q., Xu, L., and Jia, J. Hierarchical image saliency detection on extended cssd. *IEEE transactions on pattern analysis and machine intelligence 38*, 4 (2016), 717–729.

[188] Shotton, J., Winn, J., Rother, C., and Criminisi, A. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *European conference on computer vision* (2006), Springer, pp. 1–15.

[189] Silveira, M., Nascimento, J. C., Marques, J. S., Marçal, A. R., Mendonça, T., Yamauchi, S., Maeda, J., and Rozeira, J. Comparison of segmentation methods for melanoma diagnosis in dermoscopy images. *IEEE Journal of Selected Topics in Signal Processing 3*, 1 (2009), 35–45.

[190] Song, Y., Zhu, Y., Li, G., Feng, C., He, B., and Yan, T. Side scan sonar segmentation using deep convolutional neural network. In *OCEANS–Anchorage, 2017* (2017), IEEE, pp. 1–4.

[191] Staal, J., Abràmoff, M. D., Niemeijer, M., Viergever, M. A., and Van Ginneken, B. Ridge-based vessel segmentation in color images of the retina. *IEEE transactions on medical imaging 23*, 4 (2004), 501–509.

[192] Szirányi, T., László, K., Czúni, L., and Ziliani, F. Object oriented motion-segmentation for video-compression in the cnn-um. *Journal of VLSI signal processing systems for signal, image and video technology 23*, 2-3 (1999), 479–496.

[193] Tan, K. S., and Isa, N. A. M. Color image segmentation using histogram thresholding–fuzzy c-means hybrid approach. *Pattern Recognition 44*, 1 (2011), 1–15.

[194] Tatiraju, S., and Mehta, A. Image segmentation using k-means clustering, em and normalized cuts. *University Of California Irvine* (2008).

[195] Tobias, O. J., and Seara, R. Image segmentation by histogram thresholding using fuzzy sets. *IEEE transactions on Image Processing 11*, 12 (2002), 1457–1465.

[196] Treml, M., Arjona-Medina, J., Unterthiner, T., Durgesh, R., Friedmann, F., Schuberth, P., Mayr, A., Heusel, M., Hofmarcher, M., Widrich, M., et al. Speeding up semantic segmentation for autonomous driving. In *MLITS, NIPS Workshop* (2016).

[197] Tu, W.-C., Liu, M.-Y., Jampani, V., Sun, D., Chien, S.-Y., Yang, M.-H., and Kautz, J. Learning superpixels with segmentation-aware affinity loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 568–576.

[198] Uijlings, J. R., Van De Sande, K. E., Gevers, T., and Smeulders, A. W. Selective search for object recognition. *International journal of computer vision 104*, 2 (2013), 154–171.

[199] Vala, M. H. J., and Baxi, A. A review on otsu image segmentation algorithm. *International Journal of Advanced Research in Computer Engineering and Technology (IJARCET) 2*, 2 (2013), pp–387.

[200] Van de Sande, K. E., Uijlings, J. R., Gevers, T., and Smeulders, A. W. Segmentation as selective search for object recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (2011), IEEE, pp. 1879–1886.

[201] Van Ginneken, B., Stegmann, M. B., and Loog, M. Segmentation of anatomical structures in chest radiographs using supervised methods: a comparative study on a public database. *Medical image analysis 10*, 1 (2006), 19–40.

[202] Varma, G., Subramanian, A., Namboodiri, A., Chandraker, M., and Jawahar, C. Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments. *arXiv preprint arXiv:1811.10200* (2018).

[203] Veit, A., Matera, T., Neumann, L., Matas, J., and Belongie, S. Coco-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140* (2016).

[204] Vilarino, D. L., Cabello, D., and Brea, V. M. An analogic cnn-algorithm of pixel level snakes for tracking and surveillance tasks. In *Cellular Neural Networks and Their Applications, 2002.(CNNA 2002). Proceedings of the 2002 7th IEEE International Workshop on* (2002), IEEE, pp. 84–91.

[205] Wang, K., Babenko, B., and Belongie, S. End-to-end scene text recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (2011), IEEE, pp. 1457–1464.

[206] Wei, G.-Q., Arbter, K., and Hirzinger, G. Real-time visual servoing for laparoscopic surgery. controlling robot motion with color image segmentation. *IEEE Engineering in Medicine and Biology Magazine 16*, 1 (1997), 40–45.

[207] Xia, X., and Kulis, B. W-net: A deep model for fully unsupervised image segmentation. *arXiv preprint arXiv:1711.08506* (2017).

[208] Xu, J., Wang, G., and Sun, F. A novel method for detecting and tracking vehicles in traffic-image sequence. In *Fifth International Conference on Digital Image Processing (ICDIP 2013)* (2013), vol. 8878, International Society for Optics and Photonics, p. 88782P.

[209] Xu, N., Yang, L., Fan, Y., Yue, D., Liang, Y., Yang, J., and Huang, T. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327* (2018).

[210] Yang, C., Zhang, L., Lu, H., Ruan, X., and Yang, M.-H. Saliency detection via graph-based manifold ranking. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* (2013), IEEE, pp. 3166–3173.

[211] Yu, F., and Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122* (2015).

[212] Yu, F., Xian, W., Chen, Y., Liu, F., Liao, M., Madhavan, V., and Darrell, T. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687* (2018).

[213] Yuan, J., Gleason, S. S., and Cheriyadat, A. M. Systematic benchmarking of aerial image segmentation. *IEEE Geoscience and Remote Sensing Letters 10*, 6 (2013), 1527–1531.

[214] Zeiler, M. D., and Fergus, R. Visualizing and understanding convolutional networks. In *European conference on computer vision* (2014), Springer, pp. 818–833.

[215] Zeiler, M. D., and Fergus, R. Visualizing and understanding convolutional networks. In *European conference on computer vision* (2014), Springer, pp. 818–833.

[216] Zhan, X., Pan, X., Liu, Z., Lin, D., and Loy, C. C. Self-supervised learning via conditional motion propagation. *arXiv preprint arXiv:1903.11412* (2019).

[217] Zhang, Q., Goldman, S. A., Yu, W., and Fritts, J. E. Content-based image retrieval using multiple-instance learning. In *ICML* (2002), vol. 2, pp. 682–689.

[218] Zhang, R., Isola, P., and Efros, A. A. Colorful image colorization. In *European conference on computer vision* (2016), Springer, pp. 649–666.

[219] Zhao, B., Feng, J., Wu, X., and Yan, S. A survey on deep learning-based fine-grained object classification and semantic segmentation. *International Journal of Automation and Computing 14*, 2 (2017), 119–135.

[220] Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. Pyramid scene parsing network. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 2881–2890.

[221] Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., and Torr, P. H. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 1529–1537.

[222] Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. Semantic understanding of scenes through the ade20k dataset. *arXiv preprint arXiv:1608.05442* (2016).

[223] Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. Scene parsing through ade20k dataset. In *Proc. CVPR* (2017).

[224] Zikic, D., Ioannou, Y., Brown, M., and Criminisi, A. Segmentation of brain tumor tissues with convolutional neural networks. *Proceedings MICCAI-BRATS* (2014), 36–39.