International Journal of Engineering Sciences Paradigms and Researches (IJESPR)
(Vol. 32, Issue 01) and (Publishing Month: July 2016)
(An Indexed, Referred and Impact Factor Journal)
ISSN: 2319-6564
www.ijesonline.com

Evolution of Image Segmentation Utilizing Deep Convolutional Neural Network: A Survey

Prakash Dehury[1]
Aryan Institute of Engineering & Technology, Bhubaneswar, Odisha

Prakash Kumar Sarangi[2]
NM Institute of Engineering and Technology, Bhubaneswar, Odisha

Bhaktipadarbinda Mohanty[3]
Raajdhani Engineering College, Bhubaneswar, Odisha

Abstract

From the independent vehicle heading to clinical finding, the necessity of the undertaking of picture division is all over the place. Division of a picture is one of the essential errands in PC vision. This errand is similarly muddled than other vision assignments as it needs low-level spatial data. Essentially, picture division can be of two sorts: semantic division and example division. The consolidated adaptation of these two fundamental assignments is known as panoptic division. In the new period, the accomplishment of profound convolutional neural organizations (CNN) has impacted the field of division extraordinarily and gave us different fruitful models to date. In this review, we will take a look at the development of both semantic and occasion division work dependent on CNN. We have additionally determined relative design subtleties of some cutting edge models and talk about their preparation subtleties to introduce a clear comprehension of hyper-boundary tuning of those models. We have likewise drawn a correlation among the presentation of those models on various datasets. In conclusion, we have given a brief look at some best in class panoptic division models.

Keywords: Convolutional Neural Network, Deep Learning, Semantic Segmentation, Instance Segmentation, Panoptic Segmentation, Survey.

1. Introduction

We are living in the era of artificial intelligence (AI) and the advancement of deep learning is fueling AI to spread over rapidly [1], [2]. Among different deep learning models, convolutional neural network(CNN) [3, 4, 5] has shown outstanding performance in different high level computer vision task such as image classification [6, 7, 8, 9, 10, 11, 12, 13, 14, 15], object detection [16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29] etc. Though the advent and success of AlexNet [6] turned the field of computer vision towards CNN from traditional machine learning algorithms. But the concept of CNN was not a new one. It started from the discovery of Hubel and Wiesel [30] which explained that there are simple and complex neurons in the primary visual cortex and the visual processing always starts with simple structures such as oriented edges. Inspired by this idea, David Marr gave us the next insight that vision is hierarchical [31]. Kunihiko Fukushima was deeply inspired by the work of Hubel and Wiesel and built a multi-layered neural network called Neocognitron [32] using simple and complex neurons. It was able to recognize patterns in images and was spatial invariant. In 1989, Yann LeCun turned the theoretical idea of Neocognitron into a practical one called LeNet-5 [33]. LeNet-5 was the first CNN developed for recognizing handwritten digits. LeCun et al. used back propagation [34][11] algorithm to train his CNN. The in-

vention of LeNet-5 paved the way for the continuous success of CNN in various high-level computer vision tasks as well as motivated researchers to explore the capabilities of such networks for pixel-level classification problems like image segmentation. The key advantage of CNN over traditional machine learning methods is the ability to learn appropriate feature representations for the problem at hand in an end-to-end training fashion instead of using hand-crafted features that require domain expertise [35].

Applications of image segmentation are very vast. From the autonomous car driving [36] to medical diagnosis [37, 38], the requirement of the task of image segmentation is everywhere. Therefore, in this article, we have tried to give a survey of different image segmentation models. This survey study has covered recent CNN-based state-of-the-art. Mainly semantic segmentation and instance segmentation of an image are discussed. Herein, we have described comparative architectural details of notable different state-of-the-art image segmentation models. Also, different aspects of those models are presented in tabular form for clear understanding. In addition, we have given a glimpse of recent state-of-the-art panoptic segmentation models.

Contributions of this paper

- Gives taxonomy and survey of the evolution of CNN based image segmentation.
- Explores elaborately some CNN based popular state-of-the-art segmentation models.
- Compares training details of those models to have a clear view of hyper-parameter tuning.
- Compares the performance metrics of those state-of-the-art models on different datasets.

Organization of the Article

Starting from the introduction in section 1, the paper is organized as follows: In section 2, we have given background details of our work. In sections 3 and 4, semantic segmentation and instance segmentation works are discussed respectively with some subsections. In section 5, Panoptic segmentation is presented in brief. The paper is concluded in section 6.

2. Background Details

2.1. Why Convolutional Neural Networks?

The computer vision has various tasks among them image segmentation as mentioned in section 2.2 is the focus of this article. Various researchers are addressing this task in different way using traditional machine learning algorithms like in [39, 40, 41] with the help various technique such as thresholding [42], region growing [43, 44], edge detection [45, 46, 47], clustering [48, 49, 50, 51, 52, 53, 54, 55], super-pixel [56, 57], etc for years. Most of the successful works are based on handcrafted machine learning features such as HOG [58, 59, 60, 61], SIFT [62, 63], etc. First of all, feature engineering needs domain expertise and the success of those machine learning-based models was slowed down around the era when deep learning was started to take over the world of computer vision. To give a outstanding performance, deep learning only needs data and it does not need any traditional handcrafted feature engineering techniques. Also, traditional machine learning algorithm can not adjust itself for a wrong prediction. On the other hand, deep learning has that capability to adapt itself according to the predicted result. Among different deep learning algorithms, CNN got tremendous success in different fields of computer vision as well as grab the area of image segmentation [64, 65].

2.2. Image Segmentation

In computer vision, image segmentation is a way of segregating a digital image into multiple regions according to the different properties of pixels. Unlike classification and object detection, it is typically a low-level or pixel-level vision task as the spatial information of an image is very important for segmenting different regions semantically. Segmentation aims to extract meaningful information for easier analysis. In this case, the image pixels are labeled in such a way that every pixel in an image shares certain characteristics such as color, intensity, texture, etc. [66, 67]. Mainly, image segmentation is of two types: semantic segmentation and instance segmentation. Also, there is another type called panoptic segmentation[68] which is the unified version of two basic segmentation processes. Figure 1 shows different types of segmentation and figure 2 shows the same with examples. In subsequent sections, we have elaborately discussed state-of-the-art of different CNN-based image segmentations.

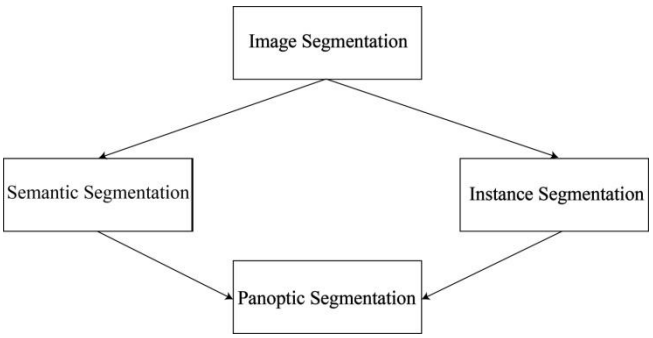

Figure 1: Different types of image segmentation

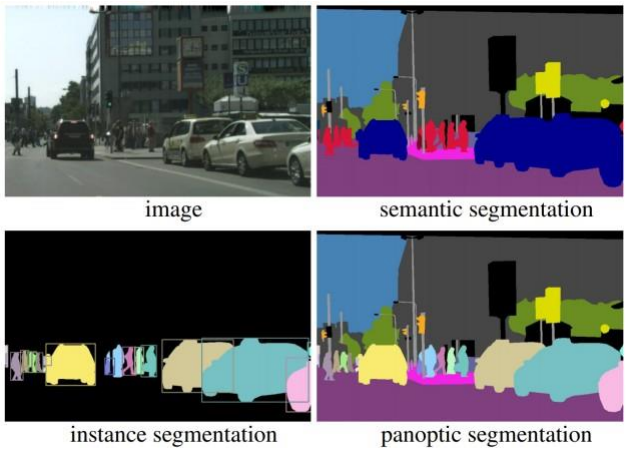

Figure 2: An example of different types of image segmentation. From [69]

.

In addition, CNN is also used successfully for video object segmentation. In a study [70], Caelles et al. have first used a Fully convolutional network for one-shot video object segmentation. In another study [71], the authors have used ResNet [10] based Siamese Encoder with Global Convolutional Block for video object segmentation. On the other hand Miao et al. have used a CNN based semantic segmentation network and proposed Memory Aggregation Network (MA-Net) [72] to handle interactive video object segmentation(iVOS).

The authors of [73] has used a CNN based semantic segmentation network as a base network

for Collaborative Video Object Segmentation by Foreground-Background Integration(CFBI). These are some of CNN- based video segmentation models that got state-of-the-art results on various video segmentation datasets. Due to the scope and size of the article, we have not covered this topic in detail in the present article.

3. Semantic Segmentation

Semantic segmentation describes the process of associating each pixel of an image with a class label [74]. Figure 3 shows the black-box view of semantic segmentation. After the success of AlexNet in 2012, we have got different successful semantic segmentation models based on CNN. In this section, we have present a survey of the evolution of CNN based semantic segmentation models. In addition, we have brought up here an elaborate exploration of some state-of-the-art models.

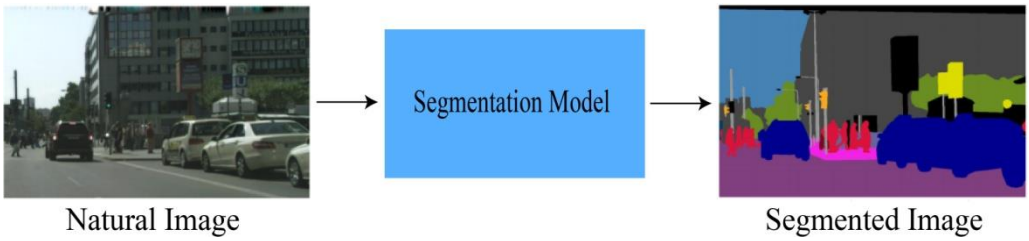

Figure 3: The process of semantic segmentation.

Evolution of CNN based Semantic Segmentation Models

The application of CNN in semantic segmentation models has started with a huge diver-sity. In study [75], the authors have used multi-scale CNN for scene labeling and achieved state-of-the-art results in the Sift flow [76], the Bercelona dataset [77] and the Standford background dataset [78]. R-CNN [79] used selective search [80] algorithm to extract region proposals first and then applied CNN upon each proposal for PASCAL VOC semantic seg- mentation challenge [81]. R-CNN achieved record result over second order pooling (O_2P) [82] which was a leading hand-engineered semantic segmentation system at that time. Atthe same time, Gupta et al. [61] used CNN along with geocentric embedding on RGB-D images for semantic segmentation.

Among different CNN based semantic segmentation models, Fully Convolutional Network(FCN) [83], as discussed in subsection 3.2.1, gained the maximum attention and an FCN based semantic segmentation model trend has emerged. To retain the spatial informa- tion of an image, FCN based models removed fully connected layers of traditional CNN. In studies [84] and [85], the authors have used contextual features and achieved state of the art performance. Recently, in [86], the authors have used fully convolutional two stream fusion network for interactive image segmentation.

Chen et al aggregate 'atrous' algorithm and conditional random (CRF) field in seman- tic segmentation and proposed DeepLab [87] as discussed in subsection 3.2.2. Later the authors have incorporated 'Atrous Special Pyramid Pooling (ASPP)' in DeepLabv2 [88]. DeepLabv3 [89] has gone further and used a cascaded deep ASPP module to incorporate multiple contexts. All three versions of DeepLab have achieved good results.

Deconvnet [90] used convolutional network followed by hierarchically opposite de-convolutional network for semantic segmentation as discussed in section 3.2.3. Ronneberger et al used a U-shaped network called U-Net [91] which has a contracting and an expansive pathway to approach semantic segmentation. Contracting path extracts feature maps and reduces spatial

information as a traditional convolution network. Expansive pathway takes the contracted feature map as input and apply an up-convolution. Section 3.2.3 discussed the model in more detail. Recently, in [92], the authors have used U-Net with *multiRes* block for multimodal biomedical image segmentation and got better result than using classical U-Net. SegNet [93] is a encoder-decoder network for semantic segmentation. The encoder is a basic VGG16 network excluding FC layers. The decoder is identical to encoder but the layers are hierarchically opposite. SegNet is discussed in section 3.2.3. The basic architectural intuition of U-Net, Deconvnet, and SegNet are similar except some individual modifications. The second half of those architectures is the mirror image of the first half.

Liu et al. mixed the essence of global average pooling and L2 normalization layer in FCN [83] architecture, and proposed ParseNet [94] to achieve state of the art result in various datasets. Zhao et al. proposed Pyramid Scene Parsing Network(PSPNet) [95]. They have used Pyramid Pooling Module on top of the last extracted feature map to incorporate global contextual information for better segmentation. Peng et al. used the idea of global convolution using a large kernel to apply the advantage of both local and global features. Pyramid Attention Network (PAN) [96], ParseNet [94], PSPNet[95] and GCN[97] have used global context information with local feature to have better segmentation. Sections 3.2.4 and 3.2.5 will discuss those models in detail.

Fully convolutional DenseNet [10] is used to address semantic segmentation in [98, 99]. DeepUNet [100], a ResNet based FCN, used to segment sea land. At the same time, ENet[101], ICNet[102] are used as real-time semantic segmentation models for the autonomous vehicles. Some recent works [103, 104, 105] have used combination of encoder-decoder architecture and dilated convolution for better segmentation. Transfer learning or domain adaption also uses for semantic segmentation [106]. Kirillov et al.[107] used point-based rendering in DeepLabV3[89] and in semanticFPN to produce state-of-the-art semantic segmentation models.

Researcher from different field of deep learning has also infused CNN to address semantic segmentation. In study[108], the authors have trained CNN along with adversial network. Luo et al. have also used CNN as generator and discriminator in a adversial network and proposed Category level Advisory Network(CLAN)[109]. In [110], The authors have used same configuration as CLAN with information bottleneck for domain adaptive semantic segmentation and proposed Information Bottlenecked Adversarial Network(IBAN) and Significance-aware Information Bottlenecked Adversarial Network (SIBAN). In another study [111], the authors have used CNN based adversial network named Macro-Micro Adversila Network (MMAN) for human parsing.

Some researcher have used CNN models for attention based image segmentation. Wang et al. used Non-local Neural Network in [112]. Huang et al. used DeepLab for feature map extraction and then the feature maps are fed into recurrent criss cross attention module [113] for semantic segmentation. In another study [114], The authors have aggregated long range contextual information in convolutional feature map using global attention network to address scene parsing. In [115], the authors have used dual attention network in combination of CNN for scene segmentation.

Some popular state-of-the-art semantic segmentation models

In this section, we are going to explore architectural details of some state of the art CNN based semantic segmentation models in detail. The models are categorized on the basis of the most impotant feature used. At the end of each categorical discussion, we have also briefly discussed the advantages and weaknesses of a particular model category.

Based on Fully Convolutional Network:

FCN: Long et al. proposed the idea of Fully Convolutional Network(FCN) [83] to address the semantic segmentation task. They have used AlexNet[6], VGGNet[8] and GoogleNet[9](all three pre-trained on ILSVRC [116] data) as base models. They transferred these models from classifiers to dense FCN by substituting fully connected layers with 1×1 convolutional layers and append a 1×1 convolution with channel dimension 21 to predict scores for each of the 20 PASCAL VOC [117] classes and background class.

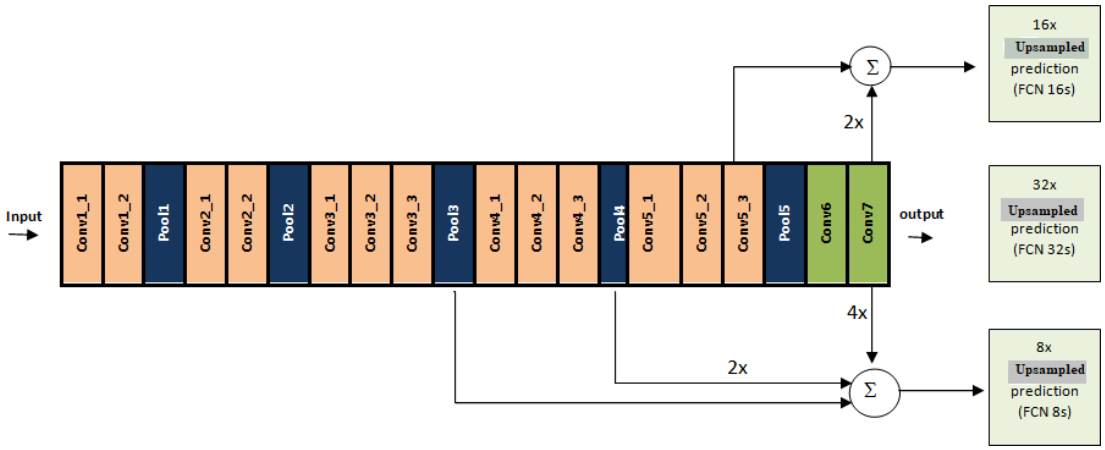

Figure 4: Architecture of FCN32s, FCN16s, FCN8s

This process produces a class presence heat map in low resolution. The authors have experienced that among FCN-AlexNet, FCN-VGG16 and FCN-GoogLeNet, FCN-VGG16 gave the highest accuracy on PASCAL VOC 2011 validation dataset. So, they choose the FCN-VGG16 network for further experiments. As the network produces coarse output locations, the authors used bilinear interpolation to upsample the coarse output 32× to make it pixel dense. But this upsampling was not enough for fine-grained segmentation. So they have used skip connection[118] to combine the final prediction layer and feature-rich lower layers of VGG16 and call this combination as *deep jet*. Figure 4 shows different *deep jet*s : FCN-16s and FCN-8s and FCN-32s. Among them, FCN-8s gave the best result in PASCAL VOC 2011 & 2012 [117] test dataset and FCN-16s gave the best result on both NYUDv2 [119] & SIFT Flow [76] datasets.

Major changes in FCN which helped the model to achieve state of the art result are the base model VGG16, bipolar interpolation technique for up-sampling the final feature map and skip connection for combining low layer and high layer features in the final layer for fine-grained semantic segmentation.

FCN has used only local information for semantic segmentation but only local information makes semantic segmentation quite ambiguous as it looses global semantic context of the image. To reduce ambiguity contextual information from the whole image is much helpful.

Based on Dialtation/Atrous convolution:

Dialatednet: Traditional CNN, used for classification tasks, loses resolution in its way and it is not suitable for dense prediction. Yu and Koltun have introduced a modified version of traditional CNN, called dialated convolution or DialatedNet [120], to accumulate multi-scale contextual information systematically for better segmentation without suffering the loss of resolution. DialatedNet is like a rectangular prism of convolutional layers, unlike

conventional pyramidal CNN. Without losing any spatial information, it can support the exponential expansion of receptive fields as shown in figure 5.

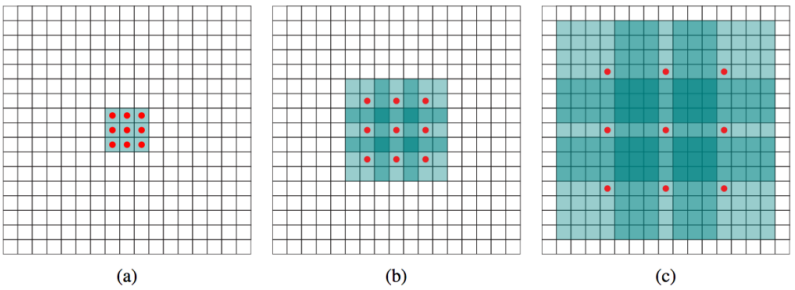

Figure 5: (a) 1-dialatednet with receptive field 3×3, (b) 2-dialatednet with receptive field 7×7 and (c)-dialatednet with receptive field 15×15. From [120]

.

DeepLab: Chen et al. has brought together methods from Deep Convolutional Neural Network(DCNN) and probabilistic graphical model. The authors have faced two technical difficulties in the application of DCNN to semantic segmentation: down sampling and spatial invariance. To handle the first problem, they have employed 'atrous' [121] algorithm for efficient dense computation of CNN. Figure 6a and 6b shows atrous algorithm in 1-D and in 2-D. To handle the second problem, they have applied a fully connected pairwise conditional random field (CRF) to capture fine details. In addition, the authors have reduced the size of the receptive field 6× than the original VGG16 [8] network to reduce the time consumption of the network and also used multi-scale prediction for better boundary localization.

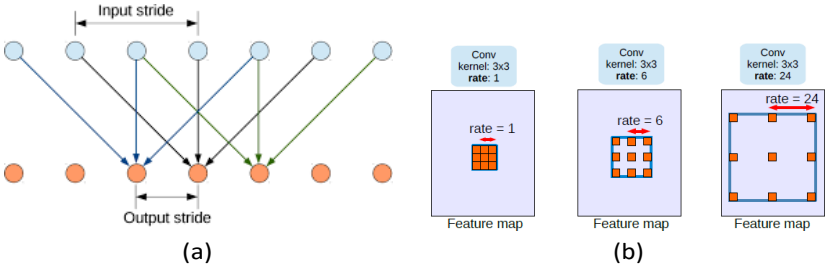

Figure 6: Illustration of atrous algorithm (a) in 1-D, when kernel size=3, input-stride=2 and output-stride=1. From [87] and (b) in 2-D, when kernel size 3×3, with rate 1, 6 and 24. From [89]

Advantage of dilation based model is that it helps to retain spatial resolution of the image to produce dense prediction. But use of dilation convolution isolates image pixels from its global context which makes it prone to misclassification.

Based on Top-down/Bottom-up approach:

Deconvnet: Deconvnet [90], proposed by Noh et al., has a convolutional and de-convolutional network. The convolutional network is topologically identical with the first 13 convolution layers and 2 fully connected layers of VGG16[8] except for the final classification layer. As in VGG16, pooling and rectification layers are also added after some of the convolutional layers. The De-convolutional network is identical to the convolutional network but hierarchically opposite. It also has multiple series of deconvolution, un-pooling and rectification layers. All the layers of convolutional and de-convolutional network extract feature maps except the last layer of the de-convolutional network which generates pixel-wise class

probability maps of the same size as the input image. In the deconvolutional network, the authors have applied unpooling which is the reverse operation of the pooling operation of the convolutional networks to reconstruct the original size of activation. Following [7] and [122], unpooling is done using max-pooling indices which are stored at the time of convolution operation in the convolutional network. To densify enlarged but sparse un-pooled feature maps, convolution like operation is done using multiple learned filters by associating single input activation with multiple outputs. Unlike FCN, the authors applied their network on object proposals extracted from the input image and produced pixel-wise prediction. Then they have aggregated outputs of all proposals to the original image space for segmentation of the whole image. This instance wise segmentation approach handles multi-scale objects with fine detail and also reduces training complexity as well as memory consumption for training. To handle the internal covariate shift in the network, the authors have added batch normalization [123] layer on top of convolutional and de-convolutional layers. The architecture of Deconvnet is shown in figure 7.

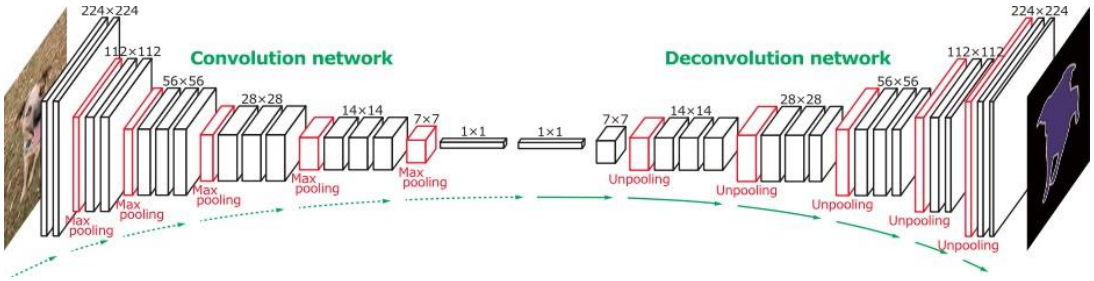

Figure 7: Convolution-Deconvolution architecture of Decovnet. From [90]

U-Net: U-Net [91] is a U-shaped semantic segmentation network which has a contracting path and an expansive path. Every step of the contracting path consists of two consecutive 3×3 convolutions followed by ReLU nonlinearity and max-pooling using 2×2 window with stride 2. During the contraction, the feature information is increased while spatial information is decreased. On the other hand, every step of the expansive path consists of up-sampling of feature map followed by a 2×2 up-convolution. This reduces the feature map size by a factor of 2. Then the reduced feature map is concatenated with the corresponding cropped feature map from the contracting path. Then two consecutive 3×3 convolution operations are applied followed by ReLU nonlinearity. In this way, the expansive pathway combines the features and spatial information for precise segmentation. The architecture of U-Net is shown in figure 8.

SegNet: SegNet [93] has encoder-decoder architecture followed by a final pixel-wise classification layer. The encoder network has 13 convolutional layers as in VGG16 [8] and the corresponding decoder part also has 13 de-convolutional layers. The authors did not use fully connected layers of VGG16 to retain the resolution in the deepest layer and it reduces the number of parameters from 134M to 14.7M. In each layer in the encoder network, a convolutional operation is performed using a filter bank to produce feature maps. Then, to reduce internal covariate shift the authors have used batch normalization [124] [125] followed by ReLU [126] nonlinearity operation. Resulting output feature maps are max-pooled using a 2×2 non-overlapping window with stride 2 followed by a sub-sampling operation by a factor of 2. A combination of max-pooling and sub-sampling operation achieves better classification accuracy but reduces the feature map size which leads to lossy image representation with blurred boundaries which is not ideal for segmentation purposes where boundary information

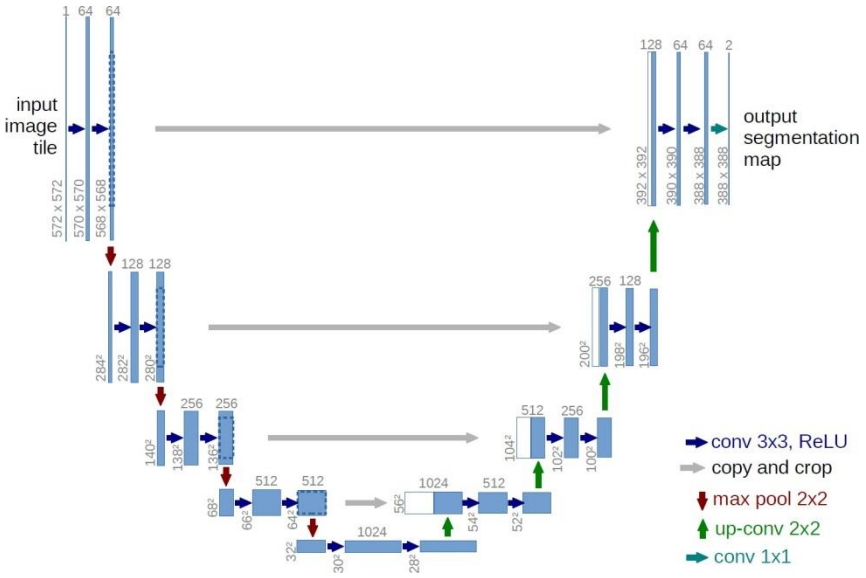

Figure 8: Contracting and expansive architecture of U-Net. From [91]

is important. To retain boundary information in the encoder feature maps before sub-sampling, SegNet stores only the max-pooling indices for each encoder map. For semantic segmentation, the output image resolution should be the same as the input image. To achieve this, SegNet does up-sampling in its decoder using the stored max-pooling indices from the corresponding encoder feature map resulting high-resolution sparse feature map. To make the feature maps dense, the convolution operation is performed using a trainable decoder filter bank. Then the feature maps are batch normalized. The high-resolution output feature map produced form final decoder is fed into a trainable multi-class softmax classifier for pixel wise labeling. The architecture of SegNet is shown in figure 9.

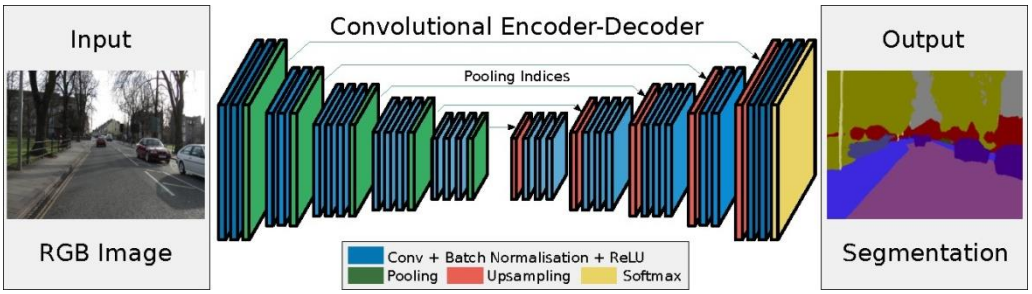

Figure 9: Encoder-decoder architecture of SegNet. From [93]

FC-DenseNet: DenseNet [10] is a CNN based classification network that contains only a down-sampling pathway for recognition. Jégou et al. [127] has extended DenseNet by adding an up-sampling pathway to regain the full resolution of the input image. To con- struct the up-sampling pathway, the authors followed the concept of FCN. They have referred the down-sampling operation of DenseNet as Transition Down (TD) and up-sampling oper- ation in extended DenseNet as Transition UP (TU) as shown in figure 10. The rest of the convolutional layers follows the sequence of Batch Normalization, ReLU, 3 3 convolution and dropout of 0.2 as shown in the top right block in figure 10. The up-sampling pathway used the sequence of dense block [10] instead of convolution operation of FCN and used transposed convolution as an up-sampling operation. The up-sampling feature maps are

concatenated with the feature maps derived from corresponding layers of the down-sampling pathway. In figure 10, these long skip connections are shown as yellow circle.

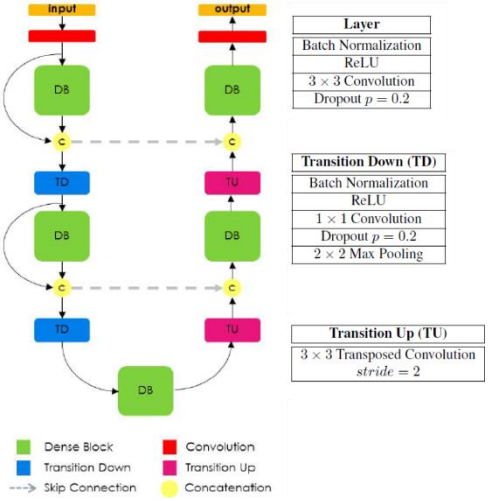

Figure 10: Architecture of Fully Convolutional DenseNet for semantic segmentation with some building blocks. From [127]

As the upsampling rate of FCN based model in final layer is very high, it produces coarse output in final layer. So fine-grained semantic segmentation is not possible. On the other hand top-down/bottom-up approach based models used gradually increasing upsampling rate which leads to more accurate segmentation. But in this case the model also lacks incorporation of global contextual information.

Based on Global Context:

ParseNet: Liu et al. proposed an end-to-end architecture called ParseNet [94] which is an improvement of Fully Convolution Neural Network [83]. The authors have added global feature or global context information for better segmentation. In figure 11, the model description of ParseNet is shown. Till convolutional feature map extraction, the ParseNet is the same as FCN [83]. After that, the authors have used global average pooling to extract global contextual information. Then the pooled feature maps are un-pooled to get the same size as input feature maps. Now, the original feature maps and un-pooled feature maps are combined for predicting the final classification score. As the authors have combined two different feature maps from two different layers of the network, those feature maps would be different in scale and norm. To make the combination work, they have used two L2 normalization layers: one after global pooling and another after the original feature map extracted from FCN simultaneously. This network achieved state-of-the-art performance on ShiftFlow [76], PASCAL-context [128] and near the state of the art on PASCAL VOC 2012 dataset.

GCN: Like ParseNet, Global Convolution Network [97] has also used global features along with local features to make the pixel-wise prediction more accurate. The task of semantic segmentation is the combination of classification and localization tasks. These two tasks are contradictory in nature. The classification should be transformation invariant and localization should be transformation sensitive. Previous state-of-the-art models focused on localization more than classification. In GCN, the authors did not use any fully connected layers or global pooling layers to retain spatial information. On the other hand, they have used a large kernel size (global convolution) to make their network transformation invariant

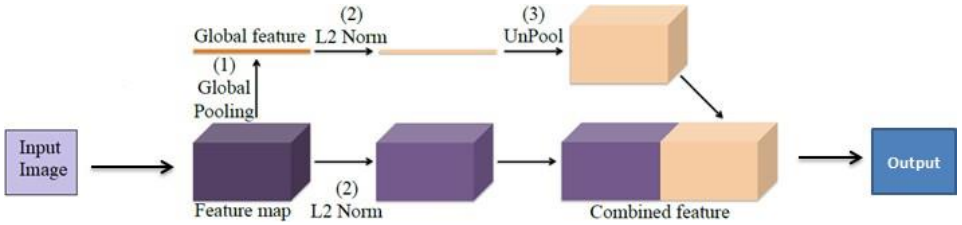

Figure 11: ParseNet Model Design [94]

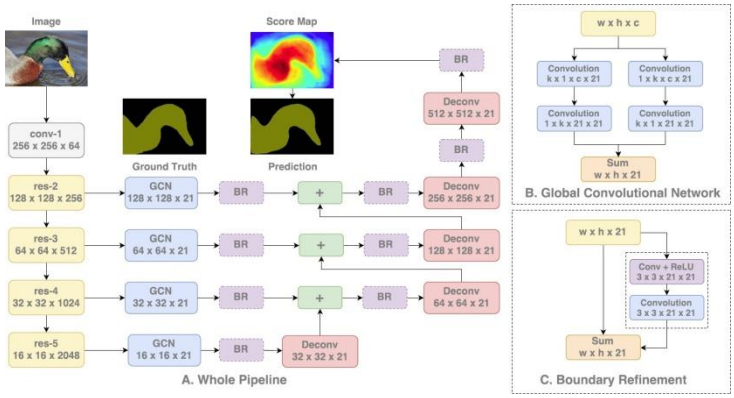

Figure 12: Pipeline network of GCN. From [97]

in the case of pixel-wise classification. To refine the boundary further the authors have used Boundary Refinement (BR) block. As shown in figure 12, ResNet is used as a backbone. GCN module is inserted in the network followed by the BR module. Then score maps of lower resolution are up-sampled with a deconvolution layer, and then added up with higher ones to generate new score maps for final segmentation.

EncNet: Zhang et al. have applied the idea of global context introducing novel context encoding module. The authors have used Semantic Encoding Loss(SE-loss) to help incorporation of global scene context information. This loss unit helps in regularizing the network training procedure in such a way that the network can predicts the presence of different category objects as well as learns the semantic context of an image. Using the above mentioned idea, the authors have proposed Context Encoding Network (EncNet)[129] as shown in figure 13. The network contains a pre-trained Deep ResNet. On top of the ResNet, Context Encoding Model is used. The authors have used dialation strategy, multi GPU Batch Normalization and Memory efficient encoding layer to enhance the acuuracy of semantic segmetation.

Though application of global convolution helps to improve accuracy but it lacks the scaling information of multi scale objects.

3.2.5. Based on receptive field enlargement and multi-scale context incorporation:

DeepLabv2 and DeepLabV3: The authors of DeepLab modified their network using Atrous Special Pooling Pyramid (ASPP) to aggregate multi-scale features for better local-ization and proposed DeepLabv2 [88]. Figure 14 shows ASPP. This architecture used both ResNet [10] and VGGNet[8] as base network.

In DeepLabv3[89], to incorporate multiple contexts in the network, the authors have used cascaded modules and have gone deeper especially with the ASPP module.

PSPNet: Pyramid Scene Parsing Network(PSPNet) [95], proposed by Zhao et al., has

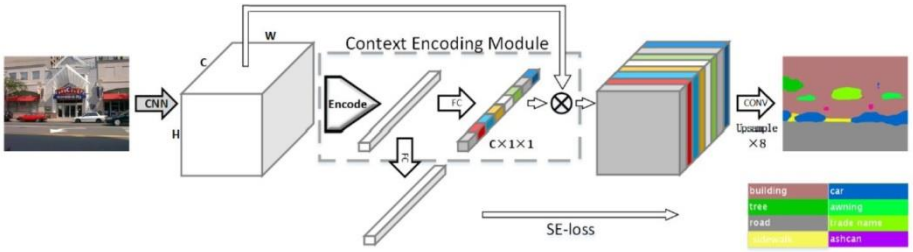

Figure 13: EncNet from [129] (Notation: FC → Fully Convolutional Layer,Conv → Convolutional Layer, Encode → Encoding Layer and ⊗ → Channel wise multiplication)

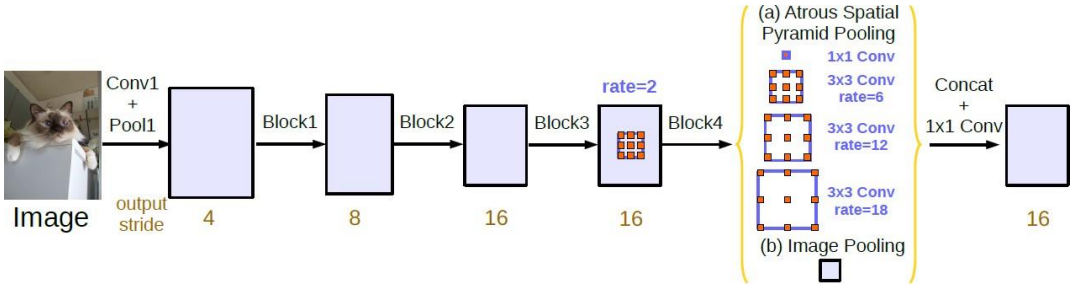

Figure 14: Atrous Spatial Pooling Pyramid. From [89]

used global contextual information for better segmentation. In this model, the authors have used Pyramid Pooling Module on top of the last feature map extracted using dilated FCN. In Pyramid Pooling Module, feature maps are pooled using 4 different scales corresponding to 4 different pyramid levels each with bin size 1×1, 2×2, 3×3 and 6×6. To reduce dimension,

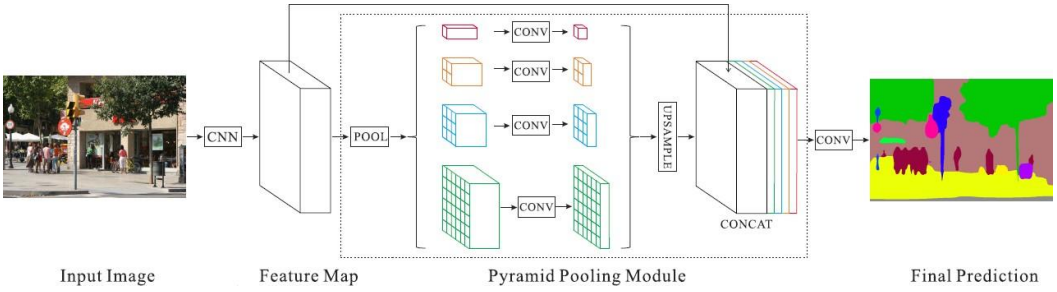

Figure 15: PSPNet Model Design. From [95]

the pooled feature maps are convolved using 1 1 convolution layer. The output of the convolution layers are up-sampled and concatenated to the initial feature maps to finally combine the local and the global contextual information. Then, those output are again processed by a convolutional layer to generate the pixel-wise prediction. In this network, the pyramid pooling module observes the whole feature map in sub-regions with a different locations. In this way, the network understands a scene better which also leads to better semantic segmentation. In figure 15, the architecture of PSPNet is shown.

Gated-SCNN: Takikawa et al. proposed Gated - Shape CNN(GSCNN) [99] for Se- mantic Segmentation. As shown in figure 16, GSCNN consists of two streams of networks: regular stream and shape stream. The regular stream is a classical CNN for processing semantic region information. Shape stream consists of multiple Gated Convolution Layer

(GCL) which process boundary information of regions using low-level feature maps from the regular stream. Outputs of both streams are fed into a fusion module. In fusion modules, both outputs are combined using Atrous Special Pyramid Pooling [88] module. The use of ASPP helps their model to preserve multi-scale contextual information. Finally, the Fusion module produced semantic region of objects with a refined boundary.

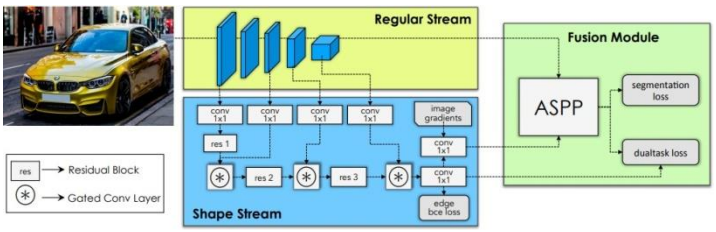

Figure 16: Architecture of Gated Shape CNN for semantic segmentation. From [99]

The process of enlarging the receptive field using multi-resolution pyramid based representation helps the above model to incorporate scale information of objects to acquire fine-grained semantic segmentation. But capturing contextual information using receptive field enlargement may not be the only solution left for better semantic segmentation.

Discussion

From the year 2012, different CNN based semantic segmentation models have emerged in successive years to date. In subsection 3, we have described major up-gradation in the networks of various state-of-the-art models for better semantic segmentation. Among different models, Fully Convolutional Network (FCN) has set a path for semantic segmentation. Various models have used FCN as their base model. DeepLab and its versions have used atrous algorithm in different ways. SegNet, DeconvNet, U-Net have a similar architecture where the second part of those architectures is hierarchically opposite of the first half. ParseNet, PSPNet, and GCN have addressed semantic segmentation with respect to global contextual information. FC-DenseNet used top-down/bottom-up approach to incorporate low-level features with high-level features. So far, we have seen that the performance of a semantic segmentation model depends on the internal architecture of a network. In following subsections, we will see that it also depends on some other aspects such as the size of the data set, number of semantically annotated data, different training hyperparameters, optimization algorithm, loss function, etc. We have shown those different comparative aspects of each model in tabular form.

Optimization details of different State-of-the-art Semantic Segmentation Models:

Table 1 shows different optimization details of different models where we can see that the success of a model not only depends on the architecture. Comparison of different models with respect to optimization or training details shows that most of the researcher used stochastic gradient descent (SGD) as optimization algorithm but with different mini batch size of images. The choice of mini-batch size depends also on the number of GPU used to train a particular model.It is also shown here that most of the researcher have used momentum approximately same as 0.9. The main important feature in training a model is choosing the learning rate so that the model can converge in a optimized way. Regularization term is also an important factor in model training to combat overfitting. So, as network design, choosing appropriate hyperparameters is also a crucial thing in training to reach to a desirable accuracy.

Table 1: Optimization details of different state-of-the-art semantic segmentation models

Name of the model	Optimization Algorithm	Mini Batch Size	Learning Rate	Momentum	Weight Decay
FCN-VGG16 [83]	SGD [130]	20 images	0.0001	0.9	0.0016 or 0.0625
DeepLab [87]	SGD	20 images	initially 0.001 (0.01 for final classification layer), increasing it by 0.1 at every 2000 iteration.	0.9	0.0005
Deconvnet [90]	SGD	-	0.01	0.9	0.0005
U-Net [91]	SGD	Single image		0.99	
DialatedNet [120]	SGD	14 images	0.001	0.9	-
ParseNet [94]	SGD		1e – 9	0.9	
SegNet [93]	SGD	12 images	0.1	0.9	
GCN [97]	SGD	Single image		0.99	0.0005
PSPNet [95]	SGD	16 images	'poly' learning rate with base learning rate of 0.01 and power to 0.9	0.9	0.0001
FC-DenseNet103 [127]	SGD		initially 1e – 3 with an exponential decay of 0.995		1e – 4
EncNet [129]	SGD	16 images	0.001 with the power of 0.9	0.9	0.0001
Gated-SCNN [99]	SGD	16 images	1e – 2 with polynomial decay policy	0	

Table 2: Base Model, data preprocessing technique and loss functions of different state-of-the-art semantic segmentation models.

Name of the model	Base Network	Data pre-processing	Loss Funtion
FCN-VGG16 [83]	AlexNet[6], VGGnet[8], GoogLeNet[9] (All pre-trained on ILSVRC dataset [116])		Per-pixel multinomial logistic loss
DeepLab [87]	VGG16 [8] pre-trained on ILSVRC dataset	Data augmentation using extra annotated data of [131]	Sum of cross-entropy loss
Deconvnet [90]	VGG16 pre-trained on ILSVRC dataset	Data augmentation using extra annotated data of [131]	
U-Net [91]	FCN [83]	Data augmentation by applying random elastic deformation to the available training images	Cross entropy loss
DialateNet [120]	VGG16 [8]	Data augmentation using extra annotated data of [131]	
ParseNet [94]	FCN [83]		
SegNet [93]	VGG16 [8]	Local contrast normalization to RGB data	Cross entropy loss
GCN [97]	ResNet152 [10] as feature network and FCN-4 [83] as segmentation network	Semantic Boundaries Dataset [131] is used as auxiliary dataset	
PSPNet [95]	Pretrained ResNet [10]	Data augmentation: random mirror and random resize between 0.5 and 2, random rotation between -10 and 10 degrees, random Gaussian blur	Four losses: • Additional loss for initial result generation • Final loss for learning the residue later • Auxiliary loss for shallow layers • Master branch loss for final prediction
FC-DenseNet [127]	DensNet [10]	Data augmentation using random cropping and vertical flipping	
EncNet [129]	ResNet	Data augmentation using random flipping, scaling, rotation and finally cropping	Semantic Encoding loss
Gated-SCNN [99]	ResNet101[10] and WideResNet[132]		• Segmentation loss for regular stream • Dual task loss for shape stream •• Standard binary cross entropy loss for boundary refinement •• Standard cross entropy for semantic segmentation

Table 2 presents base network (pre-trained on ImageNet [135] dataset), data pre-processingtechnique (basically data augmentation) and different loss function used for different models. Choosing of base network of a semantic segmentation model changes overtime according to
385 the evolution of classification model. Optimization of a model always starts with some kind

Table 3: Some important features of different state-of-the-art semantic segmentation models

Model	Important Features
FCN-VGG16	• Dropout is used to reduce overfitting • End to end trainable
DeepLab	• End to end trainable • Piecewise training for DCNN and CRF • Inference time during testing is 8frame per second • Used Atrous Special Pyramid Pooling module for aggregating multi-scale features
Deconvnet	• Used edge-box to generate region proposal • Used Batch Normalization to reduce internal covariate shift and removed dropout • Two-stage training for easy examples and for more challenging examples • End to end trainable • Drop-out layer is used at the end of the contracting path
U-Net	• End to end trainable • Inference time for testing was less than 1 sec per image
DialatedNet	Two stage training: • Front end module with only dilated convolution • Dilated convolution with multi-scal context module
ParseNet	• End to end trainable • Batch Normalization is used • Drop-out of 0.5 is used in deeper layers
SegNet	• Different Ablation study
GCN	• Large Kernel Size • Included Global Contextual information
PSPNet	• End to end training • Contains dialated convolution • Batch normalization • Used pyramid pooling module for aggregating multi-scale features
FC-DensNet	• Initialized the model with HeUniform[133] and trained it with RMSprop dataset[134] • Used dropout of 0.2 • Used the model parameters efficiently
EncNet	• Used Context Encoding Module • Applied SE-loss to incorporate global semantic context
Gated -SCNN	• End to end trainable • Applied ablation study

of data pre-processing. Different researcher have used different technique to pre-process the data. Most commonly used data preprocessing technique is data augmentation. As a loss function, cross entropy loss is used in most cases. Also, according to the complexity of model design, researchers have used different loss function to get higher accuracy. So, the choice of base network, data pre-processing technique, loss function etc are also very important to design a successful model.

To give a clear view on the success of each model, we have listed some important features of each state-of-the-art model in table 3.

Comparative Performance of State-of-the-art Semantic Segmentation Models:

In this section, we are going to show the comparative result of different state-of-the-art semantic segmentation models on various datasets in table 4. The performance metric used here is mean average precision (mAP) as Intersection over Union (IoU) threshold. To have a better understanding, we have listed them in chronological order.

4. Instance Segmentation

Like semantic segmentation, the applicability of CNN has been spread over instance segmentation too. Unlike semantic segmentation, instance segmentation masks each instance of an object contained in an image independently [140, 141]. The task of object detection and instance segmentation are quite correlated. In object detection, researchers use the bounding box to detect each object instance of an image with a label for classification. Instance segmentation put this task one step forward and put a segmentation mask for each instance.

Table 4: Comparative accuracy of different semantic segmentation models in terms of mean average precision (mAP) as Intersection over Union (IoU)

Model	Year	Used Dataset	mAP as IoU
FCN-VGG16 [83]	2014	Pascal VOC 2012 [81]	62.2%
DeepLab[87]	2014	Pascal VOC 2012	71.6%
Deconvnet[90]	2015	Pascal VOC 2012	72.5%
U-Net[91]	2015	ISBI cell tracking challenge 2015	92% on PhC-U373 and 77.5% on DIC-HeLa dataset
DialatedNet [120]	2016	Pascal VOC 2012	73.9%
ParseNet [94]	2016	• ShiftFlow [76] • PASCAL- Context [128] • Pascal VOC 2012	40.4% 36.64% 69.8%
SegNet [93]	2016	• CamVid road scene segmentation [136] • SUN RGB-D indoor scene segmentation[137]	60.10% 31.84%
GCN[97]	2017	• PASCAL VOC 2012 •Cityscapes [138]	82.2% 76.9%
PSPNet [95]	2017	• PASCAL VOC 2012 • Cityscapes	85.4% 80.2%
FC-DenseNet103 [127]	2017	• CamVid road scene segmentation • Gatech[139]	66.9% 79.4%
EncNet [129]	2018	• Pascal VOC 2012 • Pascal Context	85.9% 51.7%
Gated-SCNN [99]	2019	• Cityscapes	82.8%

Concurrent to semantic segmentation research, instance segmentation research has also started to use the convolutional neural network(CNN) for better segmentation accuracy. Herein, we are going to survey the evolution of CNN based instance segmentation models. In addition, we are going to bring up here an elaborate exploration of some state-of-the-art models for instance segmentation task.

4.1. Evolution of CNN based Instance Segmentation Models:

CNN based instance segmentation has also started its journey along with semantic segmentation. As we have mentioned in section 4 that instance segmentation task only adds a segmentation mask to the output of object detection task. That is why most of the CNN based instance segmentation models have used different CNN based object detection models to produce better segmentation accuracy and to reduce test time.

Hariharan et al. have followed the architecture of R-CNN [79] object detector and proposed a novel architecture for instance segmentation called Simultaneous Detection and Segmentation(SDS) [141] which is a 4 step instance segmentation model as described in section 4.2.1.

Till this time CNN based models have only used the last layer feature map for classification, detection and even for segmentation. In 2014, Hariharan et al. have again proposed a concept called Hyper-column [142] which has used the information of some or all intermediate feature maps of a network for better instance segmentation. The authors added the concept of Hyper-column to SDS and their modified network achieved better segmentation accuracy.

Different object detector algorithms such as R-CNN, SPPnet [17], Fast R-CNN [18] have used two stages network for object detection. The first stage detects object proposals using Selective Search [80] algorithm and second stage classify those proposals using different CNN based classifier. Multibox [143, 144], Deepbox [145], Edgebox [146] have used CNN based proposal generation method for object detection. Faster R-CNN [19] have used CNN based 'region proposal network (RPN)' for generating box proposal. However, the mode of all these proposal generations is using a bounding box and so the instance segmentation models. In parallel to this, instance segmentation algorithms such as SDS and Hyper-column have used Multi-scale Combinatorial Grouping (MCG)[147] for region proposal generation. DeepMask

[148], as discussed in section 4.2.2, has also used CNN based RPN as Faster R-CNN to generate region proposals so that the model can be trained end to end.

Previous object detection and instance segmentation modules such as [79], [17], [18], [19] [141], [142], [148] etc. have used computationally expensive external methods for generating object level or mask level proposals like Selective Search, MCG, CPMC [82], RPN etc. Dai et al. [149] break the tradition of using a pipeline network and did not use any external mask proposal method. The authors have used a cascaded network for incorporating features from different CNN layers for instance segmentation. Also, the sharing of convolution features leads to faster segmentation models. Detail of the network is discussed in section 4.2.1.

SDS, DeepMask, Hyper-columns have used feature maps from top layers of the network for object instance detection which leads to coarse object mask generation. Introduction of skip connection in [150, 151, 152, 153] reduces the coarseness of masks which is more helpful for semantic segmentation rather instance segmentation. Pinheiro et. al.[154] have used their model to generate a coarse feature map using CNN and then refined those models to get pixel-accurate instance segmentation masks using a refinement model as described in section 4.2.2.

In papers [155], [156], [85], [17], [157], [142], [83], [158], researchers used contextual information and low level features into CNN in various ways for better segmentation. Zagoruko et al. [153] has also used those ideas by integrating skip connection, foveal structure and integral loss in Fast R-CNN [18] for better segmentation. Further description is given in section 4.2.3.

Traditional CNNs are translation invariant i.e images with the same properties but with different contextual information will score the same classification score. Previous models, specially FCN, used a single score map for semantic segmentation. But for instance segmentation, a model must be translation variant so that the same image pixel of different instances having different contextual information can be segmented separately. Dai et al [21] integrated the concept of relative position into FCN to distinguish multiple instances of an object by assembling a small set of score maps computed from the different relative positions of an object. Li et al [159] extended the concept of [21] and introduced two different position-sensitive score maps as described in section 4.2.4.

SDS, Hypercolumn, CFM [160], MNC [149], MultiPathNet[153] used two different sub-networks for object detection and segmentation which prevent the models to become an end to end trainable. On the other hand [161],[162] extends instance segmentation by grouping or clustering FCNs score map which involves a large amount of post-processing. [159] introduced a joint formulation of classification and segmentation masking sub-networks in an efficient way.

While [163, 164, 165, 166] have used semantic segmentation models, Mask R-CNN [20] extends the object detection model Faster R-CNN by adding a binary mask prediction branch for instance segmentation. In [167], Huang et al infused a network block in Mask R-CNN to learn the predicted mask in a qualitative way and proposed Mask Scoring R-CNN. Recently, Kirillov et al.[168] used point-based rendering in Mask R-CNN and produce state-of-the-art instance segmentation model.

The authors of [169], [170] has introduced direction features to predict different instances of a particular object. [169] has used template matching technique with direction feature to extract the center of an instance whereas [170] followed the assembling process of [159, 21] to get instances.

The papers [171, 172, 164, 142] have used features form intermediate layers for better performance. Liu et al.[173] have also used the concept of feature propagation from a lower level to top-level and built a state-of-the-art model based on Mask R-CNN as discussed in section 4.2.5. In [174], Newell et al used a novel idea to use CNN with associative embedding

for joint detection and grouping to handle instance segmentation.

Object detection using the sliding window approach gave us quite successful work such as Faster R-CNN, Mask R-CNN, etc. with refinement step and SSD[23], RetinaNet[26] without using refinement stage. Though sliding window approach is popular in object detection but it was missing in case of instance segmentation task. Chen et al. [175] have introduced dense instance segmentation to fill this gap and introduced TensorMask.

Some State-of-the-art Instance Segmentation Models:

In this section, we are going to elaborately discuss architectural details of some state-of-the-art CNN based instance segmentation models. The models are categorized on the basis of the most impotant feature used. At the end of each categorical discussion, we have also briefly discussed the advantages and weaknesses of a particular model category in brief.

Based on bounding box proposal generation:

SDS: Simultaneous Detection and Segmentation (SDS) [141] model consists of 4 steps for instance segmentation. The steps are proposal generation, feature extraction, region classification, and region refinement respectively. On input image, the authors have used

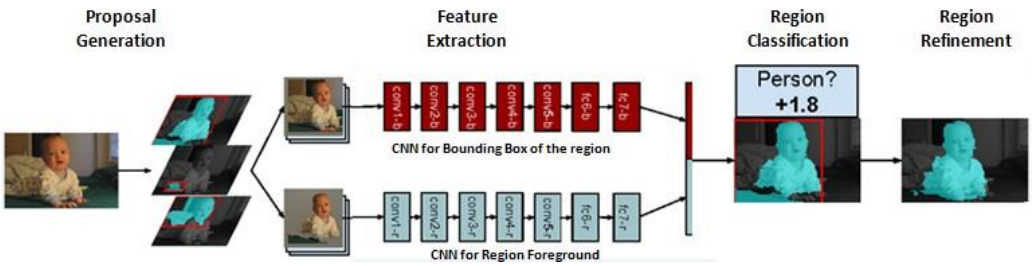

Figure 17: Architecture of SDS Network. From [141]

Multi-scale Combinatorial Grouping(MCG) [147] algorithm for generating region proposals. Then each region proposals are fed into two CNN based sibling networks. As shown in figure 17, the upper CNN generates a feature vector for bounding box of region proposals and the bottom CNN generates a feature vector for segmentation mask. Two feature vectors are then concatenated and class scores are predicted using SVM for each object candidate. Then non-maximum suppression is applied on the scored candidates to reduce the set of same category object candidates. Finally, to refine surviving candidates CNN feature maps are used for mask prediction.

Multi-task Network Cascades (MNC): Dai et al. [149] used a network with the cascaded structure to share convolutional features and also used region proposal network (RPN) for better instance segmentation. The authors have decomposed the instance segmentation task into three sub tasks: instance differentiation (class agnostic bounding box generation for each instance), mask estimation (estimated a pixel-level mask/instance) and object categorization (instances are labeled categorically). They proposed Multi-task Network Cascades (MNC) to address these sub-tasks in three different cascaded stages to share convolutional features. As shown in figure 18, MNC takes an arbitrary sized input which is a feature map extracted using VGG16 network. Then at the first stage, the network generates object instances from the output feature map as class agnostic bounding boxes with an objectness score using RPN. Shared convolutional features and output boxes of stage-1 then go to the second stage for regression of mask level class-agnostic instances. Again, shared

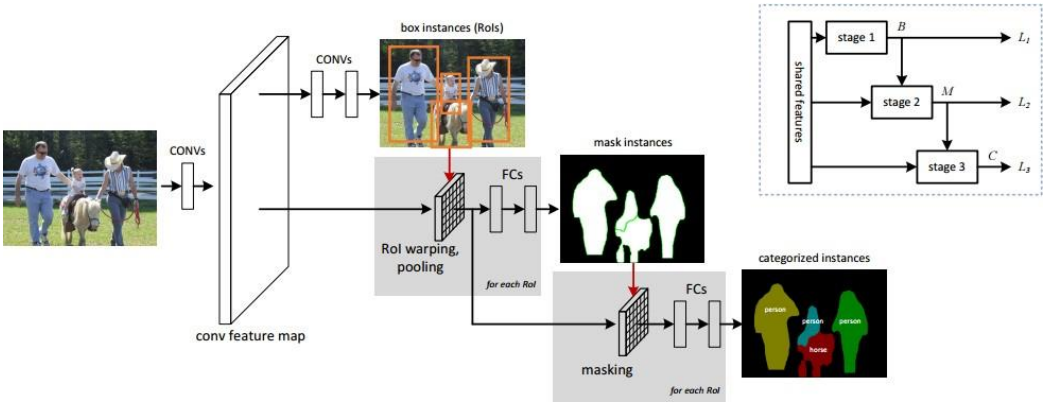

Figure 18: Three stage architecture of Multi-task Network Cascades. From [149].

convolutional features and output of the previous two stages are fed into the third stage for generating category score for each instance.

Mask R-CNN: Mask R-CNN[20] contains three branches for predicting class, bounding-box and segmentation mask for instances within a region of interest (RoI). This model is the extension of Faster R-CNN. As Faster R-CNN, Mask R- CNN contains two stages. In the first stage, it uses RPN to generate RoIs. Then to preserve the spatial location, the authors have used RoIAlign instead of RoIPool as in Faster R-CNN. In the second stage, it simultaneously predicts a class label, a bounding box offset and a binary mask for each individual RoI. In Mask R-CNN, the prediction of binary mask for each class was independent and it was not a multi-class prediction.

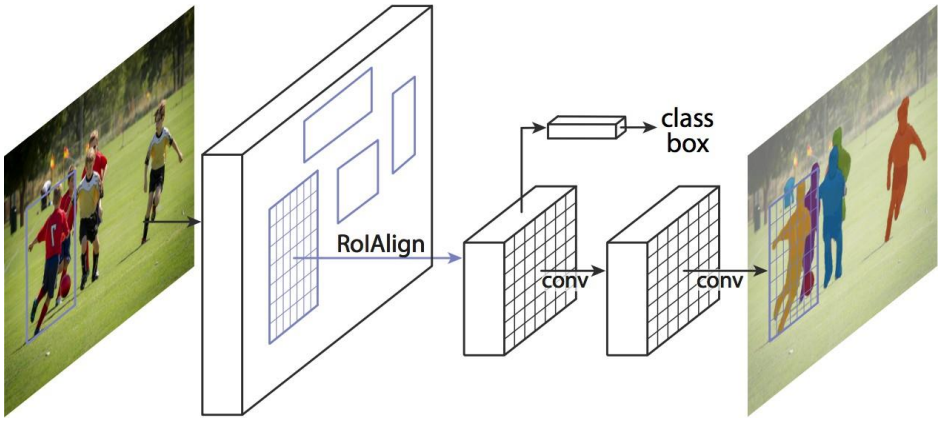

Figure 19: Architecture of Mask R-CNN. From[20].

Generation of bounding box is computationally cost effective and its very helpful for detecting object. But its leads to computationally expensive alignment procedures. Also bounding box generation based models needs to generate masks for each instance separately. To overcome this problem researcher tried to generate segmentation mask proposal instead of bonding box proposal.

Based on segmentation mask proposal generation:

DeepMask: DeepMask [148] used CNN to generate segmentation proposals rather than less informative bounding box proposal algorithms such as Selective Search, MCG, etc.

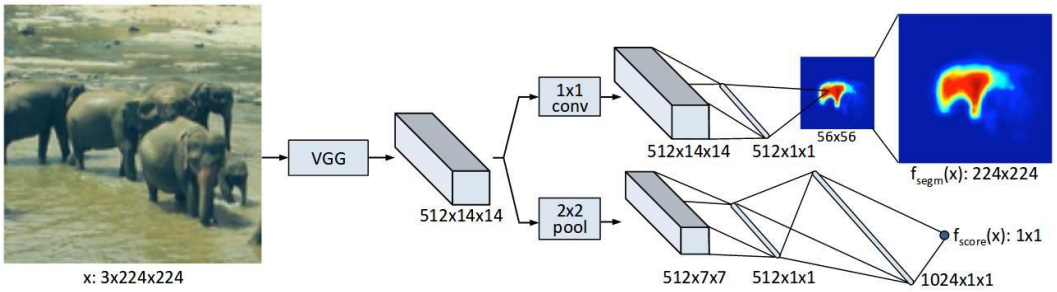

Figure 20: Model illustration of DeepMask. From [148].

DeepMask used VGG-A [8] model (discarding last max-pooling layer and all fully connected layers) for feature extraction. As shown in figure 20, the feature maps are then fed into two sibling branches. The top branch which is the CNN based object proposal method of DeepMask predicts a class-agnostic segmentation mask and bottom branch assigns a score for estimating the likelihood of patch being centered on the full object. The parameters of the network are shared between the two branches.

SharpMask: DeepMask generates accurate masks for object-level but the degree of alignment of the mask with the actual object boundary was not good. SharpMask [154] contains a bottom-up feed-forward network for producing coarse semantic segmentation mask and a top-down network to refine those masks using a refinement module. The authors have used feed-forward DeepMask segmentation proposal network with their refinement module and named it as SharpMask. As shown in figure 21, the bottom-up CNN architecture produces coarse mask encoding. Then the output mask encoding is fed into a top-down architecture where a refinement module un-pool it using matching features from the bottom-up module. This process continues until the reconstruction of the full resolution image and the final object mask.

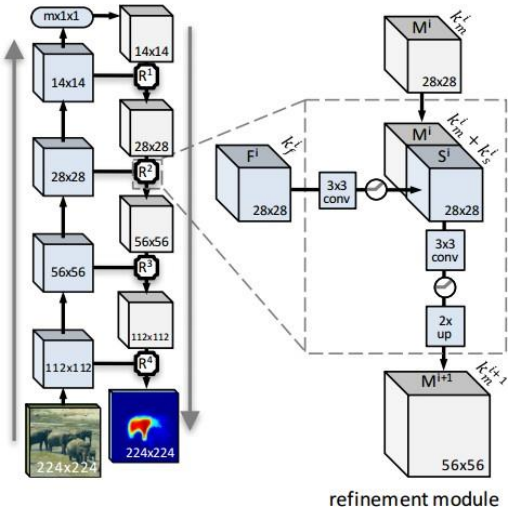

Figure 21: Bottom-up/top-down architecture of SharpMask. From [154].

Segmentation mask proposal generation using CNN helps the models to have better accuracy. But it does not have the power of capturing instances of object with different scales.

Based on multi-scale feature incorporation:

MultiPath Network: Zagoruko et al. integrate three modifications in the Fast R-CNN object detector and proposed Multipath Network [153] for both object detection and segmentation tasks. Three modifications are skip connections, foveal structure, and integral loss. Recognition of small objects without context is difficult. That is why, in [155], [156], [85], [17], [176], the researcher used contextual information in various ways in CNN based model for better classification of objects. In Multipath Network, the authors have used four contextual regions called foveal regions. The view size of those regions are 1×, 1.5×, 2×, 4× of the original object proposal. On the other hand, researchers of [157], [142],

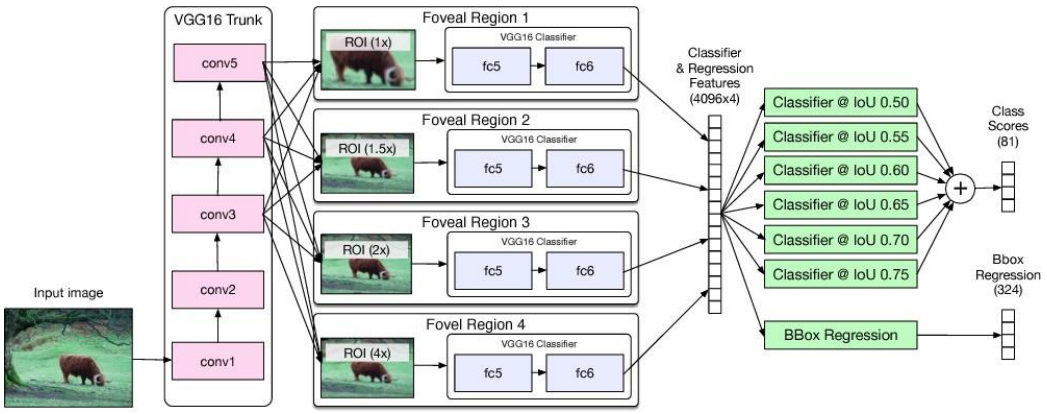

Figure 22: Architecture of MultiPath Network. From [153].

[83], [158] has used feature from higher-resolution layers of CNN for effective localization of small objects. In Multipath Network, the authors have connected third, fourth and fifth convolutional layers of VGG16 to the four foveal regions to use multi-scale features for better object localization. Figure 22 shows the architectural pipeline of MultiPath Network. Feature maps are extracted from an input image using the VGG16 network. Then using skip connection those feature maps go to four different Foveal Region. The output of those regions are concatenated for classification and bounding box regression. The use of the DeepMask segmentation proposal helped their model to be the 1st runner-up in MS COCO 2015 [177] detection and segmentation challenges.

This model tried to incorporate multi-scale feature maps to become scale invariant and also used skip connection to incorporate contextual information for better segmentation. But it lacks knowledge about relative position of an object instances.

Based on capturing relative position of object instances:

InstanceFCN: The fully convolutional network (FCN) is good for single instance segmentation of an object category. But it can not distinguish multiple instances of an object. Dai et al have used the concept of relative position in FCN and proposed instance sensitive fully convolutional network (InstanceFCN) [21] for instance segmentation. The relative position of an image is defined by a $k \times k$ grid on a square sliding window. This produces a set of k^2 instance sensitive score maps rather than one single score map as FCN. Then the instance sensitive score maps are assembled according to their relative position in a $m \times m$ sliding window to produce object instances. In DeepMask[148], shifting sliding window for one stride leads to the generation of two different fully connected channels for the same pixel which is computationally exhaustive. In InstanceFCN, the authors have used the concept of local coherence [178] which means sliding a window does not require different computations for a single object. Figure 23 shows the architecture of InstanceFCN.

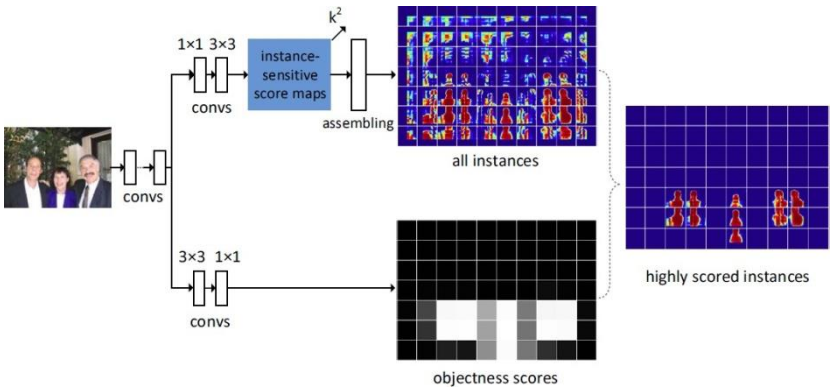

Figure 23: Architecture of Instance-sensitive fully convolutional network. From [21].

FCIs: InstanceFCN introduced position-sensitive score mapping to signify the relative position of an object instance but the authors have used two different sub networks for object segmentation and detection. Because of two different networks, the solution was not end to end. Li et al. [159] proposed the first end to end trainable fully convolutional network based model in which segmentation and detection are done jointly and concurrently in a single network by score map sharing as shown in figure 24. Also instead of the sliding window approach, the model used box proposals following [19]. The authors have used two different position-sensitive score maps: position-sensitive inside score maps and position sensitive outside score maps. These two score maps depend on detection score and segmentation score of a pixel in a given region of interests (RoIs) with respect to different relative position. As shown in figure 24 RPN is used to generate RoIs. Then RoIs are used on score maps to detect and segment object instances jointly.

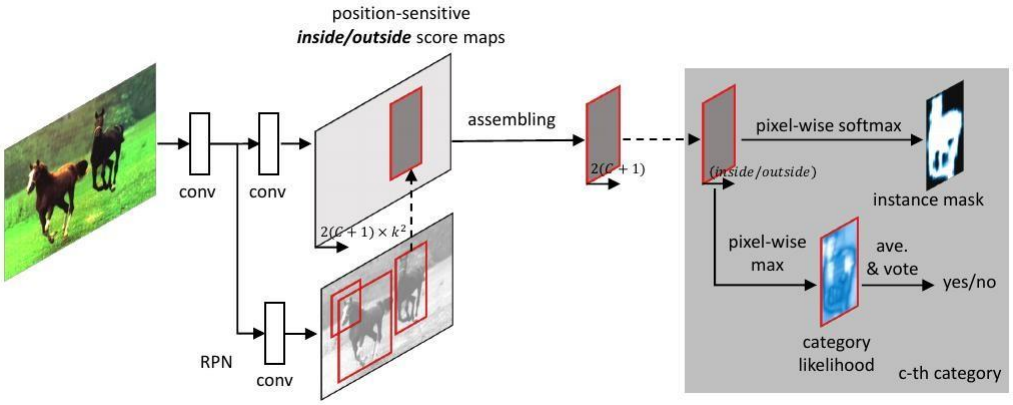

Figure 24: Architecture of FCIs. From [159].

MaskLab: MaskLab [170] has utilized the merits of both semantic segmentation and object detection to handle instance segmentation. The authors have used Faster R-CNN[19] (ResNet-101[10] based) for predicting bounding boxes for object instances. Then they have calculated semantic segmentation score maps for labeling each pixel semantically and direc- tion score maps for predicting individual pixels direction towards the center of its correspond-ing instance. Those score maps are cropped and concatenated for predicting a coarse mask for target instance. The mask is then again concatenated with hyper-column features[142] extracted from low layers of ResNet-101 and processed using a small CNN of three layers for

further refinement.

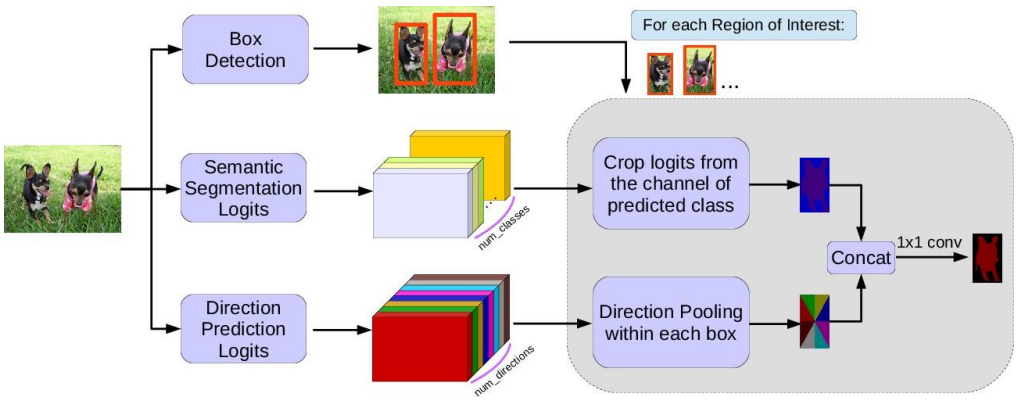

Figure 25: Architecture of MaskLab. From [170].

Using position sensitive score maps the above models tried to capture the relative position of object instances.

Based on feature propagation:

Aneural network is very important as the low-level feature maps are information-rich in terms of localization and the high-level feature maps are rich in semantic information. Liu et al. focused on this idea. Basedon Mask R-CNN and Feature Pyramid Network(FPN) [179], they have proposed a Path Aggregation Network (PANet) [173] for instance segmentation. PANet used FPN as its basenetwork to extract features from different layers. To propagate the low layer feature through the network, a bottom-up augmented path is used. Output of each layer is generated using previous layers high-resolution feature map and a coarse map from FPN using a lateral connection. Then an adaptive pooling layer is used to aggregate features from all levels. Inthis layer, a RoIPooling layer is used to pool features from each pyramid level and elementwise max or sum operation is used to fuse the features. As Mask R-CNN, the output of thefeature pooling layer goes to three branches for prediction of the bounding box, prediction of the object class and prediction of the binary pixel mask.

Using feature propagation network and pooling pyramid, this model incorporates low to high level feature as well as multi-scale features which leads to information rich instance segmentation.

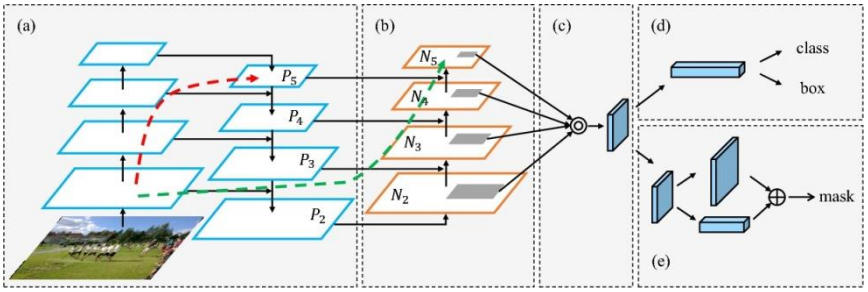

Figure 26: Architecture of PANet. From [173].

Based on sliding window approach:

Tensor Mask: Previous instance segmentation models used methods in which the objects are detected using bounding box then segmentation is done. Chen et al. have used the

dense sliding window approach instead of detecting the object in a bounding box named TensorMask [175]. The main concept of this architecture is the use of structured high-dimensional (4D) tensors to present mask over an object region. A 4D tensor is a quadruple of (V, U, H, W). The geometric sub-tensor (H, W) represents object position and (V, U) represents the relative mask position of an object instance. Like feature pyramid network, TensorMask has also developed a pyramid structure, called *tensorbipyramid* over a scale-indexed list of 4D tensors to acquire the benefits of multi-scale.

Discussion:

In the previous subsection 4.2, we have presented important architectural details of different state-of-the-art models. Among them, some models are based on different object detection models such as R-CNN, Fast R-CNN, Faster R-CNN, etc. Some models are based on semantic segmentation models such as FCN, U-Net, etc. SDS, DeepMask, SharpMask are based on proposal generation. InstanceFCN, FCIs, MaskLab calculate position-sensitive score maps for instance segmentation. PANet emphasized on feature propagation across the network. TensorMask used the sliding window approach for dense instance segmentation. So, architectural differences help different models to achieve success in various instance segmentation dataset. On the other hand, fine-tuning of hyper-parameters, data pre-processing methods, choice of the loss function and optimization function, etc are also played an important role in the success of a model. In this subsections, we are going to present some of those important features in a comparative manner.

Optimization Details of State-of-the-art Instance Segmentation Models:

The training and optimization process is very crucial for a model to become successful. Most of the state-of-the-art models used stochastic gradient descent(SGD) [180] as an optimization algorithm with different initialization of corresponding hyper parameters such as mini-batch size, learning rate, weight decay, momentum etc. Table 5 shows those hyper-parameters in a comparative way. As semantic segmentation, most of the instance segmentation models used momentum of 0.9 with different weight decay. Variation of choosing a learning rate is also not much.

Table 5: Optimization details of different state-of-the-art instance segmentation models

Name of the model	Optmization Algorithm	Mini Batch Size	Learning Rate	Momentum	Weight Decay
DeepMask [148]	SGD	32 images	0.001	0.9	0.00005
MNC [149]	SGD	1 images per GPU, total 8 GPUs are used	0.001 for 32k iteration, 0.0001 for next 8K iteration		
MultPath Network [153]	SGD	4 images,1 image per GPU, each with 64 object proposals	initially 0.001, after 160k iterations, it was reduced to 0.0001	-	-
SharpMask [154]	SGD		$1e^{-3}$		
InstanceFCN [21]	SGD	8 images each with 256 sampled windows, 1 image/GPU	0.001 for initial 32k iterations and 0.0001 for the next 8k.	0.9	0.0005
FCIs [159]	SGD	8 images/batch, 1 image per GPU	0.001 for the first 20k and 0.0001 for the next 10k iterations		
Mask R-CNN [20]	SGD	16 images/batch, 2 images per GPU	0.02 for first 160k iteration and 0.002 for next 120k iterations	0.0001	0.9
PANet [173]	SGD	16 images	0.02 for 120k iterations and 0.002 for 40k iterations	0.0001	0.9
TensorMask [175]	SGD	16 images, 2 images per GPU	0.02 with linear warm-up[180] of 1k iteration	0.9	0.0005

Different models have used different CNN based classification, Object detection and semantic segmentation model as their base network according to the availability. It is anopen choice to the researchers to choose a base model (may be pre-trained on some dataset)

Table 6: Base Model, data preprocessing technique and loss functions of different Stat-of-the-art models.

Name of the model	Base Network	Data pre-processing	Loss Funtion
SDS			
DeepMask	VGG-A pretrained on ImageNet dataset	• Randomly jitter 'canonical' positive examples for increasing the model's robustness • Applied translation shift, scale deformation and horizontal flip for data augmentation	Sum of binary logistic regression losses • One for each location of the segmentation network • Other for the objectness score
MNC	• VGG-16 • ResNet-101		Unified loss function • RPN loss for box regression/instance • Mask regression loss/instance • loss function for categorizing instances • Inference time per image is 1.4sec
MultPath Network	Fast R-CNN	Horizontal flip as data augmentation	Integral loss function: • Integral log loss function for classification • Bounding box function
SharpMask	DeepMask		Same as used in DeepMask
InstanceFCN	VGG-16, pretrained on ImageNet	Arbitray sized images are used for training with scale jittering following [17]	Logistic regression loss for predicting abjectness score and segment instances
FCIs	ResNet-101		
Mask R-CNN	Faster R-CNN based on ResNet		Multi-task loss: • log loss function for classification • L1 loss function for bounding box regression • Average binary cross entropy loss for mask prediction
MaskLab [170]	ResNet-101 based Faster R-CNN pretrained on ImageNet		
PANet	ResNet-50, ResNeXt-101[181] based Mask R-CNN and FPN		
TensorMask	ResNet-50, FPN	Scale jittering is used	Weighted sum of all task loss specially for mask, per-pixel binary classification loss is used. Focal loss is used to handle foreground background class imbalance.

Table 7: Some important features of different state-of-the-art instance segmentation models

Model	Important Features
SDS	• Used MCG to generate region proposals • Used segmentation data from SBD[131]
DeepMask	• The inference time in MS COCO is 1.6s per image • The inference time in PASCAL VOC 2007 is 1.2s per image
MNC	• End to End trainable • Convolutional feature sharing leads to reduction of test time of 360ms/image.
Multi-path Network	• Skip Connection for sharing feature among multiple levels • Foveal structure to capture multi-scale object • Integrated loss function for improving localization • DeepMask region proposal algorithm to generate region proposals • Training time 500ms/image
SharpMask	• Bottom-up/top-down approach • DeepMask used in bottom-up network to generate object proposal • Top-down network is stack of refinement model which aggregate features from corresponding layers from bottom-up path • Two stage training: One for bottom-up and another for top-down network
InstanceFCN	• A small set of score maps computed from different relative position of an image patch are assembled for predicting the segmentation mask • Applied 'hole algorithm[87]' in last three layers of VGGNet
FCIs	• End to end trainable FCN based model • Based on position sensitive inside and outside score map • Inference time 0.24 seconds/image • Six times faster than MNC
Mask R-CNN	• RoIAlign layers are used instead of RoIPool layer to preserve special location • Inference time was 200 ms per frame
MaskLab	• Used atrous convolution to extract denser feature map to control output resolution • End to end trainable model • To cover 360 degree of an instance 8 directions are used with 4 number of distance quantization bins for direction pooling
PANet	• FPN is used as Backbone network • Adaptive feature pooling layer is introduced
TensorMask	• Dense instance segmentation using sliding window approach • The model works on 4D tensor

according to their application domain. Most of the data preprocessing basically includes different data augmentation technique. Differences in loss function depend on the variation

of the model architecture as shown in table 6. Table 7 is showing some important features of different models.

Table 8: Comparison of different instance segmentation models as average precision according to IoU threshold

Model	Year	Used Dataset	mAP as IoU
SDS	2014	• PASCAL VOC 2011 −→ • PASCAL VOC 2012 −→	51.6% 52.6%
DeepMask	2015	• PASCAL VOC • MS COCO	Fast R-CNN using DeepMask outperforms original Fast R-CNN and achieved 66.9% accuracy on PASCAL VOC 2007 test dataset
MNC	2016	• PASCAL VOC 2012 −→ • MS COCO 2015 −→	63.5% on validation set 39.7% on *test − devset* (both mAP@).5 IoU threshold)
MultPath Network	2015	MS COCO 2015 −→	25.0%(AP), 45.4%(AP^{50}) and 24.5% (AP^{75}), all on test dataset. Superscripts of AP denotes IoU threshold
SharpMask +MPN [153]	2016	MS COCO 2015 −→	25.1%(AP), 45.8%(AP^{50}) and 24.8% (AP^{75}), all on test dataset. Superscripts of AP denotes IoU threshold
InstanceFCN+MNC	2016	PASCAL VOC 2012 validation dataset	61.5%(mAP@0.5) and 43.0%(mAP@0.7)
FCIs	2017	• Pascal voc 2012 −→ • MS COCO 2016−→	65.7%(mAP@0.5) and 52.1%(mAP@0.7) 59.9%(mAP@0.5)(ensemble)
Mask R-CNN	2017	MS COCO	60.0%(AP^{50}) and 39.4% (AP^{75}) , all on test dataset.Superscripts of AP denotes IoU threshold
MaskLab	2018	MS COCO (test-dev) −→	61.1%(mAP@0.5) and 40.4%(mAP@0.7)
PANet	2018	• MS COCO 2016 −→ • MS COCO 2017−→	65.1%(AP^{50}) and 45.7% (AP^{75}) 69.5%(AP^{50}) and 51.3% (AP^{75}) , (Mask AP).Superscripts of AP denotes IoU threshold
TensorMask	2019	MS COCO (test-dev)	37.3% (AP), 59.5%(AP^{50}) and 39.5% (AP^{75}).Superscripts of AP denotes IoU threshold

Comparative Performance of State-of-the-art Instance Segmentation Models:

Around 2014, concurrent with semantic segmentation task, CNN based instance segmentation models have also started gaining better accuracy in various data sets such as PASCAL VOC, MS COCO, etc. In table 8, we have shown the comparative performance of various state-of-the-art instance segmentation models on those datasets in chronological order.

5. Panoptic Segmentation

Panoptic segmentation (PS) [69, 182, 183, 184, 185, 186] is the combination of semantic segmentation and instance segmentation. This is a new research area these days. In this task, we need to associate all the pixels in the image with a semantic label for classification and also identify the instances of a particular class. The output of a panoptic segmentation model will contain two channels: one for pixel's label (semantic segmentation) and another for predicting each pixel instance (instance segmentation). In the following paragraphs we have given some examples of recently designed panoptic segmentation models.

Some popular state-of-the-art panoptic segmentation Models

Krillov et al. first proposed panoptic segmentation [69] by unifying semantic segmentation and instance segmentation to encompass both stuff and thing classes. The have used novel panoptic quality (PQ) metric to measure the performance of the segmentation. Their model produces simple but general output. The authors have used their model on Cityscapes, ADE20K [187]and Mapillary Vista [188] datasets and got better accuracy on segmentation.

OANet: Liu et al. proposed end-to-end Occlusion Aware Network(OANet) [183] for panoptic segmentation. The authors have used Feature Pyramid Network to extract featuremaps from the input image. Upon extracted feature they have applied two different branches:One for semantic segmentation and another for instance segmentation. Mask R-CNN isused for instance segmentation branch. Output of both branches are fed into novel SpatialRanking Module for final output of panoptic segmentation. They have applied their model on COCO panoptic segmentation benchmark and got promising results.

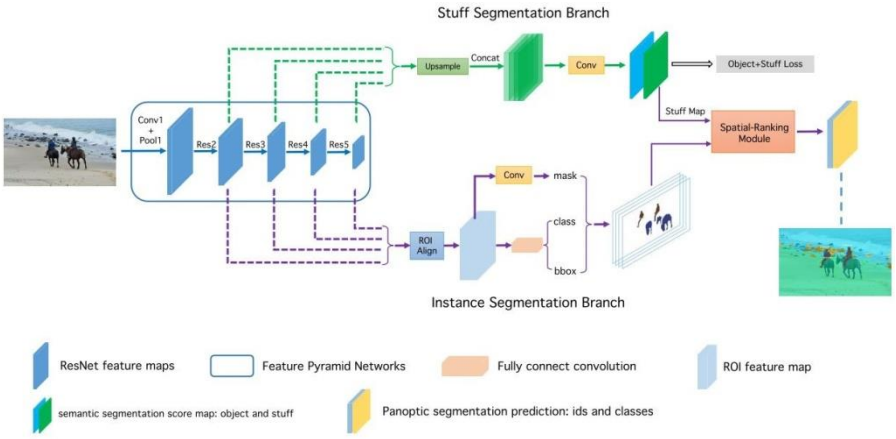

Figure 27: Architecture of Occlusion Aware Network(OANet). From [183]

UPSNet: Xiong et al. proposed a unified panoptic segmentation network (UPSNet) [185] to handle panoptic segmentation. The authors have used ResNet and FPN based Mask R-CNN as a backbone network to extract convolutional feature map. Those convolutional feature maps are fed into three sub-networks: for Semantic segmentation,for instance segmentation and for panoptic segmentation. Smantic segmentation sub-network consists of deformable convolutional network [189] for segmenting staff classes. Instance segmentation sub-network consists of three branch for bounding box regression, classification and sementation mask. All the outputs from these two subnetwork further goes to the panoptic segmentation sub-network for final panoptic segmentation. The authors have used teir model on Cityscapes and COCO datasets.

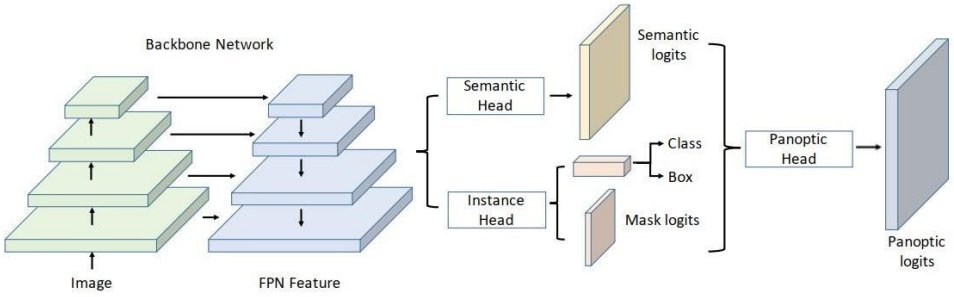

Figure 28: Architecture of unified panoptic segmentation network (UPSNet). From [185]

Multitask Network for Panoptic Segmentation: Andra Petrovai and Sergiu Nedevschi have proposed an end to end trainable multi-task network [186] for panoptic Segentation with the capability of object occlusion and scene depth ordering. As [185], the authors have used ResNet and FPN based back bone network for multi-scale feature extraction. the output of backbone network fed into 4 individual sub-networks for four tasks. First sub-network is for object detection and classification using Faster R-CNN, Second one is for instance segmentation using Mask R-CNN, third one is for semantic segmentation using pyramid pooling module as used in PSPNet [95] and fourth one is for panoptic segmentation. The authors have used their model on Cityscapes dataset and got 75.4% mIoU and 57.3% PQ.

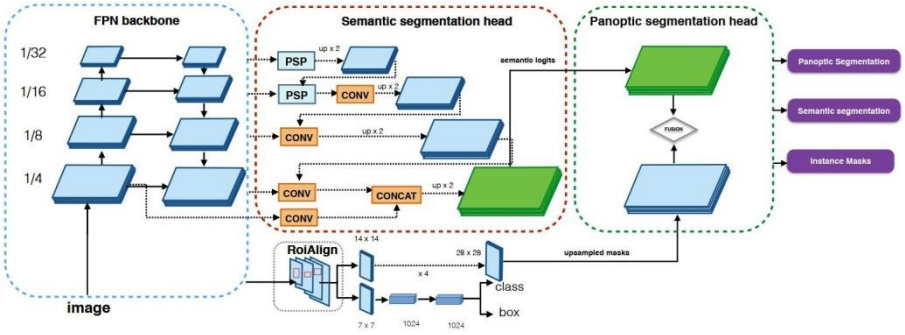

Figure 29: Architecture of Multitask Network for Panoptic Segmentation. From [186]

6. Conclusion

Image segmentation is a challenging work as it needs spatially variant features to preserve the context of a pixel for semantic labeling. Semantic segmentation categorizes each pixel with a semantic label whereas instance segmentation segments individual instances of objects contained in an image. The success of recent state-of-the-art models depends mostly on different network architecture. Except for that, various other aspects such as choice of the optimization algorithm, the value of hyper-parameters, data-preprocessing technique, choice of the loss function, etc are also responsible for becoming a successful model. In our article, we have presented the evolution of Convolutional Neural Networks based image segmentation models. We have categorically explored some state-of-the-art segmentation models along with their optimization details, and a comparison among the performance of those models on different datasets. Lastly, we have given a glimpse of recent state-of-the-art panoptic segmentation models. The application area of image segmentation is vast. According to the requirement of the application task, a suitable model can be applied using some domain-specific fine-tuning using dataset. Overall this article gives systematic ideas about present state-of-the-art image segmentation that will help researchers of this area for further proceedings.

Acknowledgment

The authors of the article are thankful to the editor and anonymous reviewers for mentioning some comments and suggestions on initial submission. For such comments and suggestions, the article got this present shape.

References

[1] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521 (7553) (2015) 436.

[2] I. Goodfellow, Y. Bengio, A. Courville, Deep learning, MIT press, 2016.

[3] J. Wu, Introduction to convolutional neural networks, National Key Lab for Novel Software Technology.Nanjing University. China 5 (2017) 23.

[4] K. O'Shea, R. Nash, An introduction to convolutional neural networks, arXiv preprint arXiv:1511.08458.

[5] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, D. De, Fundamental Concepts of Convolutional Neural Network, 2020, pp. 519–567. doi:10.1007/978-3-030-32644-9_36.

[6] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems 25, 2012, pp. 1097–1105.

[7] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, 2014, pp. 818–833.

[8] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, CoRR abs/1409.1556. arXiv:1409.1556.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[10] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[11] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324. doi:10.1109/5.726791.

[12] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, L. D. Jackel, Handwritten digit recognition with a back-propagation network, in: D. S. Touretzky (Ed.), Advances in Neural Information Processing Systems 2, Morgan-Kaufmann, 1990, pp. 396–404.

[13] S. Sabour, N. Frosst, G. E. Hinton, Dynamic routing between capsules, in: Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 3856–3866.
URL http://papers.nips.cc/paper/6975-dynamic-routing-between-capsules.pdf

[14] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.

[15] F. Sultana, A. Sufian, P. Dutta, Advancements in image classification using convolutional neural network, in: 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), 2018, pp. 122–129. doi:10.1109/ICRCICN.2018.8718718.

[16] R. Girshick, J. Donahue, T. Darrell, J. Malik, Region-based convolutional networks for accurate object detection and segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 38 (1) (2016) 142–158. doi:10.1109/TPAMI.2015.2437384.

[17] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 346–361.

[18] R. Girshick, Fast r-cnn, in: 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440–1448.

[19] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (6) (2017) 1137–1149.

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

[21] J. Dai, Y. Li, K. He, J. Sun, R-fcn: Object detection via region-based fully convolutional networks, in: Advances in Neural Information Processing Systems 29, Curran Associates, Inc., 2016, pp. 379–387.

[22] J. Redmon, S. K. Divvala, R. B. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 779–788. doi:10.1109/CVPR.2016.91.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, A. C. Berg, Ssd: Single shot multibox detector, in: ECCV, 2016.

[24] J. Redmon, A. Farhadi, Yolo9000: Better, faster, stronger, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017) 6517–6525.

[25] S. Zhang, L. Wen, X. Bian, Z. Lei, S. Z. Li, Single-shot refinement neural network for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4203–4212.

[26] T. Lin, P. Goyal, R. Girshick, K. He, P. Dollr, Focal loss for dense object detection, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2999–3007.

[27] C. Peng, T. Xiao, Z. Li, Y. Jiang, X. Zhang, K. Jia, G. Yu, J. Sun, Megdet: A large mini-batch object detector (2018) 6181–6189doi:10.1109/CVPR.2018.00647.

[28] Y. Wang, X. Luo, L. Ding, S. Fu, X. Wei, Detection based visual tracking with convolutional neural network, Knowledge-Based Systems 175 (2019) 62 – 71. doi:https://doi.org/10.1016/j.knosys. 2019.03.012.

[29] F. Sultana, A. Sufian, P. Dutta, A review of object detection models based on convolutional neural network, CoRR abs/1905.01614. arXiv:1905.01614.

[30] D. H. Hubel, T. N. Wiesel, Receptive fields and functional architecture of monkey striate cortex, Journal of Physiology (London) 195 (1968) 215–243.

[31] D. Marr, Vision: A Computational Investigation into the Human Representation and Processing of Visual Information, Henry Holt and Co., Inc., New York, NY, USA, 1982.

[32] K. Fukushima, Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position, Biological Cybernetics 36 (4) (1980) 193–202. doi:10. 1007/BF00344251.

[33] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel, Back-propagation applied to handwritten zip code recognition, Neural Comput. 1 (4) (1989) 541–551. doi:10.1162/neco.1989.1.4.541.

[34] Y. L. Cun, A theoretical framework for back-propagation.

[35] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, T. Chen, Recent advances in convolutional neural networks, Pattern Recognition 77 (2018) 354 – 377. doi:https://doi.org/10.1016/j.patcog.2017.10.013.

[36] B. De Brabandere, D. Neven, L. Van Gool, Semantic instance segmentation for autonomous driving, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2017.

[37] I. R. I. Haque, J. Neubert, Deep learning approaches to biomedical image segmentation, Informatics in Medicine Unlocked (2020) 100297doi:https://doi.org/10.1016/j.imu.2020.100297.

[38] J. Song, L. Xiao, M. Molaei, Z. Lian, Multi-layer boosting sparse convolutional model for generalized nuclear segmentation from histopathology images, Knowledge-Based Systems 176 (2019) 40 – 53. doi:https://doi.org/10.1016/j.knosys.2019.03.031.

[39] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Vol. 1, 2001, pp. I–I. doi:10.1109/CVPR.2001.990517.

[40] R. Vaillant, C. Monrocq, Y. L. Cun, Original approach for the localisation of objects in images, IEE Proceedings - Vision, Image and Signal Processing 141 (4) (1994) 245–250. doi:10.1049/ip-vis: 19941301.

[41] P. Dollr, Z. Tu, P. Perona, S. J. Belongie, Integral channel features., in: A. Cavallaro, S. Prince, D. C. Alexander (Eds.), BMVC, 2009, pp. 1–11.

[42] V. Sivakumar, V. Murugesh, A brief study of image segmentation using thresholding technique on a noisy image, in: International Conference on Information Communication and Embedded Systems (ICICES2014), 2014, pp. 1–6. doi:10.1109/ICICES.2014.7034056.

[43] O. Gómez, J. A. González, E. F. Morales, Image segmentation using automatic seeded region growing and instance-based learning, in: L. Rueda, D. Mery, J. Kittler (Eds.), Progress in Pattern Recognition, Image Analysis and Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 192–201.

[44] M. Mary Synthuja Jain Preetha, L. Padma Suresh, M. John Bosco, Image segmentation using seeded region growing, in: 2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET), 2012, pp. 576–583. doi:10.1109/ICCEET.2012.6203897.

[45] Y. Huang, C. Kuo, Image segmentation using edge detection and region distribution, in: 2010 3rd International Congress on Image and Signal Processing, Vol. 3, 2010, pp. 1410–1414. doi:10.1109/CISP.2010.5646352.

[46] S. S. Al-Amri, N. Kalyankar, S. Khamitkar, Image segmentation by using edge detection, International journal on computer science and engineering 2 (3) (2010) 804–807.

[47] W.-Y. Ma, B. Manjunath, Edge flow: a framework of boundary detection and image segmentation, in: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997, pp. 744–749.

[48] X. Zheng, Q. Lei, R. Yao, Y. Gong, Q. Yin, Image segmentation based on adaptive k-means algorithm, EURASIP Journal on Image and Video Processing 2018 (1) (2018) 68. doi:10.1186/ s13640-018-0309-3.

[49] M. Celenk, A color clustering technique for image segmentation, Computer Vision, Graphics, and Image Processing 52 (2) (1990) 145 – 170. doi:https://doi.org/10.1016/0734-189X(90)90052-W.

[50] A. Kavitha, C. Chellamuthu, Implementation of gray-level clustering algorithm for image segmentation, Procedia Computer Science 2 (2010) 314 – 320, proceedings of the International Conference and Exhibition on Biometrics Technology. doi:https://doi.org/10.1016/j.procs.2010.11.041.

[51] M. A. Ali, L. S. Dooley, G. C. Karmakar, Object based image segmentation using fuzzy clustering, in: 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, Vol. 2, 2006, pp. II–II. doi:10.1109/ICASSP.2006.1660290.

[52] J. Galbiati, H. Allende, C. Becerra, Dynamic image segmentation method using hierarchical clustering, in: E. Bayro-Corrochano, J.-O. Eklundh (Eds.), Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, 2009, pp. 177–184.

[53] L. Franek, D. D. Abdala, S. Vega-Pons, X. Jiang, Image segmentation fusion using general ensemble clustering methods, in: R. Kimmel, R. Klette, A. Sugimoto (Eds.), Computer Vision – ACCV 2010, 2011, pp. 373–384.

[54] A. Alush, J. Goldberger, Break and conquer: Efficient correlation clustering for image segmentation, in: E. Hancock, M. Pelillo (Eds.), Similarity-Based Pattern Recognition, 2013, pp. 134–147.

[55] J. Yarkony, A. Ihler, C. C. Fowlkes, Fast planar correlation clustering for image segmentation, in: A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (Eds.), Computer Vision – ECCV 2012, 2012, pp. 568–581.

[56] S.-F. Chang, Segmentation using superpixels: A bipartite graph partitioning approach, in: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), CVPR '12, 2012, pp. 789–796.

[57] X. Xie, G. Xie, X. Xu, L. Cui, J. Ren, Automatic image segmentation with superpixels and image-level labels, IEEE Access 7 (2019) 10999–11009. doi:10.1109/ACCESS.2019.2891941.

[58] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1, 2005, pp. 886–893 vol. 1. doi:10.1109/CVPR.2005.177.

[59] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27.

[60] S. Tuermer, F. Kurz, P. Reinartz, U. Stilla, Airborne vehicle detection in dense urban areas using hog features and disparity maps, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 6 (6) (2013) 2327–2337.

[61] S. Gupta, R. Girshick, P. Arbeláez, J. Malik, Learning rich features from rgb-d images for object detection and segmentation (2014) 345–360.

[62] W. Burger, M. J. Burge, Scale-Invariant Feature Transform (SIFT), Springer London, London, 2016, pp. 609–664. doi:10.1007/978-1-4471-6684-9_25.

[63] Akira Suga, Keita Fukuda, Tetsuya Takiguchi, Yasuo Ariki, Object recognition and segmentation using sift and graph cuts, in: 2008 19th International Conference on Pattern Recognition, 2008, pp. 1–4. doi:10.1109/ICPR.2008.4761400.

[64] S. Srinivas, R. K. Sarvadevabhatla, K. R. Mopuri, N. Prabhu, S. S. Kruthiventi, R. V. Babu, A taxonomy of deep convolutional neural nets for computer vision, Frontiers in Robotics and AI 2 (2016) 36.

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

[66] N. R. Pal, S. K. Pal, A review on image segmentation techniques, Pattern Recognition 26 (9) (1993) 1277 – 1294. doi:https://doi.org/10.1016/0031-3203(93)90135-J.

[67] N. M. Zaitoun, M. J. Aqel, Survey on image segmentation techniques, Procedia Computer Science 65 (2015) 797 – 806, international Conference on Communications, management, and Information technology (ICCMIT'2015). doi:https://doi.org/10.1016/j.procs.2015.09.027.

[68] D. Zhang, Y. Song, D. Liu, H. Jia, S. Liu, Y. Xia, H. Huang, W. Cai, Panoptic segmentation with an end-to-end cell r-cnn for pathology image analysis (2018) 237–244.

[69] A. Kirillov, K. He, R. Girshick, C. Rother, P. Dollár, Panoptic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2019, pp. 9404–9413.

[70] S. Caelles, K. . Maninis, J. Pont-Tuset, L. Leal-Taix, D. Cremers, L. Van Gool, One-shot video object segmentation, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5320–5329.

[71] S. W. Oh, J. Lee, K. Sunkavalli, S. J. Kim, Fast video object segmentation by reference-guided mask propagation, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7376–7385.

[72] J. Miao, Y. Wei, Y. Yang, Memory aggregation networks for efficient interactive video object segmentation, arXiv preprint arXiv:2003.13246.

[73] Z. Yang, Y. Wei, Y. Yang, Collaborative video object segmentation by foreground-background integration, arXiv preprint arXiv:2003.08333.

[74] Y. Wei, X. Liang, Y. Chen, Z. Jie, Y. Xiao, Y. Zhao, S. Yan, Learning to segment with image-level annotations, Pattern Recognition 59 (2016) 234 – 244, compositional Models and Structured Learning for Visual Recognition. doi:https://doi.org/10.1016/j.patcog.2016.01.015.

[75] C. Farabet, C. Couprie, L. Najman, Y. LeCun, Learning hierarchical features for scene labeling, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (8) (2013) 1915–1929. doi: 10.1109/TPAMI.2012.231.

[76] C. Liu, J. Yuen, A. Torralba, Sift flow: Dense correspondence across scenes and its applications, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (5) (2011) 978–994. doi:10.1109/ TPAMI.2010.147.

[77] J. Tighe, S. Lazebnik, Superparsing: Scalable nonparametric image parsing with superpixels, in: K. Daniilidis, P. Maragos, N. Paragios (Eds.), Computer Vision – ECCV 2010, 2010, pp. 352–365.

[78] S. Gould, R. Fulton, D. Koller, Decomposing a scene into geometric and semantically consistent regions, in: 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 1–8. doi: 10.1109/ICCV.2009.5459211.

[79] R. B. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, 2014 IEEE Conference on Computer Vision and Pattern Recognition (2014) 580–587.

[80] J. Uijlings, K. van de Sande, T. Gevers, A. Smeulders, Selective search for object recognition, International Journal of Computer Visiondoi:10.1007/s11263-013-0620-5.

[81] M. Everingham, A. Zisserman, C. K. I. Williams, L. V. Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, G. Dork, S. Duffner, J. Eichhorn, J. D. R. Farquhar, M. Fritz, C. Garcia, T. Griffiths, F. Jurie, D. Keysers, M. Koskela, J. Laaksonen, D. Larlus, B. Leibe, H. Meng, H. Ney, B. Schiele, C. Schmid, E. Seemann, J. Shawe-taylor, A. Storkey, O. Szedmak, B. Triggs, I. Ulusoy, V. Viitaniemi, J. Zhang, The pascal visual object classes challenge 2007 (voc2007) results.

[82] J. Carreira, C. Sminchisescu, Cpmc: Automatic object segmentation using constrained parametric min-cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (7) (2012) 1312–1328. doi:10.1109/TPAMI.2011.231.

[83] E. Shelhamer, J. Long, T. Darrell, Fully convolutional networks for semantic segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (4) (2017) 640–651. doi:10.1109/TPAMI.2016.2572683.

[84] M. Mostajabi, P. Yadollahpour, G. Shakhnarovich, Feedforward semantic segmentation with zoom-out features. (2015) 3376–3385.
URL http://dblp.uni-trier.de/db/conf/cvpr/cvpr2015.html#MostajabiYS15

[85] C. Szegedy, S. E. Reed, D. Erhan, D. Anguelov, Scalable, high-quality object detection, CoRR abs/1412.1441. arXiv:1412.1441.

[86] Y. Hu, A. Soltoggio, R. Lock, S. Carter, A fully convolutional two-stream fusion network for interactive image segmentation, Neural Networks 109 (2019) 31 – 42. doi:https://doi.org/10.1016/j.neunet.2018.10.009.

[87] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, Semantic image segmentation with deep convolutional nets and fully connected crfs, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.

[88] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, CoRR abs/1606.00915. arXiv:1606.00915.

[89] L. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, CoRR abs/1706.05587. arXiv:1706.05587.

[90] H. Noh, S. Hong, B. Han, Learning deconvolution network for semantic segmentation (2015) 1520–1528.

[91] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: N. Navab, J. Hornegger, W. M. Wells, A. F. Frangi (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, 2015, pp. 234–241.

[92] N. Ibtehaz, M. S. Rahman, Multiresunet : Rethinking the u-net architecture for multimodal biomedical image segmentation, Neural Networks 121 (2020) 74 – 87. doi:https://doi.org/10.1016/j.neunet.2019.08.025.

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

[94] W. Liu, A. Rabinovich, A. C. Berg, Parsenet: Looking wider to see better, CoRR abs/1506.04579. arXiv:1506.04579.

[95] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, CoRR abs/1612.01105. arXiv:1612.01105.

[96] H. Li, P. Xiong, J. An, L. Wang, Pyramid attention network for semantic segmentation, CoRR abs/1805.10180. arXiv:1805.10180.

[97] C. Peng, X. Zhang, G. Yu, G. Luo, J. Sun, Large kernel matters–improve semantic segmentation by global convolutional network (2017) 4353–4361.

[98] S. Brahimi, N. Ben Aoun, A. Benoit, P. Lambert, C. Ben Amar, Semantic segmentation using reinforced fully convolutional densenet with multiscale kernel, Multimedia Tools and Applications 78 (15) (2019) 22077–22098. doi:10.1007/s11042-019-7430-x.

[99] T. Takikawa, D. Acuna, V. Jampani, S. Fidler, Gated-scnn: Gated shape cnns for semantic segmentation (2019) 5229–5238.

[100] R. Li, W. Liu, L. Yang, S. Sun, W. Hu, F. Zhang, W. Li, Deepunet: A deep fully convolutional network for pixel-level sea-land segmentation, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing PP. doi:10.1109/JSTARS.2018.2833382.

[101] A. Paszke, A. Chaurasia, S. Kim, E. Culurciello, Enet: A deep neural network architecture for real-time semantic segmentation, CoRR abs/1606.02147. arXiv:1606.02147.

[102] H. Zhao, X. Qi, X. Shen, J. Shi, J. Jia, ICNet for real-time semantic segmentation on high-resolution images, in: ECCV, 2018.

[103] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation (2018) 801–818.

[104] C. Liu, L.-C. Chen, F. Schroff, H. Adam, W. Hua, A. L. Yuille, L. Fei-Fei, Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 82–92.

[105] K. Sun, Y. Zhao, B. Jiang, T. Cheng, B. Xiao, D. Liu, Y. Mu, X. Wang, W. Liu, J. Wang, High-resolution representations for labeling pixels and regions, CoRR abs/1904.04514. arXiv:1904.04514.

[106] Y. Zhang, M. Ye, Y. Gan, W. Zhang, Knowledge based domain adaptation for semantic segmentation, Knowledge-Based Systems (2019) 105444doi:https://doi.org/10.1016/j.knosys.2019.105444.

[107] A. K. Y. W. K. He, R. Girshick, Pointrend: Image segmentation as rendering, networks (CNNs) 27 26.

[108] P. Luc, C. Couprie, S. Chintala, J. Verbeek, Semantic segmentation using adversarial networks, arXiv preprint arXiv:1611.08408.

[109] Y. Luo, L. Zheng, T. Guan, J. Yu, Y. Yang, Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2018) 2502–2511.

[110] Y. Luo, P. Liu, T. Guan, J. Yu, Y. Yang, Significance-aware information bottleneck for domain adaptive semantic segmentation, 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019) 6777–6786.

[111] Y. Luo, Z. Zheng, L. Zheng, T. Guan, J. Yu, Y. Yang, Macro-micro adversarial network for human parsing, in: Computer Vision – ECCV 2018, Springer International Publishing, Cham, 2018, pp. 424–440.

[112] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7794–7803.

[113] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, W. Liu, Ccnet: Criss-cross attention for semantic segmentation, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 603–612.

[114] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. C. Loy, D. Lin, J. Jia, Psanet: Point-wise spatial attention network for scene parsing, in: Computer Vision – ECCV 2018, Springer International Publishing, Cham, 2018, pp. 270–286.

[115] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 3141–3149.

[116] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, International Journal of Computer Vision (IJCV) 115 (3) (2015) 211–252. doi:10.1007/s11263-015-0816-y.

[117] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, International Journal of Computer Vision 88 (2) (2010) 303–338. doi:10. 1007/s11263-009-0275-4.

[118] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer-Verlag, Berlin, Heidelberg, 2006.

[119] N. Silberman, D. Hoiem, P. Kohli, R. Fergus, Indoor segmentation and support inference from rgbd images, in: A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (Eds.), Computer Vision – ECCV 2012, 2012, pp. 746–760.

[120] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, in: 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings, 2016.

[121] S. Mallat, A wavelet tour of signal processing (2. ed.)., Academic Press, 1999.

[122] M. D. Zeiler, G. W. Taylor, R. Fergus, Adaptive deconvolutional networks for mid and high level feature learning, in: 2011 International Conference on Computer Vision, 2011, pp. 2018–2025. doi: 10.1109/ICCV.2011.6126474.

[123] N. Bjorck, C. P. Gomes, B. Selman, K. Q. Weinberger, Understanding batch normalization (2018) 7694–7705.

[124] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, CoRR abs/1502.03167. arXiv:1502.03167.

[125] V. Badrinarayanan, B. Mishra, R. Cipolla, Understanding symmetries in deep networks, CoRR abs/1511.01029. arXiv:1511.01029.

[126] A. F. Agarap, Deep learning using rectified linear units (relu), cite arxiv:1803.08375Comment: 7 pages, 11 figures, 9 tables (2018).

[127] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, Y. Bengio, The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation (2017) 11–19.

[128] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, A. Yuille, The role of context for object detection and semantic segmentation in the wild, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[129] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, A. Agrawal, Context encoding for semantic segmentation, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7151–7160.

[130] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: Proceedings of COMP-STAT'2010, Physica-Verlag HD, Heidelberg, 2010, pp. 177–186.

[131] B. Hariharan, P. Arbelez, L. Bourdev, S. Maji, J. Malik, Semantic contours from inverse detectors, in: 2011 International Conference on Computer Vision, 2011, pp. 991–998. doi:10.1109/ICCV.2011. 6126343.

[132] S. Zagoruyko, N. Komodakis, Wide residual networks, CoRR abs/1605.07146. arXiv:1605.07146.

[133] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification (2015) 1026–1034.

[134] Y. N. Dauphin, H. de Vries, J. Chung, Y. Bengio, Rmsprop and equilibrated adaptive learning rates for non-convex optimization., CoRR abs/1502.04390.

[135] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, IJCV 115 (3) (2015) 211–252. doi:10.1007/s11263-015-0816-y.

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

[137] S. Song, S. P. Lichtenberg, J. Xiao, Sun rgb-d: A rgb-d scene understanding benchmark suite., in: CVPR, 2015, pp. 567–576.

[138] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding (2016) 3213–3223.

[139] S. Hussain Raza, M. Grundmann, I. Essa, Geometric context from videos (2013) 3081–3088.

[140] Z. Wu, C. Shen, A. v. d. Hengel, Bridging category-level and instance-level semantic image segmentation, arXiv preprint arXiv:1605.06885.

[141] B. Hariharan, P. Arbeláez, R. Girshick, J. Malik, Simultaneous detection and segmentation (2014) 297–312.

[142] B. Hariharan, P. Arbeláez, R. Girshick, J. Malik, Hypercolumns for object segmentation and fine-grained localization (2015) 447–456.

[143] D. Erhan, C. Szegedy, A. Toshev, D. Anguelov, Scalable object detection using deep neural networks (2014) 2147–2154.

[144] C. Szegedy, S. E. Reed, D. Erhan, D. Anguelov, Scalable, high-quality object detection, CoRR abs/1412.1441. arXiv:1412.1441.

[145] W. Kuo, B. Hariharan, J. Malik, Deepbox: Learning objectness with convolutional networks (2015) 2479–2487.

[146] C. L. Zitnick, P. Dollár, Edge boxes: Locating object proposals from edges, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, 2014, pp. 391–405.

[147] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, J. Malik, Multiscale combinatorial grouping, in: Computer Vision and Pattern Recognition, 2014.

[148] P. O. Pinheiro, R. Collobert, P. Dollár, Learning to segment object candidates (2015) 1990–1998.

[149] J. Dai, K. He, J. Sun, Instance-aware semantic segmentation via multi-task network cascades (2016) 3150–3158.

[150] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation (2015) 3431–3440.

[151] S. Xie, Z. Tu, Holistically-nested edge detection (2015) 1395–1403.

[152] P. Sermanet, b. p. y. title=Pedestrian detection with unsupervised multi-stage feature learning, author=Sermanet, Pierre and Kavukcuoglu, Koray and Chintala, Soumith and LeCun, Yann.

[153] S. Zagoruyko, A. Lerer, T. Lin, P. H. O. Pinheiro, S. Gross, S. Chintala, P. Dollár, A multipath network for object detection, CoRR abs/1604.02135. arXiv:1604.02135.

[154] P. O. Pedro H. O. Pinheiro andauthor="Pinheiro, T.-Y. Lin, R. Collobert, e. B. Dollár, Piotr", J. Matas, N. Sebe, t. t. R. O. S. b. V. E. . y. p. I. P. a. p.-. i.-.-.-.-. Welling, Max".

[155] A. Torralba, Contextual priming for object detection, International Journal of Computer Vision 53 (2) (2003) 169–191. doi:10.1023/A:1023052124951.

[156] P. Sermanet, K. Kavukcuoglu, S. Chintala, Y. LeCun, Pedestrian detection with unsupervised multi-stage feature learning (2013) 3626–3633.

[157] S. Bell, C. Lawrence Zitnick, K. Bala, R. Girshick, Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks (2016) 2874–2883.

[158] D. Hoiem, Y. Chodpathumwan, Q. Dai, Diagnosing error in object detectors, in: A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (Eds.), Computer Vision – ECCV 2012, 2012, pp. 340–353.

[159] Y. Li, H. Qi, J. Dai, X. Ji, Y. Wei, Fully convolutional instance-aware semantic segmentation (2017) 2359–2367.

[160] J. Dai, K. He, J. Sun, Convolutional feature masking for joint object and stuff segmentation (2015) 3992–4000.

[161] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, S. Yan, Proposal-free network for instance-level object segmentation, CoRR abs/1509.02636. arXiv:1509.02636.

[162] S. Liu, X. Q, J. shi, H. Zhang, J. Jia, Multi-scale patch arregation(mpa) for simultaneous detection and segmentation.

[163] A. Arnab, P. H. Torr, Pixelwise instance segmentation with a dynamically instantiated network (2017) 441–450.

[164] M. Bai, R. Urtasun, Deep watershed transform for instance segmentation (2017) 5221–5229.

[165] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, C. Rother, Instancecut: from edges to instances with multicut (2017) 5008–5017.

[166] S. Liu, J. Jia, S. Fidler, R. Urtasun, Sgn: Sequential grouping networks for instance segmentation, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 3516–3524. doi: 10.1109/ICCV.2017.378.

[167] Z. Huang, L. Huang, Y. Gong, C. Huang, X. Wang, Mask scoring r-cnn, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 6402–6411.

[168] A. K. Y. W. K. He, R. Girshick, Pointrend: Image segmentation as rendering.

[169] J. Uhrig, M. Cordts, U. Franke, T. Brox, Pixel-level encoding and depth layering for instance-level semantic labeling, in: B. Rosenhahn, B. Andres (Eds.), Pattern Recognition, 2016, pp. 14–25.

[170] L.-C. Chen, A. Hermans, G. Papandreou, F. Schroff, P. Wang, H. Adam, Masklab: Instance segmentation by refining object detection with semantic and direction features (2018) 4013–4022.

[171] T. Kong, A. Yao, Y. Chen, F. Sun, Hypernet: Towards accurate region proposal generation and joint object detection (2016) 845–853.

[172] S. Zagoruyko, A. Lerer, T. Lin, P. H. O. Pinheiro, S. Gross, S. Chintala, P. Dollár, A multipath network for object detection, CoRR abs/1604.02135. arXiv:1604.02135.

[173] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation (2018) 8759–8768.

[174] A. Newell, Z. Huang, J. Deng, Associative embedding: End-to-end learning for joint detection and grouping, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 2277–2287.

[175] X. Chen, R. Girshick, K. He, P. Dollár, Tensormask: A foundation for dense object segmentation (2019) 2061–2069.

[176] S. Gidaris, N. Komodakis, Object detection via a multi-region and semantic segmentation-aware cnn model (2015) 1134–1142.

[177] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, 2014, pp. 740–755.

[178] C. Barnes, E. Shechtman, A. Finkelstein, D. B. Goldman, Patchmatch: A randomized correspondence algorithm for structural image editing, in: ACM SIGGRAPH 2009 Papers, SIGGRAPH '09, 2009, pp. 24:1–24:11. doi:10.1145/1576246.1531330.

[179] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection (2017) 2117–2125.

[180] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks (2017) 1492–1500.

[181] S. Xie, R. B. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, CoRR abs/1611.05431. arXiv:1611.05431.

[182] D. de Geus, P. Meletis, G. Dubbelman, Single network panoptic segmentation for street scene understanding, CoRR abs/1902.02678. arXiv:1902.02678.

[183] H. Liu, C. Peng, C. Yu, J. Wang, X. Liu, G. Yu, W. Jiang, An end-to-end network for panoptic segmentation (2019) 6172–6181.

[184] A. Kirillov, R. Girshick, K. He, P. Dollár, Panoptic feature pyramid networks (2019) 6399–6408.

[185] Y. Xiong, R. Liao, H. Zhao, R. Hu, M. Bai, E. Yumer, R. Urtasun, Upsnet: A unified panoptic segmentation network (2019) 8818–8826.

[186] A. Petrovai, S. Nedevschi, Multi-task network for panoptic segmentation in automated driving, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019, pp. 2394–2401. doi:10.1109/ITSC.2019.8917422.

[187] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ade20k dataset, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5122–5130.

[188] G. Neuhold, T. Ollmann, S. R. Bul, P. Kontschieder, The mapillary vistas dataset for semantic understanding of street scenes, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 5000–5009.

[189] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, Deformable convolutional networks, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 764–773.