Exploring Semantic Segmentation: Dataset, Techniques, and Challenges

Mehadi Hasan , Mohammad Shorif Uddin

Dept. of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh
E-mail: mehadihasannayan1998@gmail.com, shorifuddin@juniv.edu

Abstract

Semantic segmentation is a cornerstone of computer vision, playing a crucial role in scene understanding and object recognition. The main goal of semantic segmentation is to give each pixel in an image its own label breaking it up into meaningful areas. This method helps machines understand visual data better allowing them to look at and make sense of images at a higher level. This work provides an extensive investigation of deep learning methods for semantic segmentation, with applications in multiple domains. The introduction of key terms and concepts is followed by a thorough review of contemporary methods, highlighting their contributions and importance to the subject. Important datasets are also examined, pointing researchers in the direction of the best resources for their particular needs. Furthermore, the measures used for evaluating segmentation performance are examined. In order to provide light on how this important computer vision task is changing, the paper's conclusion examines present issues as well as potential future approaches.

Keywords

Semantic Segmentation, Convolutional neural network, Weakly supervised method, Deep Learning, computer vision.

1. Introduction

Semantic segmentation is currently one of the main issues in computer vision, whether it be used to static 2D images, video, or even 3D or volumetric data. Semantic segmentation is one of the high-level tasks that leads to comprehensive scene knowledge, when seen in the broadest context [1]. Accurate scene interpretation is desperately needed, especially with the growing number of intelligent applications (such as mobile robots). Semantic segmentation has so attracted a great deal of attention in recent years as a necessary step towards this goal [2]. Applying deep learning-based Convolutional Neural Networks (CNN) approaches has led to notable progress in the field of semantic segmentation [3]. The fact that a growing number of applications rely on deriving knowledge from imagery highlights the significance of scene understanding as a fundamental computer vision problem [4]. Among those uses are, to mention a few, human-machine interaction [5], autonomous driving [6-8], computational photography [9], picture search engines [10], and augmented reality. Two typical concerns are: how to create effective feature representations to distinguish objects of different classes and how to use contextual information to guarantee pixel

label consistency in order to achieve high-quality semantic segmentation [2]. Using hand-engineered features, like Scale Invariant Feature Transform (SIFT) [10] and Histograms of Oriented Gradient (HOG) [11], is advantageous for the majority of early approaches [12-13] when answering the first question. Utilizing learned features in computer vision tasks, including picture classification [14-15], has been very successful in the last few years thanks to the emergence of deep learning [16-17]. Consequently, the learnt features have received a lot of attention lately from the semantic segmentation field [18-21], where they are typically associated with Convolutional Neural Network (CNN or Convent) [22]. Using contextual models like Conditional Random Field (CRF) [23-25] and Markov Random Field (MRF) [26] is the most popular approach for the second problem, regardless of the feature employed.

This paper's primary goal is to present a thorough overview of semantic segmentation techniques, with an emphasis on examining the issues that are frequently raised and the associated solutions used. These days, semantic segmentation is a huge field with close ties to other computer vision tasks. The entire field cannot be covered by this review. There are already various evaluations on the state of the art in picture segmentation research, as well as semantic segmentation datasets and techniques [1-2].

The key contributions of our work are as follows:
- An extensive and well-structured analysis of the most important deep learning techniques for semantic segmentation, together with an overview of their history and contributions.
- We offer an overview of available datasets that could be helpful for deep learningbased semantic segmentation projects.
- A comprehensive analysis of performance that collects numerical measurements for things like memory, execution time, and precision.
- Draw attention to the issues that need to be resolved by upcoming researchers and the future scope in semantic segmentation.

The subsequent sections of this article are structured as follows to clarify the overall flow of this paper: Section 2 provides the basic concepts of semantic segmentation. Section 3 describes various deep learning based architecture for semantic segmentation. Section 4 provides the details of available datasets for semantic segmentation. Section 5 presents the evaluation metrics along with the performance analyses of the important semantic segmentation techniques. Section 6 highlights the challenges and future scope for the researcher in image segmentation. Finally, Section 7 draws the conclusion of this study.

2. Background and Preliminaries

2.1. Semantic Segmentation

Semantic segmentation is an essential computer vision approach that improves the efficiency with which machines evaluate and comprehend visual data. Comparing semantic

segmentation to traditional image recognition techniques—which typically give an image a single label reveals a significant improvement. Going one step further, semantic segmentation assigns a class or category to every pixel in an image according to what it symbolizes. Semantic segmentation achieves this by determining the semantic meaning of each pixel, resulting in a rich and comprehensive segmentation map that provides a finer and more accurate knowledge of the image. The foundation of many computer vision applications, including autonomous vehicles, medical imaging, and scene interpretation, is semantic segmentation.

2.2. Labels or Classes

The terms "labels" or "classes" in the context of semantic segmentation refer to the predetermined categories or semantic identities that are given to every pixel in an image at the time of segmentation. Within the visual input, these labels denote the various regions, objects, or structures that the model has been trained to identify and distinguish. A unique label designating the semantic meaning or category to which each pixel in the segmented image belongs is assigned to it.

In a street scene, for instance, common labels or classes could be "car", "pedestrian", "road", "building" and so on. In order to provide a thorough and in-depth comprehension of the scene, semantic segmentation aims to precisely identify and outline each pixel in the image in accordance with these predetermined classes.

2.3. Ground Truth

"Ground truth" in semantic segmentation refers to the manually annotated and labeled data that is the final source of reference for accurately segmenting pixels in an image. A semantic segmentation model's performance can be measured during both the training and testing stages using ground truth as a reference.

Human annotators carefully assign the appropriate semantic category or class to every pixel in an image in order to create ground truth. Annotators may designate pixels in a street scene, for instance, as belonging to the categories "car", "pedestrian", "road" or "building". The resultant annotated image serves as the ground truth for that particular image and is frequently referred to as a segmentation map.

In training, a dataset containing input images and the related ground truth annotations is used to teach a semantic segmentation model. The differences between the model's predictions and the labels that correspond to the ground truth are used to modify the model's internal parameters. The model is able to capture the complex features and patterns required for precise pixel-wise segmentation because of this iterative learning process.

During the testing or evaluation step, fresh, unobserved images are fed into the trained model, and its predictions are measured against the ground truth to gauge how well it performs. Common evaluation criteria that measure how well the model matches the real world include pixel accuracy and intersection over union (IoU).

2.4. Transfer Learning

It is frequently impractical to train a deep neural network from scratch for a variety of reasons, including the need for a large enough dataset—which is typically unavailable—and the possibility that it will take too long for the trials to be worthwhile. It is frequently beneficial to begin with pre-trained weights rather than randomly started ones, even in cases when a sufficiently big dataset is available and convergence happens quickly [27] [28]. One of the main transfer learning scenarios is fine-tuning the weights of a pre-trained network by extending the training phase.

Applying the transfer learning approach is not always simple, though. Using a pretrained network requires adherence to certain architectural requirements. Transfer learning is made possible by the fact that it is normal practice to reuse pre-existing network designs (or components) as opposed to creating entirely new ones. However, there is a small difference in the training procedure when fine-tuning as opposed to starting from fresh. Since the lower layers of the network typically contain more generic features, it is important to carefully select which layers to fine-tune. You should also choose an appropriate policy for the learning rate, which is typically smaller because the pre-trained weights are expected to be relatively good and do not require significant modification.

2.5. Data Preprocessing and Augmentation

An essential part of the training pipeline for semantic segmentation models is data preprocessing and augmentation. Improving the model's ability to generalize across many scenarios and variances in real-world data requires the application of these strategies.

To promote numerical stability during training and lessen the effect of changing illumination conditions, normalization is used to pixel values in data preprocessing to bring them to a standardized scale. Model training and inference are made more efficient by resizing, which guarantees consistency in input sizes. By cropping, extraneous computation is minimized by centering the model on pertinent regions of interest. In order to prevent the model from favoring frequently occurring classes and to improve generalization across all classes, class balancing corrects imbalances in the distribution of classes.

To improve model resilience, data augmentation entails adding changes to the training dataset. Rotations offer a variety of object orientations, and rotating an image horizontally or vertically produces mirrored replicas of the original image, increasing the dataset and lowering the chance of overfitting. By simulating varied distances, zooming and scaling aid in the model's ability to adjust to objects of various sizes and distances. Color jittering adds color fluctuations, which improves the model's adaptability to different lighting scenarios. Elastic deformation makes the model resistant to deformable objects by applying non-rigid deformations to images that resemble real-world distortions.

Data augmentation is a widely used method that has been shown to help with deep architectures in particular and machine learning models in general. It can either accelerate

convergence or function as a regularizer to prevent overfitting and improve generalization capabilities [29].

2.6. Super-pixels

A collection of pixels with comparable features or attributes is referred to as a superpixel in the context of semantic segmentation. Superpixels are produced via a technique called superpixel segmentation, in which an image is divided into uniform, perceptually significant sections, each of which is represented by a superpixel.

The main goal of employing superpixels in semantic segmentation is to preserve significant information and structures in a picture while lowering the computing burden of processing individual pixels. Superpixels offer a more condensed representation of the image as opposed to working on each pixel separately, enabling more effective and insightful analysis.

Superpixels are generally produced by algorithms that cluster pixels according to lowlevel characteristics like color, texture, or other comparable characteristics. SLIC (Simple Linear Iterative Clustering), Felzenszwalb, and QuickShift are popular superpixel segmentation techniques. The objective of these algorithms is to create coherent and significant superpixels by clustering pixels that have comparable perceptions and spatial connections.

2.7. General semantic segmentation method

The picture shows a flowchart that describes how to find anti-counterfeit characteristics in an object, beginning with taking a picture of the object to be examined. Finding possible areas of the image that could include anti-counterfeit elements is the next phase. These areas are then highlighted for more examination. The algorithm then takes these candidate regions and extracts certain properties that are suggestive of anti-counterfeit measures from them. Two instances of feature extraction are presented here, where abstract forms are highlighted in various colors. Following their extraction, the features are categorized to see if they match any known anticounterfeit features. The decisionmaking process is denoted by the text "Anti-counterfeit feature?" In order to ensure that the discovered characteristics are legitimate anti-counterfeit measures, the last stage entails additional processing and validation. The final validated features are represented by three similar abstract shapes in blue.
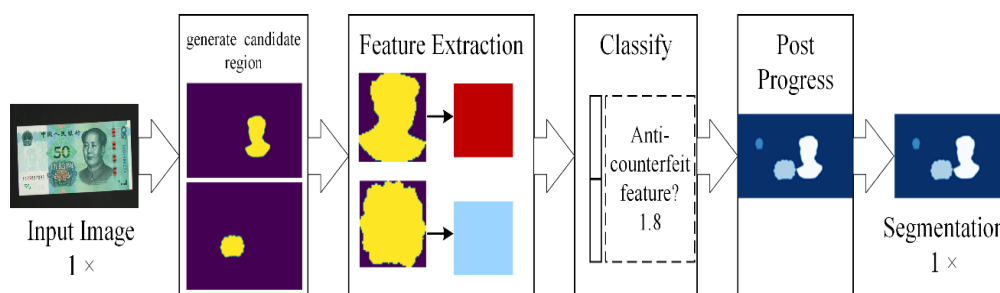

Figure 1: Schematic diagram for the general semantic segmentation approach.[68]

2.8. Semantic segmentation under the complex background

The image is a comprehensive flowchart that uses a generic example to explain the steps involved in component and object segmentation in image analysis. An input image is used as the starting point of the process to facilitate segmentation. Individual components within the image are located and isolated by component segmentation, and these components are subsequently highlighted in response maps through color coding. Object segmentation locates and isolates entire items in the image either simultaneously or later. Parts are categorized based on their functions or features by assigning a specific region and label to each split component. Establishing relationships between the segmented components and objects guarantees that every component is correctly linked to a particular object. The last phase confirms the segmentation process by confirming that each component has been appropriately identified, labeled, and their links to objects have been formed.
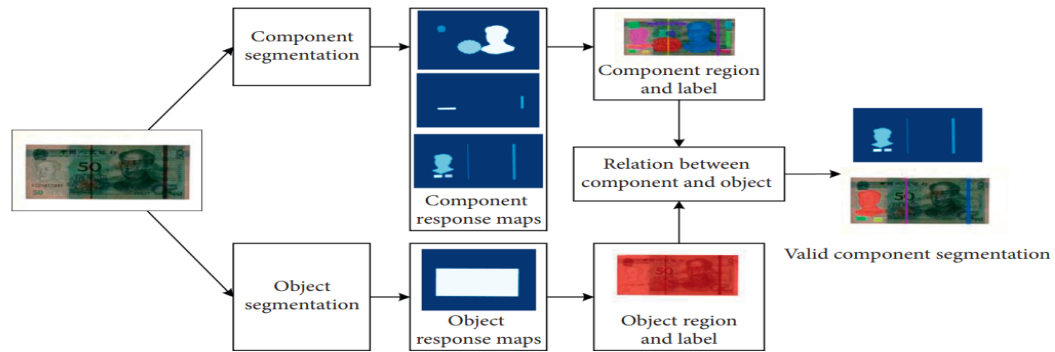

Figure 2: Semantic segmentation flowchart employing encoder-decoder network under complex background.[69]

3. Deep Network Architectures for Semantic Segmentation

We said before in the section that certain deep networks have become widely used benchmarks in the area due to their impressive performance. Among them are DeepLabv2, ResNet, VGG-16, MCG, AlexNet, and GoogLeNet. Because of their immense power, these networks are frequently the foundation of numerous segmentation models. As such, this section will be devoted to their analysis.

3.1. ResNet

The ResNet (Residual Network) architecture is introduced in the paper "Deep Residual Learning for Image Recognition" by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun [32]. This design is noteworthy for winning the ILSVRC-2016 with an astounding accuracy of 96.4%. The 152-layer network's depth and the addition of residual blocks are the main innovations. Residual blocks use identity skip connections to overcome the difficulty of training deep architectures. The disappearing gradients issue is resolved by these connections, which allow layers to replicate their inputs to the following layer. This method's logical goal

is to make sure that every layer gains fresh and distinct characteristics from its input, which improves the model's capacity to recognize complex patterns. With its inventive use of residual connections and its victory in the ImageNet competition, ResNet's effect on the area of deep learning has been enormous, influencing succeeding architectures and establishing a new benchmark for training exceedingly deep neural networks.

3.2. Dilation

According to the study paper "Multi-Scale Context Aggregation by Dilated Convolutions," the idea of dilation[71] is essential for improving convolutional neural networks' performance, especially when it comes to semantic segmentation—which is the process of giving each pixel in an image a name. Dilation is a method that increases convolutional layers' receptive fields without lowering the feature maps' spatial resolution. This is made possible by the use of dilated convolutions, which let the network gather more contextual data by allowing filters to be applied at intervals rather than at each pixel.

The mathematical formulation of the dilated convolution operator is expressed as follows:

$$(F * l\, k)(p) = \sum_{s+lt} F(s)k(t)$$

A discrete function is represented by F in this equation, a discrete filter by k, and the dilation factor by l. The dilation factor establishes the distance between filter applications, enabling a single filter to affect a greater portion of the input data. This capacity allows the model to collect data from a larger range of input without the requirement for downsampling, which can result in the loss of crucial spatial details. It is especially helpful for tasks that require comprehending the context around each pixel.

The capacity of dilated convolutions to accommodate an exponential expansion of the receptive field is among its most important advantages. The model can successfully include multi-scale contextual information because as the dilation factor rises, the region of the input that contributes to the output grows rapidly. Understanding the interactions between pixels at different scales is important for semantic segmentation, as it can greatly improve prediction accuracy. The authors carry out a number of in-depth studies to confirm the efficacy of their strategy. They show that substantial accuracy gains can be achieved by integrating dilated convolutions into current semantic segmentation designs. By offering a more thorough comprehension of the input data, the context module significantly improves the performance of cutting-edge models by enabling the network to take use of contextual cues that are essential for producing precise predictions.

3.3.VGG

The focus of the research "Very Deep Convolutional Networks for Large-Scale Image Recognition" is on examining how convolutional network depth affects the accuracy of picture recognition, with a particular emphasis on the VGG design [34]. With the use of tiny (3 x 3)

convolution filters in every layer, the authors suggest a ConvNet design that gradually adds additional convolutional layers to achieve more depth. The authors show how greater representation depth improves classification accuracy by comparing several ConvNet setups on the ILSVRC classification problem. Specifically, even with very modest pipelines, the VGG architecture achieves state-of-the-art accuracy on the ImageNet challenge dataset and exhibits exceptional performance in numerous image recognition datasets. Along with outlining significant changes made to the study, it offers insights into the ILSVRC 2014 object localization system related to VGG. In summary, the authors provide insightful research into the architecture and functionality of extremely deep ConvNets, notably VGG, and shed light on these topics. Their findings are particularly relevant for the field of large-scale image recognition.
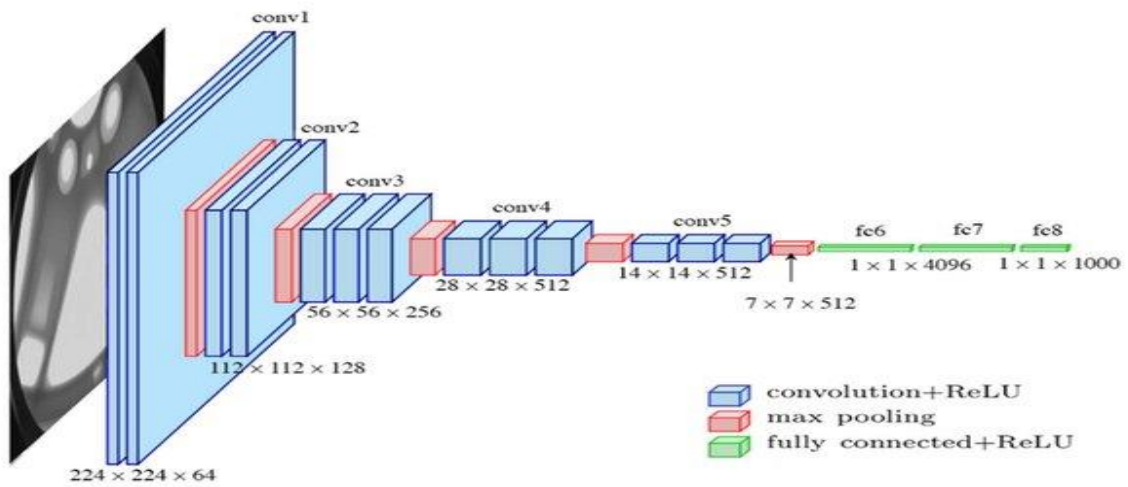

Figure 3: Typical architecture of the VGG model.

3.4. AlexNet

AlexNet, a ground-breaking deep Convolutional Neural Network (CNN) that won an unprecedented victory in the ILSVRC-2012 competition, is introduced in the paper "ImageNet Classification with Deep Convolutional Neural Networks" by Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton [14]. With a TOP-5 test accuracy of 84.6%, AlexNet outperformed rivals using conventional methods by a significant margin; in the same challenge, the nearest opponent obtained an accuracy of 73.8%. Krizhevsky et al. suggested an architecture that was very simple but quite successful. It was composed of five convolutional layers: three fully-connected layers, max-pooling layers, Rectified Linear Units (ReLUs) as non-linearities, and dropout added for regularization. The breakthrough in computer vision that AlexNet's success brought about demonstrated the promise of deep neural networks for image categorization applications. The work paved the way for deep learning's further developments and deep CNNs' broad use in a range of computer vision applications.
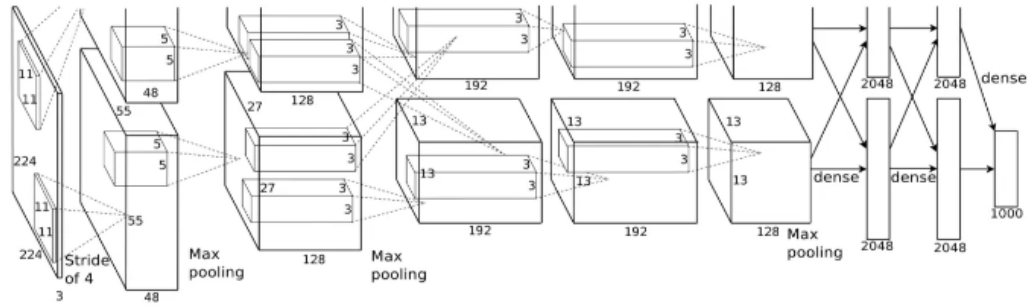
Illustration of AlexNet's architecture. Image credits to Krizhevsky et al., the original authors of the AlexNet paper.

Figure 4: Illustration of AlexNet's architecture

3.5. GoogLeNet

In the paper "Going Deeper with Convolutions," the Inception architecture is presented using the GoogLeNet model as an example [35]. This deep convolutional neural network, named after the "we need to go deeper" internet meme, emphasizes greater network depth and is intended for computer vision applications. In the ILSVRC 2014 tasks, the architecture which includes the Inception module achieves state-of-the-art performance in object identification and picture recognition, significantly surpassing previous models. Its better use of computer resources, achieved by carefully balancing depth and breadth while keeping a steady computational budget, is noteworthy. The Hebbian principle and multi-scale processing intuition drive the Inception architecture, which shows promise in enhancing neural networks for computer vision since it achieves notably higher accuracy than the state of the art. It achieves competitive performance in identification tasks even without bounding box regression or context use. Benefits of the architecture include emphasizing computational economy and achieving a considerable quality boost with a minimal increase in computing needs when compared to shallower networks. The design considerations guarantee cost-effectiveness and practical use in real-world applications, even on big datasets. All things considered, the Inception architecture which is best represented by GoogLeNet represents a noteworthy breakthrough in deep learning for computer vision, providing enhanced precision, effective use of resources, and practicality in many contexts.

3.6. DeepLab

A complex framework for semantic picture segmentation, the DeepLab architecture makes use of cutting-edge methods to boost feature extraction and increase segmentation accuracy. Atrous convolution[82] is a key component of its design because it provides fine control over feature response resolution, making it possible to capture multi-scale contextual information without adding more parameters or computing power. Using numerous parallel atrous

convolutional layers with varied sampling rates, the design uses Atrous Spatial Pyramid Pooling (ASPP) to efficiently separate objects at different scales. Besides, DeepLab combines the advantages of probabilistic graphical models and deep convolutional neural networks (DCNNs) to refine segmentation findings and enhance border localization by integrating fully connected Conditional Random Fields (CRFs). DeepLab achieves state-of-the-art performance on multiple semantic segmentation benchmarks thanks to its improved feature representation, which is gained from its foundation in residual networks (ResNet).

3.7. CRFasRNN

A sophisticated technique for semantic image segmentation called CRF-RNN (Conditional Random Fields as Recurrent Neural Networks) combines RNNs and CRFs into a single, end-to-end trainable neural network architecture. By improving pixel-by-pixel predictions and promoting smoothness in segmentation outputs—where neighboring pixels with similar appearances are more likely to share the same label—CRFs are probabilistic models that are used to enforce spatial dependencies. Traditionally, Convolutional Neural Networks (CNNs) produced the initial predictions, and then CRFs were applied as a post-processing step. On the other hand, CRF-RNN is novel in that it incorporates CRF inference as a recurrent process, enabling joint CNN and CRF training. The outputs of this integration are segmentation that is more precise and well-rounded. By using recurrent layers to update the segmentation mask iteratively, CRF-RNN is able to enhance performance on dense pixel labeling tasks and differentiate object borders more accurately. The method, which was first presented by Shuai Zheng et al. in their 2015 publication "CRF as RNN: Conditional Random Fields as Recurrent Neural Networks,"[72] has been demonstrated to perform better than standalone CNNs, particularly in datasets where accurate segmentation is crucial, such as PASCAL VOC and MS COCO.

3.8. ParseNet

Designed for semantic segmentation, ParseNet[73] is an end-to-end convolutional neural network that successfully resolves local ambiguities in pixel classification by incorporating global context. ParseNet incorporates global context directly into a fully convolutional network (FCN) without segmenting the image, in contrast to earlier techniques that depended on patch-based frameworks. This enables joint predictions of all pixel values. Using unpooling and selective application to various feature maps, the architecture creates a context vector by pooling feature maps over the entire image. This vector is then appended to the features sent to following layers. ParseNet uses L2 normalization and a scaling factor that is trained by backpropagation to handle the variations in feature scales, which greatly improves the FCN's performance. The authors show that adding global context results in accuracy that is on par with post-processing techniques that use detailed structure information, like Conditional Random Fields (CRFs). Segmentation results on the PASCAL VOC2012 test set fall within the DeepLab-LargeFOV-CRF standard deviation. With its resilience and simplicity of design,

ParseNet is an easy and effective choice for semantic segmentation tasks, achieving accuracy on par with more complicated architectures with little additional training or inference time. ParseNet achieves near state-of-the-art performance on PASCAL VOC2012 and state-of-the-art results on SiftFlow and PASCAL-Context, according to extensive experimental validation. The authors are interested in combining ParseNet with structured training and inference methods to further improve performance.

3.9. FCN

Fully Convolutional Networks (FCNs)[74] were first introduced for pixel-wise prediction tasks such as semantic segmentation in the paper "Fully Convolutional Networks for Semantic Segmentation" by Jonathan Long, Evan Shelhamer, and Trevor Darrell, which was presented at the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). The authors enabled dense output predictions by replacing fully connected layers with convolutional ones, converting conventional convolutional neural networks (CNNs) into fully convolutional architectures that allowed the network to generate pixel-level categorization maps. By upsampling feature maps using deconvolution (transposed convolution) layers to recover spatial resolution lost during pooling, FCNs effectively carry out end-to-end learning for segmentation. Skip connections allow the model to capture strong semantic knowledge while preserving fine features between deeper, low-resolution layers and early, high-resolution layers. The scientists created FCN variants of popular image classification networks like AlexNet, VGGNet, and GoogLeNet by utilizing pre-trained models for better segmentation results. On datasets such as PASCAL VOC, FCNs obtained state-of-the-art results, striking a significant advantage over region-based techniques. This work continues to have an impact on applications in satellite imagery, autonomous driving, and medical imaging. It also lay the groundwork for contemporary segmentation systems such as U-Net, SegNet, and DeepLab. FCNs are an essential tool in computer vision, having revolutionized dense prediction problems with their introduction.

3.10. SegNet

A ground-breaking deep learning architecture designed for semantic picture segmentation is presented in the 2015 publication "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation" by V. Badrinarayanan, A. Kendall, and R. Cipolla. With the help of its pre-trained weights, the encoder in SegNet's[79] encoder-decoder framework—which is based on the VGG16 model—is able to efficiently extract hierarchical features from input images. By using the pooling indices from the encoder, the architecture's decoder recreates high-resolution segmentation maps while maintaining spatial information that is frequently lost in conventional upsampling methods. The decoder's structure is mirrored by the encoder. The authors highlight SegNet's effectiveness by showing how it can produce precise segmentation results without using as much processing power as deep learning models do.

Because of this, SegNet is especially well-suited for real-time applications where accurate and timely picture segmentation is essential, such as robotic vision and autonomous driving.

In comparison to cutting-edge segmentation techniques, extensive testing on benchmark datasets like the CamVid dataset shows that SegNet performs competitively and demonstrates its durability in a variety of contexts. The benefits of employing pooling indices, the effects of varying network depths, and the importance of training procedures to maximize segmentation quality are just a few of the architecture's features that are covered in this research. In addition, SegNet laid the groundwork for other sophisticated segmentation structures, which has had a substantial impact on further research in the subject. SegNet is now a well-known point of reference for researchers and practitioners looking to enhance semantic segmentation in a variety of applications due to its effectiveness and clarity. This has established SegNet as a crucial component of the continuous advancement of deep learning techniques in computer vision.

3.11. ReSeg

F. Visin and colleagues introduced ReSeg[80], a semantic segmentation model based on an architecture based on recurrent neural networks (RNNs). It functions in an encoder-decoder architecture, where the encoder extracts high-level, coarse feature maps from input pictures using conventional convolutional neural networks (CNNs). ReSeg's decoder, which uses recurrent convolutional layers to iteratively improve the segmentation map, is its special power. Through the use of RNNs—more especially, convolutional gated recurrent units, or ConvGRUs—the model is able to repeatedly acquire and process spatial dependencies throughout the image. This is essential for increasing the accuracy of segmentation, particularly in complicated photos with occlusions or imprecise object borders. ReSeg can iteratively revisit its former states thanks to the recurrent structure, which progressively improves pixel-wise predictions and preserves fine details. More accurate segmentation results are produced by this iterative refining, which also helps to account for contextual information, improve object boundaries, and fix past errors. ReSeg shows how segmentation performance may be enhanced by combining the advantages of CNNs for feature extraction and RNNs for context-aware processing. This is especially useful in complex settings where spatial linkages or unclear object boundaries are significant.

3.12. ENet

ENet is a deep neural network architecture that was created especially for real-time semantic segmentation, with a heavy emphasis on speed and efficiency. It was first presented by A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello[81]. In contrast to numerous other segmentation models that give precedence to accuracy over computational resources, ENet is tailored for settings with constrained processing capabilities, such real-time apps and mobile devices. In order to accomplish this, it uses an encoder-decoder structure while significantly lowering the computing burden on the encoder and decoder. The architecture minimizes the

number of operations while retaining useful feature extraction by employing downsampling techniques early in the network to significantly reduce the spatial resolution of the input. ENet includes several asymmetric convolutions, bottleneck modules, and dilated convolutions that efficiently collect multi-scale information without adding to computational complexity in order to further boost efficiency. PReLU (Parametric Rectified Linear Units) activations are another technique that ENet employs to improve gradient flow during training and hasten convergence. While ENet loses some accuracy when compared to larger systems, it achieves the best possible speed and performance trade-off, which makes it perfect for real-time applications where quick processing is essential. The model demonstrates its feasibility in real-world scenarios where reasonable accuracy and speed are required, by achieving state-of-the-art segmentation performance on many datasets.

3.13. 2D-LSTM
A unique method for semantic segmentation utilizing 2D-Long Short-Term Memory (LSTM)[78] networks is presented in the publication "Scene Labeling with LSTM Recurrent Neural Networks" by W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki. Specifically designed for picture data, the 2D-LSTM architecture expands the capabilities of regular LSTMs, which are usually used to sequential data, to function over two-dimensional grids. To enable the network to comprehend pixel interactions in all directions, the 2D-LSTMs in this model process an image by capturing spatial dependencies both vertically and horizontally. This has significant importance in scene labeling tasks where correct object segmentation depends heavily on the context and spatial dependencies across various areas of the image. The way the 2D-LSTM works is that it processes each pixel in a way that resembles a grid, repeatedly going over the image in both horizontal and vertical directions while preserving hidden states that hold contextual data. The network can collect long-range dependencies throughout the image thanks to this recurrent mechanism, which helps it get over the drawbacks of conventional convolutional neural networks (CNNs), which usually rely on local receptive fields and struggle to capture more contextual information. Because the architecture uses 2D-LSTMs, it performs well in situations where precise scene understanding requires knowledge of both local details and global context, like occlusions or intricate item arrangements.

3.14. DAG-RNN
Directed Acyclic Graph Recurrent Neural Networks (DAG-RNNs)[76] are a novel technique to semantic segmentation, as presented in the paper "DAG-Recurrent Neural Networks for Scene Labeling" by B. Shuai, Z. Zuo, G. Wang, and B. Wang. By representing pixel associations as a directed acyclic graph (DAG), DAG-RNNs are engineered to capture long-range dependencies in images, in contrast to conventional Convolutional Neural Networks (CNNs) that mostly rely on local context. The fundamental idea is to multiply the ways in which contextual information spreads throughout the image, hence improving the network's understanding of global structures while preserving the local accuracy needed for pixel-level

labeling. This system aggregates contextual information throughout the entire image by treating each pixel as a node in the DAG and passing information along edges between nodes. Through the use of a recurrent process, DAG-RNNs iteratively improve the accuracy of segmentation results by refining their predictions. The network performs especially well in complicated scenarios with unclear object borders or obscured objects. The architecture outperforms classic CNN-based techniques in scene labeling challenges by capturing contextual links between objects and their surroundings more effectively. Additionally, DAG-RNNs can be combined with current CNN designs to improve speed by incorporating the ability to represent long-range dependencies. Because of its great flexibility and scalability, the model may be applied to a wide range of scene understanding tasks that call for meticulous segmentation that takes into account both local and global image data.

3.15. Multi-scale-CNN-Eigen

The "Multi-scale-CNN-Eigen"[21] method represents a substantial development in computer vision, especially with regard to scene comprehension. This method allows for a thorough comprehension of the environment by integrating surface normal estimate, depth prediction, and semantic labeling into a single multi-scale convolutional network. This is important for applications like as augmented reality, robotics, and autonomous driving, where a thorough grasp of the surroundings is necessary for efficient navigation and interaction. The method examines images at various resolutions using a convolutional neural network (CNN) architecture to capture both fine-grained details and large contextual information. The "Multi-scale-CNN-Eigen" solution does away with the necessity for low-level segmentation approaches by simply regressing pixel maps from the input images. This streamlines the pipeline and improves the model's capacity to learn from unprocessed input.

Using a series of scales, the architecture iteratively improves predictions, with coarser scales giving a broad picture of the world and finer scales concentrating on specific details. Tasks that demand precise and comprehensive outputs, such as semantic labeling, surface normal estimate, and depth prediction, benefit greatly from this multi-scale processing. For scholars and professionals in the field, the "Multi-scale-CNN-Eigen" approach presents a strong substitute since it provides a flexible and effective model that performs well on a variety of tasks. Future research and applications in computer vision will have a strong basis thanks to the field's continued evolution.

3.16. Bayesian SegNet

A more sophisticated iteration of the SegNet architecture called Bayesian SegNet[75] uses Bayesian inference to assess uncertainty in semantic segmentation tasks. This approach, which was introduced by Kendall et al., uses Monte Carlo dropout during inference, allowing for the approximation of epistemic uncertainty, or uncertainty resulting from the model's parameters. Bayesian SegNet provides confidence levels for pixel-wise classifications by aggregating predictions to estimate both mean and variance across repeated forward passes with dropout

enabled. Understanding the model's confidence is essential for making sound decisions in safety-critical applications like autonomous driving and medical imaging, where this uncertainty estimation is very helpful. Bayesian SegNet improves prediction reliability by addressing both epistemic and aleatoric uncertainty (inherent noise in the data). This makes it useful for tasks that require accurate scene understanding and for directing active learning strategies in environments where the data is noisy or unclear.

3.17. rCNN

The publication "Recurrent Convolutional Neural Networks for Scene Parsing" presents rCNNs[77], which are a type of convolutional neural network that incorporates recurrent connections to capture spatial correlations in an input image. Recurrent layers are used by rCNNs, in contrast to regular CNNs, which process information locally. This allows the network to continually process features from earlier steps and accumulate contextual knowledge over a number of iterations. By capturing both local and global context for more precise segmentation, this method improves scene parsing by modeling long-range dependencies throughout the image. A convolutional layer and a recurrent layer, which are implemented using structures like Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) units, make up the rCNN architecture. By feeding their output back into the network, these recurrent layers allow the network to gradually improve its predictions. The primary benefit of rCNN is in its capacity to manage intricate spatial relationships inside scenes, hence enhancing parsing accuracy for objects with diverse sizes and forms. Sequential spatial correlations may be accounted for by the model thanks to its recurrent structure, which is crucial for tasks like semantic segmentation and object border identification.

4. Available Datasets

In any machine learning system, data is essential, but in deep networks, the need of data is heightened. Thus, for any segmentation system that uses deep learning methods, gathering sufficient data into a dataset is crucial. One needs time, domain expertise to select pertinent information, infrastructure to obtain that data and convert it to a format that the system can understand and learn from, and other resources to produce a viable dataset, which should be large enough in scale and accurately reflect the use case of the system. This assignment is among the hardest to do in this situation, despite its seeming simplicity in comparison to intricate neural network architectural designs. Because of this, the best course of action is typically to use an established standard dataset that is sufficiently representative for the problem domain. This tactic also helps the community because standardized datasets make it possible to compare systems fairly. In fact, many datasets are a part of a challenge that withholds some data so that developers cannot assess their algorithms. This allows for the evaluation of numerous methods in a fair contest that ranks them according to their actual performance without the influence of biased data.

- **ADE20K**: A large and varied collection created for computer vision research on semantic segmentation is the ADE20K [36] (ADE20K Scene Parsing) dataset. ADE20K, which includes more than 20,000 high resolution photos, stands out for having precise annotations at the pixel level, including 150 different object categories and item classes. Semantic segmentation algorithms face a particular problem in handling complex spatial connections and differences in item sizes due to the vast spectrum of scenes captured by the dataset, which includes both indoor and outdoor locations. As a benchmark that makes cross-methodology comparisons easier, ADE20K has established itself as a key tool for assessing the resilience and performance of semantic segmentation models. The importance of the dataset is demonstrated by its critical role in the development of scene comprehension algorithms, giving researchers a platform to create models that can classify individual pixels in intricate and realistic visual contexts with exquisite granularity.
- **COCO Stuff**: Specifically designed for semantic segmentation tasks with an emphasis on stuff classes, the COCO Stuff [37] dataset is a useful addition to the COCO (Common Objects in Context) dataset. This includes comments for 91 different kinds of objects, including the sky, roads, and greenery. With its full grasp of visual scenes that goes beyond object-centric annotations, this dataset is an invaluable tool for training and assessing models in the complex context of semantic segmentation. With its extensive pixel-level annotations for a wider variety of semantic classifications, COCO Stuff is a companion to the original COCO dataset, which focuses largely on object recognition. With the use of this dataset, scene understanding research may proceed and picture analysis can become more comprehensive and contextual. Stuff classes are added to the dataset, which makes it more useful for training models that identify various components in a picture and understand complex spatial connections. The COCO Stuff dataset is a well-known benchmark in the field of semantic segmentation research and has played a significant role in advancing the discipline.
- **Pascal VOC (Visual Object Classes)**: As the industry standard for object identification and segmentation tasks, the Pascal VOC [38] (Visual Object Classes) dataset is a seminal resource in the field of computer vision. The collection provides an extensive and diversified range of visual settings, containing photos from 20 different object categories, such as automobiles, animals, and household goods. One of Pascal VOC's unique features is its painstaking annotations at the pixel level. These annotations offer each picture comprehensive ground truth data, making it easier to train and assess models that have a precise grasp of object boundaries. The training, validation, and test sets of the dataset are purposefully divided to promote standardized evaluation processes and guarantee uniform performance evaluations across various algorithms. Pascal VOC has been a driving force behind several breakthroughs and developments in the field of object identification and segmentation research, significantly influencing its course. Researchers use Pascal VOC tasks to evaluate the performance of their algorithms and compare them against real-world settings. The Pascal VOC dataset remains a fundamental resource, contributing

significantly to cooperative research endeavors, facilitating performance assessments, and propelling the advancement of cutting-edge approaches for object detection and segmentation.

- **Pascal Context**: A collection of supplementary annotations for the PASCAL VOC 2010 detection challenge, a well-liked benchmark for tasks involving object recognition and semantic segmentation, is called the "Pascal Context" [39] dataset. The dataset covers more than 400 kinds of objects, things, and hybrids, and goes beyond the original PASCAL semantic segmentation problem by providing pixel-wise labels for the whole picture. Things like automobiles, pets, and chairs are examples of objects since they are readily numbered and moved. Things that are amorphous or uncountable, like the sky, grass, and water, are referred to as stuff. Things like fences, curtains, and roads are examples of hybrids since they possess both stuff-like and object-like qualities. There are 10,103 photos in the dataset for training and validation, and 9,637 images for testing. 20 categories, including human, animal, vehicle, and indoor, are used to group the photos. The dataset helps create novel models and techniques that may make use of the extensive and varied annotations, as well as for assessing the significance of context for object recognition and semantic segmentation in the real world. Regarding semantic segmentation and associated tasks, including zeroshot learning, human parsing, saliency detection, surface normals estimation, and border detection, the dataset is utilized to suggest and contrast various methods.

- **PASCAL Part**: An additional set of annotations for the well-known PASCAL VOC 2010 dataset a benchmark for computer vision tasks including object detection and segmentation is called the PASCAL Part [40] dataset. With segmentation masks for every body part of the object, the PASCAL Part dataset extends the capabilities of the original PASCAL object detection job. It offers the silhouette annotation for categories (like boats) that don't have a fixed set of pieces. It can also be used as a set for segmenting human semantic parts: There are several people in free-form positions and occlusions in each image (1,716 for training and 1,817 for testing). It offers meticulous annotations at the pixel level for six body parts: the head, chest, upper and lower arms, and upper and lower legs. 9,637 images are used for testing and 10,103 images are used for training and validation. Along with 39 part classes, it covers 20 object classes.

- **NYU-Depth V2 (NYUDv2)**: The RGB and depth cameras of the Microsoft Kinect gadget recorded a variety of indoor scenarios that make up the NYU-Depth V2 (NYUDv2) [41] dataset. It has 1449 pairs of RGB and depth images that are aligned, and each pixel has a dense multi-class label. An instance number is also labeled on each object. With 407,024 unlabeled frames, 464 additional scenes from 3 cities are also included in the collection. Using a colorization strategy, the dataset has been preprocessed to fill in the missing depth values. Semantic seg- mentation, depth estimation, surface normal estimation, 3D item detection, and scene completion are among the tasks for which the dataset is helpful. Because of its extensive and diverse material, NYU-Depth V2 is an essential tool for

academics who want to improve computer vision models' ability to perceive depth and comprehend the structures of scenes in interior settings.

- **SUN RGBD**: An extensive collection of RGB-D photos for scene analysis tasks is called the SUN RGBD [42] dataset. It has 10,335 photos from four separate sensors that span a range of inside situations, including workplaces, classrooms, bedrooms, and kitchens. Rich annotations, such as 2D polygons, 3D bounding boxes, item orientations, room layouts, and scene categories, are included for every image in the dataset. Concerning all significant scene understanding tasks, including semantic segmentation, 3D object detection, monocular depth estimation, and scene recognition, the dataset seeks to push the state-of-the-art. Additionally, the dataset makes it possible to assess 3D metrics and cross-sensor generalization. SUN RGBD offers a realistic and comprehensive dataset that captures the nuances of interior situations, making it a standard for algorithms tackling problems like object detection, scene parsing, and 3D scene reconstruction. This has led to breakthroughs in computer vision.

- **SUN3D**: The whole 3D extent of numerous indoor spaces is captured in the SUN3D [43] dataset, which is a massive collection of RGB-D movies with object labels and camera poses. It includes 415 sequences that were taken in 41 distinct buildings and 254 distinct areas [43]. Semantic segmentation of the scene's items and camera posture information is contained in every frame. The purpose of the dataset is to support research in the areas of semantic segmentation, object recognition, structure from motion, and scene interpretation. Online resources include the SUN3D database, the web-based 3D annotation tool, and the source code for the generalized bundle adjustment. SUN3D, which comprises over 8,000 RGBD video sequences and related ground truth annotations, is now widely used as a benchmark for assessing algorithms for tasks including object detection, scene interpretation, and 3D reconstruction. The content of the dataset includes both indoor and outdoor settings, as well as a variety of difficulties such as changing illumination, occlusions, and object interactions.

- **Semantic Boundaries Dataset (SBD)**: Using semantic segmentation, the Semantic Boundaries Dataset (SBD) [44] predicts pixels on an object's edge rather than its inside. There are 8498 training and 2820 test images in the dataset, which is made up of 11318 photos from the PASCAL VOC2011 challenge's trainval set. One of the twenty Pascal VOC classes are labeled on object instance boundaries in this dataset, which also features precise figure/ground masks. Tasks including edge recognition, semantic contour prediction, and interactive segmentation benefit from the application of the SBD.

- **SYNTHetic Collection of Imagery and Annotations (SYNTHIA)**: A huge synthetic dataset called the SYNTHetic Collection of Imagery and Annotations (SYNTHIA) [45] was created specifically for the job of semantic segmentation and related scene interpretation issues in the context of driving scenarios. It is comprised of over 200,000 high-definition pictures taken from separate snapshots and video streams. The pictures are created from a virtual metropolis with various scenes, dynamic objects, different seasons,

lighting, and weather effects. Thirteen classes, including sky, building, road, car, pedestrian, etc., have accurate pixellevel semantic annotations included with the photographs. Additionally, eight RGB cameras that combine to form a binocular 360-degree camera and eight depth sensors are simulated in the dataset. SYNTHIA is a benchmark that is extremely useful for assessing algorithms in the context of urban scene analysis, where access to huge annotated datasets can be difficult due to its synthetic yet highly realistic content. Researchers use SYNTHIA to improve the state-of-theart in semantic comprehension of urban landscapes, train and validate models, and evaluate how resilient they are to various environmental conditions.

- **Berkeley Deep Drive (BDD100K)**: This extensive and varied driving video collection, called "Berkeley Deep Drive (BDD100K)," [46] is intended for use in computer vision research. One hundred thousand movies total, each lasting roughly forty seconds and captured at 30 frames per second, are included. The videos feature a variety of American locales, climates, and lighting conditions. Ten tasks, including object detection, semantic segmentation, lane detection, and drivable area segmentation, are also supported by comprehensive annotations in the dataset. Developing and testing image recognition algorithms for autonomous driving is intended to be made easier with the help of this dataset. Due to its unique emphasis on realworld driving scenarios, BDD100K is a useful benchmark that will help advance our knowledge of intricate urban settings and strengthen the resilience of computer vision models in practical applications.

- **The Cambridge-driving Labeled Video Database (CamVid)**: Videos taken from the viewpoint of a moving car make up the Cambridge-driving Labeled Video Database (CamVid) [47]. A 960x720 resolution camera mounted on an automobile dashboard originally recorded five video sequences that make up the CamVid (Cambridge-driving Labeled Video Database), a database for interpreting driving scenes and roads. More than ten minutes of excellent 30Hz video are included in the dataset, along with matching semantically tagged images at 1Hz and, in some cases, 15Hz. Each pixel in the dataset is assigned a ground truth label, which is associated with one of 32 semantic classes including column/pole, train, wall, lane markings (driving), parking block, tunnel, bicyclist, car, SUV/pickup/truck, bridge, sign, tree, pedestrian, miscellaneous text, traffic light, sidewalk, road shoulder, road, animal, child, vegetation, archway, fence, truck- /bus, motorcycle, sky, void, building, cart luggage, traffic cone, and other moving object. The dataset underwent hand annotation, and its accuracy was confirmed by two individuals. In addition, the dataset provides 3D camera postures for every frame, camera calibration sequences, and specially designed labeling software. The dataset can be used to assess methods for label propagation, pedestrian detection, and multi-class object recognition.

- **Cityscapes**: With annotations for 30 classes and 8 categories, the Cityscapes [48] dataset is a large-scale database for semantic urban scene interpretation. It offers information from 50 cities as well as a range of functions like image-to-image translation, segmentation, and depth estimation. About 5000 finely annotated photos and 20,000 coarsely annotated

images make up the dataset. During several months, during the day, and with favorable weather, data was collected in fifty cities. Due to the fact that it was initially captured as video, the frames were carefully chosen to include a lot of dynamic elements, a changing scene arrangement, and a changing background. In order to enable research that aspires to use vast volumes of (weakly) labeled data, the dataset is meant for evaluating the performance of vision algorithms for three fundamental tasks of semantic urban scene understanding: pixel-level, instance-level, and panoptic semantic labeling.

- **Youtube-Objects**: The Youtube-Objects [49] dataset is a massive collection of object videos from YouTube that was gathered by searching for the names of ten different object classes, including motorbikes, trains, dogs, cats, cows, planes, birds, boats, cars, and motorcycles, from the PASCAL VOC Challenge. For every class in the dataset, there are nine to twenty-four videos totaling five hundred thousand frames. The download size is 89 gigabytes. Every video has a different length, ranging from thirty seconds to three minutes. The films lack precise location and size annotations; instead, they merely show the existence of an object belonging to the relevant class. Along with optical flow and superpixels for every frame, the collection also includes tubes, motion segments, bounding-box annotations, and point tracks for some video frames. The dataset is meant for tasks like segmenting and tracking objects in videos as well as learning object class detectors from videos with sparse annotations.

- **Materials in Context (MINC)**: A vast, publicly available collection of materials found in the wild, "Materials in Context (MINC)" [50] includes 7061 classified material segmentations across 23 material categories in addition to 3 million labeled point samples. The rich surface texture, geometry, lighting, and clutter of real-world materials are captured in the dataset, which spans a wide spectrum of material types like cloth, glass, leather, metal, stone, wood, etc. The OpenSurfaces database, a comprehensively documented catalog of surface appearance, was the source from which the material annotations for the dataset were extracted. The resolution of a photograph is usually 500×800 or 800×500. Numerous tasks, including material recognition, material segmentation, material editing, and material synthesis, can be accomplished with the help of the MINC dataset. For academic reasons alone, the dataset is openly accessible to the general public.

- **KITTI**: One of the most often used datasets for mobile robots and autonomous driving is KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) [51]. It is made up of hours' worth of traffic scenarios captured by multiple sensor modalities, such as RGB and grayscale stereo cameras with high resolution and a 3D laser scanner. Although widely used, the dataset itself lacks ground truth for semantic segmentation. A GPS/IMU inertial navigation system, a Velodyne 3D laser scanner, and high resolution color and grayscale stereo cameras were installed on the moving platform where the video was captured. Six hours' worth of traffic scenarios from various contexts and circumstances around Karlsruhe, Germany, are included in the collection. For a few of the jobs, the dataset additionally includes evaluation metrics, online benchmarks, and accurate

ground truth annotations. For research on mobile robots and autonomous driving, one of the most well-liked and difficult datasets is the KITTI dataset.

- **Adobe's Portrait Segmentation**: For the purposes of face parsing and portrait segmentation, a set of pictures and segmentation masks is called "Adobe's Portrait Segmentation" [52]. In order to enhance the features of video conferencing apps, such as face beautifying and backdrop removal, researchers from SaluteDevices in Russia built it. 20,000 mostly indoor images of 8,377 distinct individuals make up the dataset, which also includes fine-grained segmentation masks divided into 9 groups, including skin, hair, eyes, nose, mouth, teeth, beard, spectacles, or background. High-quality and diverse data was intended to be included in the dataset, which was created via crowdsourcing platforms for image collection and labeling. The dataset can be used for skin enhancement and teeth whitening jobs since, in contrast to most face parsing datasets, the beard is not regarded as a component of the skin mask and the inside area of the mouth is separated from the teeth. Both the trained models and the dataset are accessible to the public on GitHub.

- **Densely-Annotated VIdeo Segmentation (DAVIS)**: A high-quality and highresolution dataset for the job of video object segmentation is the DenselyAnnotated VIdeo Segmentation (DAVIS) [53] dataset. Each of the 50 video sequences' 3455 frames has pixel-level masks added to identify one or more things of interest. There are two versions of the dataset available: DAVIS 2017, which contains numerous labeled objects per movie, and DAVIS 2016, which has only one annotated object per film. A variety of circumstances, including occlusions, motion blur, appearance alterations, and varied camera motions, are covered by the dataset. Additionally, the dataset offers a number of evaluation criteria to gauge how well video object segmentation algorithms perform, including region similarity, contour correctness, and temporal stability. DAVIS is distinguished by its painstaking annotation, which offers fine-grained information on object boundaries and motion patterns. With this degree of detail, DAVIS is a priceless tool for assessing algorithms in video segmentation tasks, leading to improvements in the comprehension and modeling of dynamic video sequences.

- **SIFT Flow**: Images of various landscapes and objects, including buildings, grass, trees, etc., are included in the SIFT Flow [54] dataset. The goal of the collection is to provide dense correspondence between scenes, which is the ability to identify pixel-by-pixel matches across pictures with disparate scene attributes, such perspective, scale, or location. The dataset has 2,688 photos with a 256x256 pixel resolution. Every pixel in the photographs is labeled with either one of three geometric categories (horizontal, vertical, sky, etc.) or one of 33 semantic categories (building, grass, tree, etc.). There are 2,488 training photos and 200 test images in the dataset.

- **The Object Segmentation Database (OSD):** The collection known as the Object Segmentation Database (OSD) [55] comprises 111 RGB-D pictures of diverse indoor scenarios featuring various kinds of items, including cylinders, boxes, and obscured objects. The ground truth annotation, color image, and depth image for every entry are

provided by the dataset. In order to facilitate the assessment of object segmentation techniques that make use of both color and depth infor- mation, Andreas Richtsfeld of the Automation and Control Institute at TU Wien generated the dataset. One training set of 45 images and one test set of 66 images each make up the two subsets of the dataset.

- **Stanford Background**: Stanford Background [56] is a set of 715 outdoor images with pixel-level annotations for both semantic and geometric classes. The images are roughly 320 by 240 pixels in size, and each image has at least one foreground object and the location of the horizon. The images are taken from four publicly available datasets, which are LabelMe, MSRC, PASCAL VOC, and Geometric Context; the semantic classes are sky, tree, road, grass, water, building, mountain, and foreground object; the geometric classes are sky, horizontal, and vertical; the dataset also includes distinct image regions and the horizon position for each image. The dataset is intended to be used for evaluating techniques for both geometric and semantic scene understanding.

- **RGB-D Object Dataset**: With the use of WordNet hypernym-hyponym associations, a methodology akin to ImageNet, the 300 common household objects in the RGB-D Object Dataset [57] are categorized into 51 groups. An synchronized and aligned 640x480 RGB and depth image recording device, similar to the Kinect, was used to record this dataset at 30 frames per second. Video sequences were recorded for a complete rotation while each object was positioned on a turntable. Each object is shown in three different video sequences, each shot from a different height so that the object can be seen from several horizonfacing views. All 300 items' ground truth pose data is also included in the dataset. The collection can be applied to a number of tasks, including pose estimation, object detection, segmentation, and scene comprehension.

Table 1: Popular large-scale segmentation datasets[1]

Name	Purpose	Year	Classes	Data	Resolution	Sequence	Synthetic/Real	Samples (Training)	Samples (Validation)	Samples (Test)
PASCAL VOC 2012	Segmentation	2012	21	2D	Variable	✗	R	1464	1449	Private
PASCAL-Context	Generic	2014	540 (59)	2D	Variable	✗	R	10103	N/A	9637
PASCAL-Part	Generic-Part	2014	20	2D	Variable	✗	R	10103	N/A	9637
SBD	Generic	2011	21	2D	Variable	✗	R	8498	2857	N/A
Microsoft COCO	Generic	2014	80	2D	Variable	✗	R	82783	40504	81434
SYNTHIA	Urban (Driving)	2016	11	2D	960 × 720	✗	S	13407	N/A	N/A
Cityscapes (fine)	Urban	2015	30 (8)	2D	2048 × 1024	✓	R	2975	500	1525
Cityscapes (coarse)	Urban	2015	30 (8)	2D	2048 × 1024	✓	R	22973	500	N/A
CamVid	Urban (Driving)	2009	32	2D	960 × 720	✓	R	701	N/A	N/A
KITTI-Layout	Urban/Driving	2012	3	2D	Variable	✗	R	323	N/A	N/A
KITTI-Ros	Urban/Driving	2015	11	2D	Variable	✗	R	170	N/A	46
KITTI-Zhang	Urban/Driving	2015	10	2D/3D	1226 × 370	✗	R	140	N/A	112
Stanford background	Outdoor	2009	8	2D	320 × 240	✗	R	725	N/A	N/A
SiftFlow	Outdoor	2011	33	2D	256 × 256	✗	R	2688	N/A	N/A
Youtube-Objects-Jain	Objects	2014	10	2D	480 × 360	✓	R	10167	N/A	N/A
Adobe's Portrait Segmentation	Portrait	2016	2	2D	600 × 800	✗	R	1500	300	N/A
MINC	Materials	2015	23	2D	Variable	✗	R	7061	2500	5000
DAVIS	Generic	2016	4	2D	480p	✓	R	4219	2023	2180
NYUDv2	Indoor	2012	40	2.5D	480 × 640	✗	R	795	654	N/A
SUN3D	Indoor	2013	−	2.5D	640 × 480	✓	R	19640	N/A	N/A
SUNRGBD	Indoor	2015	37	2.5D	Variable	✗	R	2666	2619	5050
RGB-D Object Dataset	Household objects	2011	51	2.5D	640 × 480	✓	R	207920	N/A	N/A

5. Evaluation Metrics

A segmentation system's performance needs to be rigorously assessed in order for it to be beneficial and truly make a substantial contribution to the area. Furthermore, that assessment needs to be carried out with recognized and conventional parameters that allow for equitable comparisons with current approaches. A system's accuracy and execution time are just two of the many factors that need to be assessed in order to determine its validity and use. Certain metrics may be more important than others depending on the goal or context of the system; for example, accuracy may be sacrificed up to a degree in favor of speed of execution for real-time applications.

6.1 Execution Time

In semantic segmentation, speed, or runtime, is an important parameter, particularly since many systems have stringent inference time requirements. Although the training time may be considerable for very sluggish operations, it is typically not as significant because training is typically done offline. But giving precise times for methods can be deceptive because direct comparisons are challenging because hardware and backend implementation play a major role on performance. Still, repeatability is important because it allows researchers to fairly compare and evaluate the effectiveness of their methods for specific applications by publishing comprehensive execution durations, hardware specs, and benchmark settings.

6.2 Accuracy

Many evaluation criteria have been proposed and are commonly used to evaluate the precision of semantic segmentation methods. These metrics are often Intersection over Union (IoU) and pixel accuracy variations. We highlight the measures that are most frequently used to assess the performance of per-pixel labeling techniques in this domain. Please take note of the following notation for clarity: A background or void class is one of the k+1 classes (L0 to Lk), and the amount of pixels from class I that are anticipated to be class j is denoted by p_{ij}. In particular, true positives are indicated by p_{ii}, but false positives and false negatives are generally denoted by p_{ij} and p_{ji}, respectively.

- **Pixel Accuracy (PA)**

  Pixel Accuracy (PA), which determines the proportion of correctly categorized pixels to all pixels, is a straightforward but useful metric used in image segmentation. It gives an overall accuracy score by comparing the anticipated pixel labels with the ground truth labels. Improved performance in precisely segmenting the image is indicated by improved pixel accuracy.

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}} \qquad (1)$$

- **Mean Pixel Accuracy (MPA)**

  An improved form of Pixel Accuracy (PA), Mean Pixel Accuracy (MPA) is intended to give a more impartial assessment in semantic segmentation tasks. Mean Pixel correctness surpasses Pixel Accuracy in that it assesses each class's correctness independently before averaging the outcomes. Pixel Accuracy computes the total percentage of properly categorized pixels.

$$MPA = \frac{1}{k+1}\sum_{i=0}^{k}\frac{pii}{\sum_{j=0}^{k}pij} \qquad (2)$$

- **Intersection over Union (IoU)**

  IoU is a popular metric for assessing image segmentation models; it is often referred to as the Jaccard Index. The overlap between the ground truth and the anticipated segmentation is measured. By dividing the amount of overlap between the ground truth and predicted segments by the area of their union, the IoU value is computed. This statistic gives a clear picture of how closely the actual segmentation matches the projected segmentation.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \qquad (3)$$

- **Mean Intersection over Union (MIoU)**

  When evaluating image segmentation tasks, Mean Intersection over Union (MIoU) is a frequently used metric. Calculating the gap between the predicted segmentation regions and the ground truth allows one to assess a segmentation model's accuracy. More specifically, it shows the proportion of the union (all pixels that are part of either the ground truth or predicted regions) to the intersection (pixels that are successfully predicted).

$$MIoU = \frac{1}{k+1}\sum_{i=0}^{k}\frac{pii}{\sum_{j=0}^{k}pij+\sum_{j=0}^{k}pji-pii} \qquad (4)$$

- **Frequency Weighted Intersection over Union (FWIoU)**

  An improved form of Mean Intersection over Union (MIoU) that takes into consideration the frequency of each class's occurrence in the dataset is called Frequency Weighted Intersection over Union (FWIoU). In situations when certain classes predominate the dataset, it is intended to provide a more balanced evaluation metric by assigning greater weight to classes that appear frequently and less weight to classes that appear infrequently.

$$FWIoU = \frac{1}{\sum_{i=0}^{k}\sum_{j=0}^{k}pij}\sum_{i=0}^{k}\frac{pii\sum_{j=0}^{k}pij}{\sum_{j=0}^{k}pij+\sum_{j=0}^{k}pji-pii} \qquad (5)$$

6. Experimental Results

Different methods achieve different accuracy levels on the same, and also other datasets. The table below illustrates this with some examples of how certain methods work better for particular subsets. This yields varying results for the each method, and evaluates their performance on a dataset by examining these accuracy metrics (the strength and weaknesses of applying techniques). By comparing these results, one can decide which methods are optimal with the choice of dataset leading to higher performance or more reliable outcomes in data analysis. This comparison highlights the significance of appropriate method selection — methods should complement data characteristics.

Table 2: Performance results on different datasets[1]

Dataset	Methods	Accuracy (IoU)	Dataset	Methods	Accuracy (IoU)
PASCAL VOC-2012	DeepLab[19][70]	79.70	CamVid	DAG-RNN[76]	91.60
	Dilation[71]	75.30		Bayesian SegNet[75]	63.10
	CRFasRNN[72]	74.70		SegNet[79]	60.10
	ParseNet[73]	69.80		ReSeg[80]	58.80
	FCN-8s[74]	67.20		ENet[81]	55.60
	Multi-scale-CNN-Eigen[21]	62.60	CityScapes	DeepLab[19][70]	70.40
	Bayesian SegNet[75]	60.50		Dilation10[71]	67.10
PASCAL-Context	DeepLab[19][70]	45.70		FCN-8s[74]	65.30
	CRFasRNN[72]	39.28		CRFasRNN[72]	62.50
	FCN-8s[74]	39.10		ENet[81]	58.30
SiftFlow	DAG-RNN[76]	85.30	PASCAL-Person-Part	DeepLab[19][70]	64.94
	rCNN[77]	77.70	Stanford Background	rCNN[77]	80.20
	2D-LSTM[78]	70.11		2D-LSTM[78]	78.56

Table 3: Summary of semantic segmentation methods based on deep learning architecture[1]

Name and Reference	Architecture	Targets			Source code	Contribution(s)
		Accuracy	Efficiency	Training		
SegNet autjor name dite hobe then reference[79]	VGG-16 + Decoder	★ ★ ★	★ ★	★	✓	Encoder-decoder
Bayesian SegNet[75]	SegNet	★ ★ ★	★	★	✓	Uncertainty modeling

DeepLab[19][70]	VGG-16/ResNet-101	★★★	★	★	✓	Standalone CRF, atrous convolutions
CRFasRNN[72]	FCN-8s	★	★★	★★★	✓	CRF reformulated as RNN
Dilation[71]	VGG-16	★★★	★	★	✓	Dilated convolutions
ENet[81]	ENet bottleneck	★★	★★★	★	✓	Bottleneck module for efficiency
Multi-scale-CNN-Eigen[21]	Custom	★★★	★	★	✓	Multi-scale sequential refinement
ParseNet[73]	VGG-16	★★★	★	★	✓	Global context feature fusion
ReSeg[80]	VGG-16 + ReNet	★★	★	★	✓	Extension of ReNet to semantic segmentation
2D-LSTM[78]	MDRNN	★★	★★	★	✗	Image context modelling
rCNN[77]	MDRNN	★★★	★★	★	✓	Different input sizes, image context
DAG-RNN[76]	Elman network	★★★	★	★	✓	Graph image structure for context modelling

A brief overview of the several deep learning architecture-based semantic segmentation techniques is given in Table 3. Along with the architecture and important goals like training performance, efficiency, and accuracy where each target is graded from one to three stars (★) depending on how much focus puts the work on it, and a mark (✗) if that issue is not addressed, it also indicates if the source code is publicly available. Each model's contributions are also discussed, including context modeling, uncertainty modeling, and encoder-decoder structures. ReSeg, which extends ReNet for segmentation problems, DeepLab with standalone CRF and atrous convolutions, and SegNet and its variation Bayesian SegNet are notable techniques. For improved context modeling, models such as CRFasRNN and DAG-RNN concentrate on combining Conditional Random Fields (CRFs) and graph topologies.

7. Challenges and Future Opportunities

7.1 Challenges
- The paper "Focal Loss for Dense Object Detection" [58] addresses the obstacle of class imbalance in image processing tasks, particularly in the context of dense object detection and semantic segmentation. The main challenges posed by class imbalance include training inefficiency, loss of discriminative information, model degradation, biased predictions, and evaluation difficulties. To overcome these obstacles, the paper introduces the Focal Loss,

a novel loss function designed to address the extreme foreground-background class imbalance encountered during training of dense detectors. By reshaping the standard cross entropy loss to down-weight the loss assigned to well classified examples, the Focal Loss focuses training on hard examples and prevents easy negatives from overwhelming the detector during training.

- **Limited Annotated Data:** In order to segment images for biomedical applications, the research [59] presents the U-Net architecture, a deep convolutional network. Lack of labeled training data is one of the primary challenges in image processing, particularly in the biomedical domain. The authors offer a training approach that effectively utilizes the existing annotated examples by relying mostly on data augmentation in order to overcome this difficulty. Using training pictures with elastic deformations, the network may learn to be invariant to these transformations without requiring a large amount of annotated data.

- **Context Understanding:** Semantic segmentation accuracy depends on an understanding of context since contextual information is typically provided by surrounding pixels. But balancing computational efficiency with the appropriate integration of local and global context is still a problem.

- **Real-time Inference:** Semantic segmentation in real time is crucial for numerous applications, including augmented reality and driverless vehicles. A major problem is to enable real-time performance on resource-constrained devices by striking a balance between segmentation accuracy and processing efficiency.

- **Challenges of Data Availability in Algorithm Training:** Large volumes of labeled data are needed for some of the better methods. This implies that under certain scenarios, the algorithms won't work because the labeled datasets aren't available. Though the training set size is more likely to be in the thousands for most applications, viable datasets for scene classification generally contain millions to hundreds of millions of training photos. Can deep learning algorithms be designed with fewer examples needed if the domain experts find it difficult or impossible to create very big training sets?

- **Challenges in Assessing Algorithm Generality for General Imagery:** On broad images, the efficacy of top algorithms is still unknown. Frequently, the most effective techniques are tailored to particular circumstances or environments, making their applicability vague. It is imperative that the research community tackle this dilemma.

- **Challenges in Achieving High Accuracy with Limited Computing Resources:** Several of the more advanced techniques involve a significant amount of training on computers that are not always available, such as near supercomputers. That is why a lot of scholars are thinking about the following question: What is the most accuracy possible given a given set of parameters?

- **Contextual Challenges in Accuracy and Segmentation:** While increasing accuracy is a good thing, it's also critical to know what happens when segmentations go wrong. It is not uncommon to run into segmentation issues that weren't covered by the training dataset in

specific circumstances, like driving a car in a city. It would be very helpful to have a very accurate image segmentation. Nevertheless, it's unclear if we have reached that stage yet.

- **Dealing with Varying Scales and Shapes of Objects:** Semantic segmentation involves many obstacles, particularly when dealing with objects of different sizes and shapes. One model cannot fully segment all of the variables in natural settings due to the wide range of sizes and forms of items. It is crucial, but difficult, to capture contextual information at various resolutions since it calls for advanced multi scale feature extraction techniques. Complexity also arises from the need to dynamically modify receptive fields to different sizes and forms using methods such as deformable convolutions. Robust data augmentation procedures are necessary for training models to be resilient to scale fluctuations, but they can be challenging to put into practice. An additional layer of complexity is introduced by using Region Proposal Networks (RPNs) to generate precise item suggestions of various sizes.

- **Managing Overlapping Objects and Occlusions:** Because it might be difficult to discern and segment specific items that partially conceal each other, handling occlusions and overlapping objects in semantic segmentation is a substantial problem. These situations are often difficult for traditional methods to handle, which results in inaccurate segmentation or blending of objects. This problem is addressed by sophisticated approaches that provide various labels for overlapping objects, such as instance segmentation, which distinguishes between instances of the same class. The model's capacity to distinguish obscured objects according to their spatial relationships is improved by methods that use depth information, such as RGB-D datasets. Furthermore, context-aware networks and attention mechanisms assist the model focus on pertinent areas of the image, which aids in distinguishing between overlapping and obscured areas.

7.2 Opportunities

- **Real-time Semantic Segmentation** [60]: For applications like augmented reality, robotics, and autonomous driving, it is critical to create lightweight and effective models that can process and segment pictures in real-time. For these models to function well on platforms with constrained hardware resources, such as mobile and edge devices, they must strike a compromise between computational efficiency and accuracy.

- **Handling Varying Scales and Complex Environments** [61]: Improving the resilience of the model is crucial to manage objects with different scales and intricate backgrounds. This entails creating methods for precisely separating tiny and large items inside a single scene as well as guaranteeing dependable operation in a variety of congested and varied environments—both of which are typical of real-world applications.

- **Integration with Depth and Multi-modal Data** [62]: The accuracy of segmentation can be greatly increased by combining typical RGB data with additional data modalities (such as LiDAR or infrared imaging) and depth information. Richer contextual information can

be obtained by using this multi-modal method, which can aid in differentiating items that share similarities in appearance but differ in depth or thermal signatures.

- **Medical Imaging Applications** [63]: Semantic segmentation models can help with tumor detection and organ segmentation, as well as enhance diagnostic accuracy when tailored and optimized for different medical imaging modalities like MRIs, CT scans, and X-rays. In order to do this, models must be sensitive to the unique qualities and noise patterns found in medical images.

- **Self-supervised and Unsupervised Learning** [64]: Using self-supervised and unsupervised learning techniques, one can democratize access to highperforming segmentation models by reducing dependence on big labeled datasets. These methods lower the expense and labor involved in manual annotation by allowing models to learn from the large amount of unlabeled data.

- **Edge Computing and IoT Applications** [65]: Optimizing segmentation models for constrained computational and memory resources is necessary for deploying them on edge devices for Internet of Things applications. This facilitates applications like smart surveillance and industrial automation by enabling real-time analysis and decision-making in smart cameras, drones, and other linked devices.

- **3D Semantic Segmentation** [66]: It is imperative to develop methods for semantic segmentation in three dimensions for applications such as robots, augmented reality, and autonomous driving. In order to separate objects in three dimensions and provide a more comprehensive picture of the environment, this entails analyzing point clouds or volumetric data.

- **Transfer Learning and Domain Adaptation** [67]: Time and resources can be saved by enhancing model performance by modifying pre-trained models for use in new tasks and domains. Domain adaption techniques guarantee that models generalize effectively across many contexts and situations, whereas transfer learning enables models trained on broad, generic datasets to be fine-tuned for specific applications.

- **Interactive and User-Guided Segmentation** [68]: Creating interactive tools that let users direct the process of segmentation can improve precision and usefulness. With the use of these technologies, users can offer suggestions or edits throughout the segmentation process, which improves efficiency and allows the process to be customized for particular needs, particularly in creative industries like graphic design and video editing.

8. Conclusion

Technological advances in computer vision, including autonomous driving, medical imaging, and augmented reality, have been fueled by the development of semantic segmentation as a key approach. The transition from conventional techniques to deep learning-based methods for semantic segmentation has been emphasized in this review, highlighting the revolutionary role of convolutional neural networks (CNNs). By using hierarchical feature extraction and contextual information, methods such as Fully Convolutional Networks (FCNs), U-Net, and DeepLab have

broken previous records. But a number of obstacles still exist in spite of tremendous development. Managing objects with different forms and sizes is still a major problem; it is frequently solved by using multi-scale and pyramid pooling techniques, but it is still challenging to obtain reliable results. Similar challenges arise from occlusions and overlapping objects, which necessitate more intricate modeling of context and spatial relationships. Although it needs more research and standardization, the integration of multi-modal data—including thermal and depth information—offers great opportunities to improve segmentation performance.

Reducing dependence on large-scale labeled datasets by using weakly-supervised, semisupervised, and unsupervised learning strategies is another exciting avenue. These methods are essential for democratizing the availability of strong segmentation models, particularly in fields where annotated data is hard to come by or prohibitively expensive. Furthermore, the development of real-time segmentation techniques is critical for applications like robotics and mobile computers that require fast and effective processing. Future studies should concentrate on strengthening segmentation models' resistance to hostile attacks and domain shifts in order to guarantee dependability in a variety of uncertain real-world scenarios. Furthermore, more comprehensive and adaptable systems may result from improving model interpretability and combining segmentation with other computer vision tasks.

In summary, even though semantic segmentation has made significant progress, more study is still needed to fully realize the technology's potential by addressing current issues and opening up new avenues. By addressing these issues, the field will be able to develop segmentation models that are more precise, effective, and flexible, which will ultimately increase the range of applications and enhance the standard of automated visual understanding.

References

1. A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," 2017. [Online]. Available: https://arxiv.org/abs/1704.06857

2. H. Yu, Z. Yang, L. Tan, Y. Wang, W. Sun, M. Sun, and Y. Tang, "Methods and datasets on semantic segmentation: A review," Neurocomput., vol. 304, no. C, p. 82–103, aug 2018. [Online]. Available: https://doi.org/10.1016/j.neucom.2018. 03.037

3. B. Emek Soylu, M. S. Guzel, G. E. Bostanci, F. Ekinci, T. Asuroglu, and K. Acici, "Deep-learning-based approaches for semantic segmentation of natural scene images: A review," Electronics, vol. 12, no. 12, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/12/2730

4. Y. Guo, Y. Liu, T. Georgiou, and M. Lew, "A review of semantic segmentation using deep neural networks," International Journal of Multimedia Information Retrieval, vol. 7, 06 2018.

5. M. Oberweger, P. Wohlhart, and V. Lepetit, "Hands deep in deep learning for hand pose estimation," 2016. [Online]. Available: https://arxiv.org/abs/1502.06807

6. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," 2016. [Online]. Available: https://arxiv.org/abs/1604.01685.

7. A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3354–3361.

8. A. Ess, T. Mueller, H. Grabner, and L. Van Gool, "Segmentation-based urban traffic scene understanding." in BMVC, vol. 1. Citeseer, 2009, p. 2.

9. Y. Yoon, H.-G. Jeon, D. Yoo, J.-Y. Lee, and I. So Kweon, "Learning a deep convolutional network for light-field image super-resolution," in Proceedings of the IEEE international conference on computer vision workshops, 2015, pp. 24–32.

10. D. G. Lowe, "Object recognition from local scale-invariant features," Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, pp. 1150–1157 vol.2, 1999. [Online]. Available: https://api.semanticscholar.org/ CorpusID:5258236

11. N. Dalal and 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.

12. N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 601–608, 2011. [Online]. Available: https://api.semanticscholar.org/CorpusID:13993169

13. S. Gupta, P. Arbelaez, and J. Malik, "Perceptual organization and recognition ´ of indoor scenes from rgb-d images," 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 564–571, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:12061055

14. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Commun. ACM, vol. 60, no. 6, p. 84–90, may 2017. [Online]. Available: https://doi.org/10.1145/3065386

15. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1409.1556

16. G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," Neural Comput., vol. 18, no. 7, p. 1527–1554, jul 2006. [Online]. Available: https://doi.org/10.1162/neco.2006.18.7.1527

17. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, pp. 436–444, 2015.

18. C. Farabet, C. Couprie, L. Najman, and Y. Lecun, "Learning hierarchical features for scene labeling," IEEE transactions on pattern analysis and machine intelligence, vol. 35, pp. 1915–1929, 08 2013.

19. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," 2016. [Online]. Available: https://arxiv.org/abs/1412.7062

20. R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," 2014. [Online]. Available: https://arxiv.org/abs/1311.2524

21. D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," 2015. [Online]. Available: https://arxiv.org/abs/1411.4734

22. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278– 2324, 1998.

23. J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," 07 2006, pp. 1–15.

24. P. Kohli, L. Ladicky, and P. Torr, "Robust higher order potentials for enforcing ´ label consistency," International Journal of Computer Vision, vol. 82, pp. 302– 324, 05 2009.

25. L. Ladicky, C. Russell, P. Kohli, and P. Torr, "Associative hierarchical crfs for object class image segmentation," 11 2009, pp. 739 – 746.

26. J. Verbeek and B. Triggs, "Region classification with markov field aspect models," 06 2007.

27. M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014.

28. J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" 2014. [Online]. Available: https: //arxiv.org/abs/1411.1792

29. S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: when to warp?" 2016. [Online]. Available: https://arxiv.org/abs/1609.08764

30. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015. [Online]. Available: https: //arxiv.org/abs/1505.04597

31. J. Huang, G. Liu, and B. Wang, "Semantic Segmentation under a Complex Background for Machine Vision Detection Based on Modified UPerNet with Component Analysis Modules," Mathematical Problems in Engineering, vol. 2020, pp. 1–13, September 2020. [Online]. Available: https://ideas.repec.org/a/ hin/jnlmpe/6903130.html

32. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

33. J. Pont-Tuset, P. Arbelaez, J. T.Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 1, p. 128–140, Jan. 2017. [Online]. Available: http://dx.doi.org/10.1109/TPAMI. 2016.2537320

34. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1409.1556

35. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," 2014. [Online]. Available: https://arxiv.org/abs/1409.4842

36. B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

37. H. Caesar, J. Uijlings, and V. Ferrari, "Coco-stuff: Thing and stuff classes in context," 2018. [Online]. Available: https://arxiv.org/abs/1612.03716

38. M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," International journal of computer vision, vol. 111, pp. 98–136, 2015.

39. R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 891–898.

40. X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille, "Detect what you can: Detecting and representing objects using holistic models and body parts," 2014. [Online]. Available: https://arxiv.org/abs/1406.2031

41. N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in European Conference on Computer Vision, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID: 545361

42. S. Song, S. P. Lichtenberg, and J. Xiao, "Sun rgb-d: A rgb-d scene understanding benchmark suite," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.

43. J. Xiao, A. Owens, and A. Torralba, "Sun3d: A database of big spaces reconstructed using sfm and object labels," in Computer Vision (ICCV), 2013 IEEE International Conference on. IEEE, 2013, pp. 1625–1632.

44. B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours ´ from inverse detectors," in 2011 International Conference on Computer Vision, 2011, pp. 991–998.

45. G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

46. F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," 2020. [Online]. Available: https://arxiv.org/abs/1805.04687

47. G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," Pattern Recognition Letters, vol. 30, no. 2, pp. 88–97, Jan. 2009.

48. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3213–3223.

49. A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari, "Learning object class detectors from weakly annotated video," in 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3282–3289.

50. S. Bell, P. Upchurch, N. Snavely, and K. Bala, "Material recognition in the wild with the materials in context database," 2015. [Online]. Available: https://arxiv.org/abs/1412.0623

51. A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3354–3361.

52. A. Kapitanov, K. Kvanchiani, and K. Sofia, "Easyportrait - face parsing and portrait segmentation dataset," arXiv preprint arXiv:2304.13509, 2023.

53. F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. SorkineHornung, "A benchmark dataset and evaluation methodology for video object segmentation," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 724–732.

54. C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 5, pp. 978–994, 2011.

55. A. Richtsfeld, "The object segmentation database(osd)," 2012.

56. S. Gould, R. Fulton, and D. Koller, "Decomposing a scene into geometric and semantically consistent regions," 2009 IEEE 12th International Conference on Computer Vision, pp. 1–8, 2009. [Online]. Available: https://api.semanticscholar. org/CorpusID:17448963

57. K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 1817–1824.

58. T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object ´ detection," 2018.

59. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.

60. H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "Icnet for real-time semantic segmentation on high-resolution images," 2018, computer Vision and Pattern Recognition. Covers image processing, computer vision,

pattern recognition, and scene understanding. Roughly includes material in ACM Subject Classes I.2.10, I.4, and I.5.

61. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 834–848, 2018.

62. C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture," in Asian Conference on Computer Vision, 2016. [Online]. Available: https: //api.semanticscholar.org/CorpusID:178063

63. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.

64. K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," 2020.

65. N. D. Lane, S. Bhattacharya, A. Mathur, P. Georgiev, C. Forlivesi, and F. Kawsar, "Squeezing deep learning into mobile and embedded devices," IEEE Pervasive Computing, vol. 16, pp. 82–88, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:27704761

66. C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," 2017.

67. Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. V. Gool, "Domain adaptive faster r-cnn for object detection in the wild," 2018.

68. J. Huang, G. Liu, and B. He, "Fast semantic segmentation method for machine vision inspection based on a fewer-parameters atrous convolution neural network," *PLOS ONE*, vol. 16, no. 2, pp. e0246093, 2021. DOI: 10.1371/journal.pone.0246093.

69. J. Huang, G. Liu, and B. Wang, "Semantic segmentation under a complex background for machine vision detection based on modified UPerNet with component analysis modules," *Mathematical Problems in Engineering*, vol. 2020, pp. 1-13, 2020. DOI: 10.1155/2020/6903130.

70. Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," arXiv preprint arXiv:1606.00915, 2016.

71. F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," arXiv preprint arXiv:1511.07122, 2015.

72. S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1529–1537.

73. W. Liu, A. Rabinovich, and A. C. Berg, "Parsenet: Looking wider to see better," arXiv preprint arXiv:1506.04579, 2015.

74. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

75. A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian segnet: Model uncertainty in deep convolutional encoderdecoder architectures for scene understanding," arXiv preprint arXiv:1511.02680, 2015.

76. B. Shuai, Z. Zuo, G. Wang, and B. Wang, "Dag-recurrent neural networks for scene labeling," CoRR, vol. abs/1509.00552, 2015. [Online]. Available: http://arxiv.org/abs/1509.00552

77. P. H. Pinheiro and R. Collobert, "Recurrent convolutional neural networks for scene labeling." in ICML, 2014, pp. 82–90.

78. W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki, "Scene labeling with lstm recurrent neural networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3547–3555.

79. V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," arXiv preprint arXiv:1511.00561, 2015

80. F. Visin, M. Ciccone, A. Romero, K. Kastner, K. Cho, Y. Bengio, M. Matteucci, and A. Courville, "Reseg: A recurrent neural network-based model for semantic segmentation," in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2016.

81. A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "Enet: A deep neural network architecture for real-time semantic segmentation," arXiv preprint arXiv:1606.02147, 201

82. L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," arXiv, vol. 1606.00915, 2017. [Online]. Available: https://arxiv.org/abs/1606.00915