1. ***Abstract:***

Semantic segmentation is a fundamental task in computer vision, playing a crucial role in scene understanding and object recognition. Semantic segmentation's main goal is to give each pixel in an image a unique name, dividing the image into sections that have semantic significance. Through this approach, a greater understanding of the visual world is made possible, allowing machines to study and interpret images at a highly abstract level. This work offers an overview of deep learning techniques for semantic segmentation with applications in several fields.

***This section will be changed:***

*First, we go through the necessary underlying ideas and the terminology used in this discipline. The primary datasets and challenges are then made public to assist researchers in selecting the ones that best meet their objectives and needs. After that, a review of current approaches is conducted, emphasizing their contributions and field importance. Ultimately, a commentary on the findings is included after the quantitative results for the methodologies and datasets under description are provided. Lastly, we highlight many exciting upcoming projects and offer our assessments on the state of semantic segmentation with deep learning methodologies.*

**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. Among those uses are, to mention a few, human-machine interaction [5], autonomous driving [6] [7] [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.
* Draw attention to the issues that need to be resolved by upcoming researchers.
* We offer an overview of available datasets that could be helpful for deep learning-based semantic segmentation projects.
* a comprehensive analysis of performance that collects numerical measurements for things like memory, execution time, and precision.

***This section will be changed:***

*The rest of this paper is structured as follows. First, the problem of semantic segmentation is introduced in Section 2, along with notation and standards that are frequently seen in the literature. Additional background ideas are also covered, such typical deep neural networks. Section 3 then goes on to discuss benchmarks, problems, and datasets that are currently available. Section 4 examines current approaches in a bottom-up complexity hierarchy according to their respective contributions. Rather of carrying out a quantitative evaluation, this section concentrates on outlining the theory and salient features of various approaches. In conclusion, Section 5 provides a succinct analysis of the techniques that have been provided, drawing on their quantitative outcomes using the previously mentioned datasets. Furthermore, directions for future research are also outlined.*

1. ***Background and Preliminaries:***
2. **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.

1. **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.

1. **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).

1. **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 pre-trained 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.

1. **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].

1. **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 low-level 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.

1. ***Challenges:***

Some common challenges in semantic segmentation are:

* Dealing with varying scales and shapes of objects.
* Handling occlusions and overlapping objects.
* Distinguishing between similar classes or fine-grained categories.
* Achieving high accuracy and efficiency in segmentation models.

1. ***Popular Deep Network Architectures:***

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 DeepLab-v2, 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.

1. DeepLab-v2:
2. **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 [30]. 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.

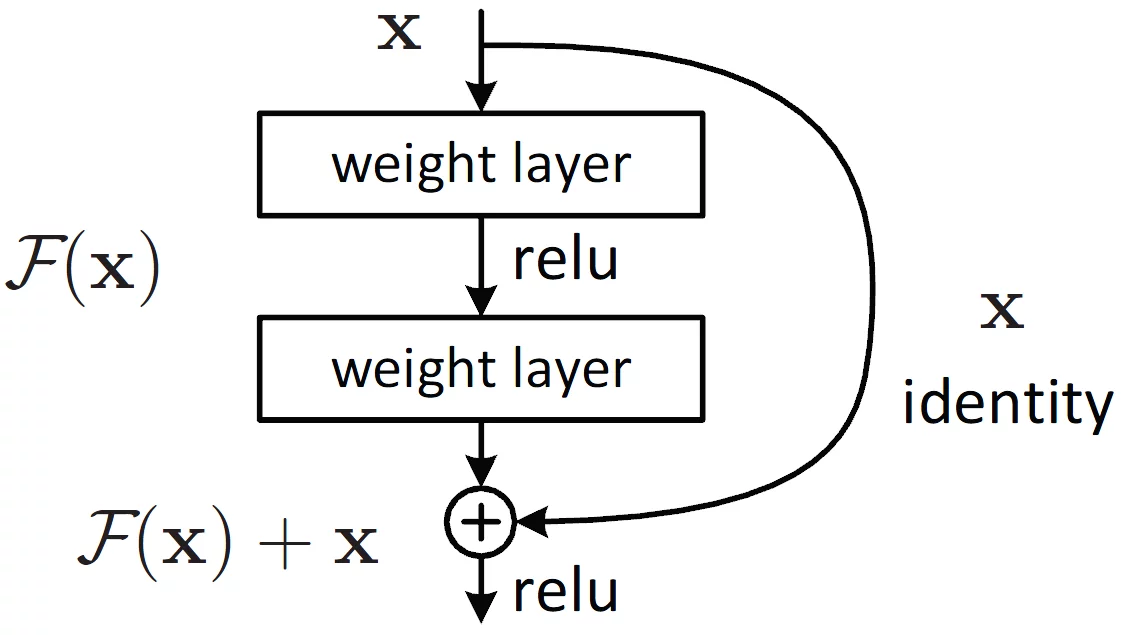


Figure: Residual learning: a building block [30].

1. **VGG-16:**

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-16 design [32]. 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-16 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-16. In summary, the authors provide insightful research into the architecture and functionality of extremely deep ConvNets, notably VGG-16, and shed light on these topics. Their findings are particularly relevant for the field of large-scale image recognition.

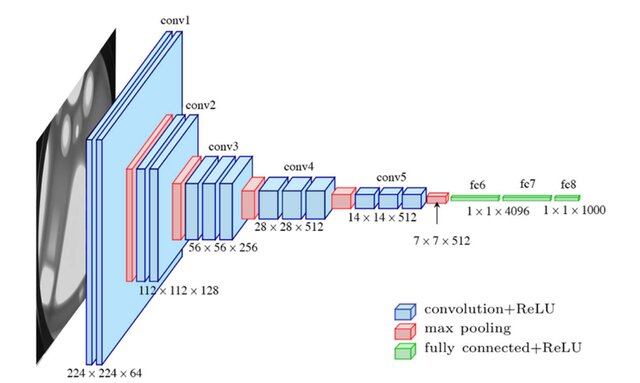


Figure: Typical architecture of the VGG model [32].

1. **MCG:**

A unified and flexible method for object candidate creation and picture segmentation is presented: Multiscale Combinatorial Grouping (MCG) [31]. MCG navigates a combinatorial space of multiscale areas to provide correct item candidates by utilizing effective normalized cutting methods, hierarchical segmentation, and grouping procedures. The approach shows state-of-the-art results for hierarchical segmentation and contour detection on the BSDS500 dataset. Specifically, using the PASCAL 2012 dataset, MCG significantly outperforms other approaches in terms of instance-level and class-level quality. In addition to introducing a quicker single-scale version of MCG and showcasing significant gains in multiscale segmentation when tested on the PASCAL dataset, the research illustrates the complementarity of MCG with other approaches. MCG is an effective technique for object recognition in photos because of its versatility and adaptation to certain applications.

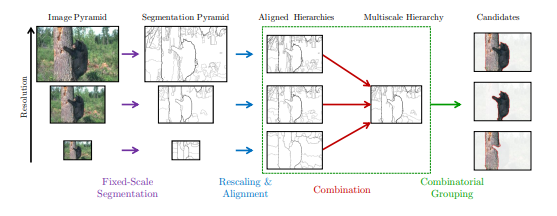


Figure: Multiscale Combinatorial Grouping [31]

1. **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.

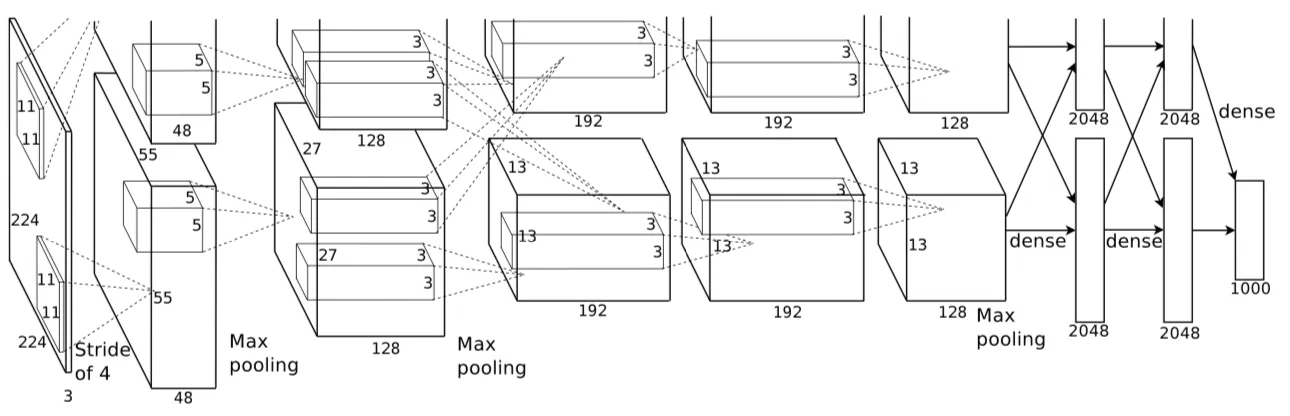


Figure: Illustration of AlexNet’s architecture [14].

1. **GoogLeNet:**

In the paper "Going Deeper with Convolutions," the Inception architecture is presented using the GoogLeNet model as an example [33]. 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.

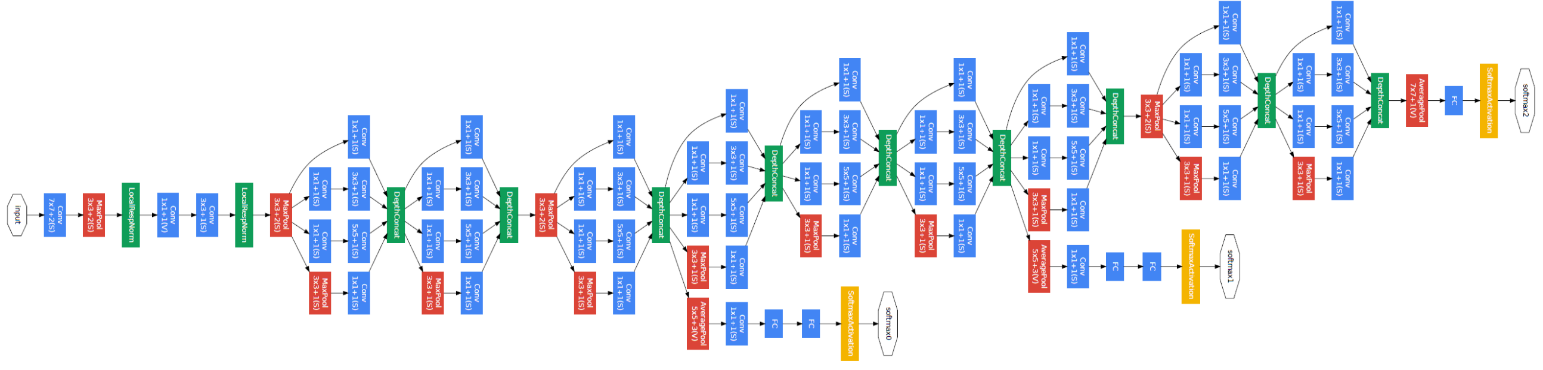


Figure: GoogLeNet network with all the bells and whistles [33].

1. ***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.

1. ***La la land***
2. ***References:***
3. A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. G. Rodr ́ıguez, “A review on deep learning techniques applied to semantic segmentation,” CoRR, vol. abs/1704.06857, 2017.
4. H. Yu, Z. Yang, L. Tan, Y. Wang, W. Sun, M. Sun, and Y. Tang, “Methods and datasets on semantic segmentation: A review,” Neurocomputing, vol. 304, pp. 82–103, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231218304077
5. 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>
6. Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, “A review of semantic segmentation using deep neural networks,” International Journal of Multimedia Information Retrieval, vol. 7, no. 2, pp. 87–93, Jun 2018
7. M. Oberweger, P. Wohlhart, and V. Lepetit, “Hands deep in deep learning for hand pose estimation,” arXiv preprint arXiv:1502.06807, 2015.
8. 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 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223.
9. 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, June 2012, pp. 3354–3361
10. A. Ess, T. Muller, H. Grabner, and L. J. Van Gool, “Segmentation- ¨ based urban traffic scene understanding.” in BMVC, vol. 1, 2009, p. 2.
11. Y. Yoon, H.-G. Jeon, D. Yoo, J.-Y. Lee, and I. So Kweon, “Learning a deep convolutional network for light-field image superresolution,” in Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015, pp. 24–32.
12. D.G. Lowe, Object recognition from local scale-invariant features, in: Proceedings of the IEEE International Conference on Computer Vision, 1999, pp. 1150–1157
13. N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 886–893.
14. N. Silberman, R. Fergus, Indoor scene segmentation using a structured light sensor, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2011, pp. 601–608
15. S. Gupta, P. Arbelaez, J. Malik, Perceptual organization and recognition of indoor scenes from RGB-D images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 564–571
16. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Advances in neural information processing systems, 2012, pp. 1097–1105
17. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556(2014).
18. G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
19. Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444
20. C. Farabet, C. Couprie, L. Najman, Y. LeCun, Learning hierarchical features for scene labeling, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1915–1929.
21. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Semantic image segmentation with deep convolutional nets and fully connected CRFs, arXiv:1412.7062(2014).
22. R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.
23. D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2650–2658.
24. Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324
25. J. Shotton, J. Winn, C. Rother, A. Criminisi, Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation, in: Proceedings of the European conference on computer vision, 2006, pp. 1–15.
26. P. Kohli, P.H. Torr, Robust higher order potentials for enforcing label consistency, Int. J. Comput. Vis. 82 (3) (2009) 302–324.
27. L. Ladick, C. Russell, P. Kohli, P.H. Torr, Associative hierarchical CRFs for object class image segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2009, pp. 739–746.
28. J. Verbeek, B. Triggs, Region classification with Markov field aspect models, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
29. 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, 2014, pp. 1717–1724.
30. J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?” in Advances in neural information processing systems, 2014, pp. 3320–3328.
31. S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, “Understanding data augmentation for classification: when to warp?” CoRR, vol. abs/1609.08764, 2016. [Online]. Available: <http://arxiv.org/abs/1609.08764>
32. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778
33. Arbeláez, P.; Pont-Tuset, J.; Barron, J.T.; Marques, F.; Malik, J. Multiscale combinatorial grouping. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 328–335.
34. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv 2014, arXiv: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,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.