Deep Semantic Segmentation for Automated Driving: Taxonomy, Roadmap and Challenges

Mennatullah Siam, Sara Elkerdawy, Martin Jagersand
University of Alberta, Canada
Email: mennatul@ualberta.ca

Senthil Yogamani
Valeo Vision Systems, Ireland
Email: senthil.yogamani@valeo.com

arXiv:1707.02432v1 [stat.ML] 8 Jul 2017

***Abstract*—Few years ago, semantic segmentation was seen as a challenging problem. Due to advancements in Deep learning, reasonably accurate solutions are now possible. In this paper, we explore the semantic segmentation problem from the perspective of automated driving and investigate how it fits into an overall system. Most of the current semantic segmentation algorithms are generic and does not incorporate structure and goal orientation for automated driving. It also does not exploit the other cues available for an automated driving system. A survey of the present state of the art algorithms in semantic segmentation is presented, and a more general problem is formulated. The practical challenges involved in deploying it into a safety system which needs high level of accuracy is also discussed. Different alternatives instead of semantic segmentation is then presented and discussed. Finally, a comparative evaluation of semantic segmentation architectures is shown in terms of accuracy and speed.**

I. Introduction

Semantic image segmentation has witnessed tremendous progress recently with deep learning. Semantic segmentation is targeted towards partitioning of the image into semantically meaningful parts with various applications for that. It has been used in robotics [1][2][3][4], medical applications [5][6], augmented reality [7], and most prominently automated driving [8][9][10][11].

Automated driving is another hot topic in the literature that is advancing recently with the rapid growth in deep learning. The goal to create an automated car started since 1989 with the work in [12] that used single hidden layer network. However, the limitations in neural networks at that time did not allow its progress further on. Recently with deep learning and advances in GPU technologies, different works on automated driving emerged.

Two main paradigms for automated driving emerged: **(1)** The mediated perception approach which parses the whole scene and uses this information for the control decision increasing the complexity and the cost of the system. **(2)** The behavior reflex paradigm that relies more on end-to-end learning to map direct sensory input to driving decision which is an ill-posed problem due to the many possible ambiguous decisions, such as the work in [13][14]. However, in [15] an intermediate approach was suggested that learns affordance indicators for the driving scene. These indicators can then feedback on a simple controller for the final driving decision. The previous work on automated driving pose the important question of whether the solution for automated driving need semantic segmentation module or not?

A related survey in [16] on semantic segmentation literature is presented. However it is not addressing the specific application of automated driving. This paper tries to address this gap by reviewing the work on semantic segmentation in the context of automated driving. The paper will address the above question on what is the importance of semantic segmentation in automated driving and reviews alternative approaches. The paper is organized as follows, section II covers the literature work on deep semantic segmentation in general. Followed by section III that focuses on the problem of automated driving and how can semantic segmentation be used in it. Then section IV presents the common challenges that face automated driving applications. Then alternative approaches are discussed including end-to-end learning and multi-task learning in section V. A comparative evaluation of semantic segmentation architectures is presented in VI. Finally sectionVII summarizes and presents the conclusions.

II. Deep Semantic Segmentation Taxonomy

In this section the categories of deep semantic segmentation are discussed. Different work under these categories are reviewed in further details with discussion of their limitations if any and future directions. The literature work in semantic segmentation is categorized into four subcategories: **(1)** Classical Methods. **(2)** Fully Convolutional Networks. **(3)** Structured Models. **(4)** Spatio-Temporal Models. The first category reviews the classical approaches before the emergence of deep learning. The second category is about the main body of work on semantic segmentation using deep learning. The third category reviews the work that tries to utilize structure in the problem of semantic segmentation. Thus following the assumption that neighboring pixel labels should be coherent. Then the fourth category exploits the temporal information that is present in videos. Table I shows the detailed taxonomy of semantic segmentation approaches.

A. *Classical methods*

Semantic segmentation was seen as a difficult problem five years ago with no reasonable accuracy. The main approaches used in semantic segmentation was based on random forest classifier or conditional random fields. In [17] decision forests were used, where each tree was trained on random subset

TABLE I: Taxonomy of Semantic Segmentation Architectures

Classical Methods	**Fully Convolutional Networks**	**Structured Models**	**Spatio-Temporal Models**
Decision Forests [17][18]	Patchwise Training [19][20][21]	Merged with CRFs [22][23][24]	Clockworks Net [25]
CRFs [26][27]	Pixelwise Training [28][29][30]	Using RNNs [31]	Using RNNs [32][33][34]
Boosting [26][35]	Multiscale [36][19][29][37][38][39]		

of the training data. These methods implicitly cluster the pixels while explicitly classifying the patch category. In [18] a randomized decision forest was also used however instead of using appearance based features, motion and structure features were used. These features include surface orientation, height above camera, and track density where faster moving objects have sparser tracks than static objects. However, these techniques rely on hand crafted features and perform pixel-wise classification independently without utilizing the structure in the data.

On the other hand conditional random fields(CRF) were proven to be a good approach for structured prediction problems. In [26][27] segmentation is formulated as CRF problem. The energy function used in CRF formulation usually contains unary potential and pairwise potential. The unary potential gives a probability of whether the pixel belongs to a certain class. While pairwise potential which is also referred to as smoothness term ensures label consistency among connected pixels. Boosting is another method that can be used to classify pixels instead of random decision forests. It is based on combining multiple weak classifiers that are based on some shape filter responses, as in [26][35]. However the progress in classical methods was always bounded by the performance of the hand crafted features used. But that was overcome with deep learning as will be discussed in the following sections.

B. Fully Convolutional Networks(FCN)

The area of semantic segmentation using convolutional neural networks witnessed tremendous progress recently. There were mainly three subcategories of the work that developed. The first [19][20][21] used patch-wise training to yield the final classification. In[19] an image is fed into a Laplacian pyramid, each scale is forwarded through a 3-stage network to extract hierarchical features and patch-wise classification is used. The output is post processed with a graph based classical segmentation method. In [21] a deep network was used for the final pixel-wise classification to alleviate any post processing needed. However, it still utilized patch-wise training.

The second subcategory [28][29][30] was focused on end-to-end learning of pixel-wise classification. It started with the work in [28] that developed fully convolutional networks(FCN). The network learned heatmaps that was then upsampled with-in the network using deconvolution to get dense predictions. Unlike patch-wise training methods this method uses the full image to infer dense predictions. In [29] a deeper deconvolution network was developed, in which stacked deconvolution and unpooling layers are used. In Segnet [30] a similar approach was used where an encoder-decoder architecture was deployed. The decoder network upsampled the feature maps by keeping the maxpooling indices

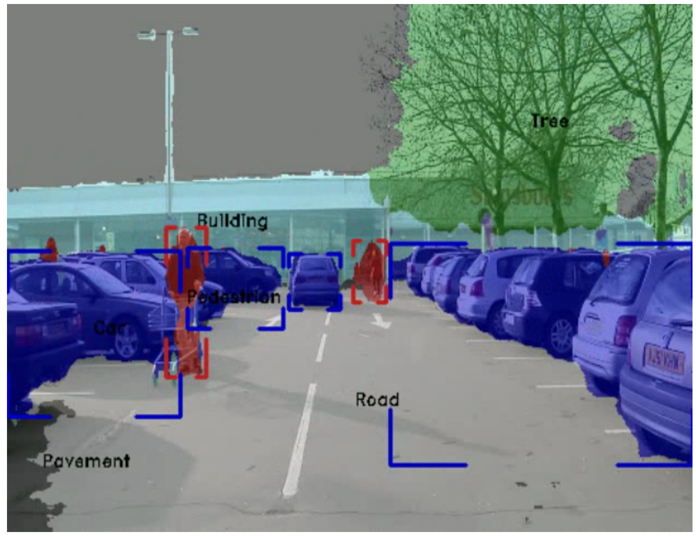

Fig. 1: Segmentation of an automotive scene

from the corresponding encoder layer. In Figure 1 an example of the semantic segmentation output of segnet applied in an automated driving setting is shown.

Finally, the work in [36][19][29][37][38][39] focused on multiscale semantic segmentation. Initially in [19] the scale issue was addressed by introducing multiple rescaled versions of the image to the network. However with the emergence of end-to-end learning, the skip-net architecture in [28] merged heatmaps from different resolutions. However, since these architectures rely on downsampling the image, loss of resolution can hurt the final prediction. The work in [39] proposed a u-shaped architecture network where feature maps from different initial layers are upsampled and concatenated for the next layers. Another work in [36] introduced dilated convolutions, which expanded the receptive field without losing resolution based on the dilation factor. Thus it provided a better solution for handling multiple scales. Finally the recent work in [37] provided a better way for handling scale. It uses attention models that provides a mean to focus on the most relevant features with-in the image. This attention model is able to learn a weighting map that weighs feature maps pixel-by-pixel from different scales.

C. Structured Models

The previous approaches in fully convolutional networks do not utilize the structure in the data. Thus, recent work was directed towards using the prior structure in the data. Specifically in automotive scenes prior structure can be exploited for better segmentation. There are two main categories for solving the structured prediction problem. Some work was based on conditional random field formulation, such as in [22][23][24]. In [22] condition random fields is used as a post processing step after the segmentation network. In [23] condition random fields are also used as post processing to a dilated convolution network to take contextual information into consideration. Finally, in [24] the mean field inference algorithm that is used within CRF formulation was formulated as a recurrent network.

Other work was based on recurrent neural network, such as in [31]. It introduced a different formulation for solving the structured prediction problem. A Recurrent layer is used to sweep the image horizontally and vertically, which ensures the usage of contextual information for a better segmentation.

D. Spatio-Temporal Models

All the discussed work was focused on still image segmentation. Recently some approaches emerged for video semantic segmentation that utilized temporal information [25][32] [33][34]. In [25] introduced clockworks which are clock signals that control the learning of different layers with different rates. In [32] spatio temporal FCN is introduced by using a layer grid of Long Short term memory models (LSTMs). However conventional LSTMs do not utilize the spatial coherence and would ned up with more parameters to learn.

In a recent work [33] convolutional gated recurrent networks was used to learn temporal information to leverage the semantic segmentation of videos. The recurrent unit used in this work was convolutional which enables it to learn both spatial and temporal information with less number of parameters. Thus, it was easier to train and memory efficient. The work in [34] combined the power of both convolutional gated architectures and spatial transformers for leveraging video semantic segmentation.

III. Deep Semantic Segmentation in Automated Driving

A. Problem Structure

Semantic segmentation for automated driving has many a priori constraints relative to a general version. In this section, we discuss the various aspects which brings a simplifying structure to the problem.

1) Scene Structure: Prior information could simplify model complexity greatly. There are different types of prior information that can be used. Spatial priors such as the fact that lanes lie on a ground plane, or that road segmented is mostly in the bottom half of the images. Geometric priors on the shapes of objects, for examples lanes are thick lines that are all converging into a vanishing point. Color priors such as the color of traffic lights or white lanes. Finally, Location priors, for example the lane, road or buildings locations based on high definition maps or aerial maps.

2) Multi-camera Structure: Typically automotive setting uses a multi-camera network. current systems have five and it is increasing to ten cameras deployed. There is spatial structure across the four surround view for example which can be modeled. As shown in Figure 2 this surround view can help to enforce prior information on the segmentation of frames from different cameras.

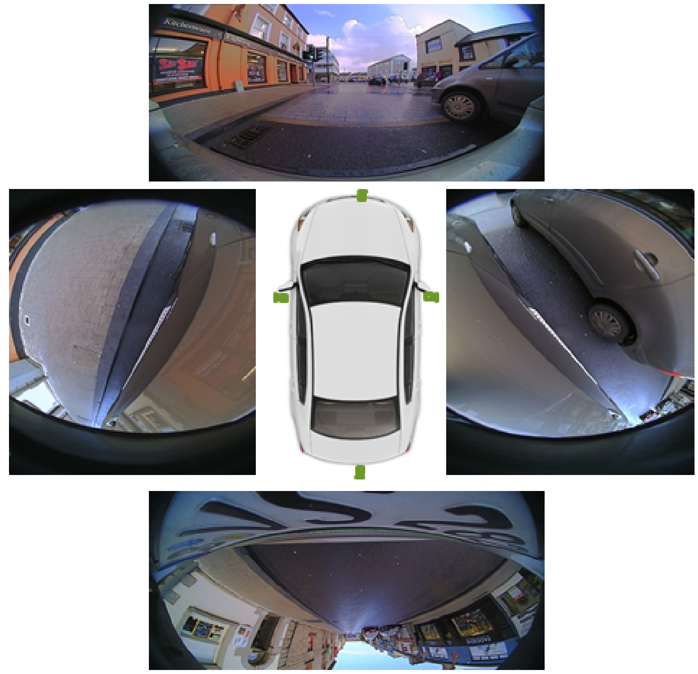

Fig. 2: Surround view camera cocoon

B. Dense High Definition(HD) maps

High accuracy of Object detection is very difficult to achieve and HD maps is an important cue to improve it. There are two types of HD maps: **(1)** Dense Semantic Point Cloud Maps and **(2)** Landmark based Maps. The former is the dense version where the entire scene is modeled by 3D point cloud with semantics. Google and TomTom adopt this strategy. As this is high end, it is expensive to cover the entire world and needs large memory requirements. If there is good alignment, all the static objects (road, lanes, curb, traffic signs) are obtained from the map already and dynamic objects are obtained through background subtraction. TomTom RoadDNA[40] provides an interface to align various sensors like Lidar, Cameras, and others. Figure 3 illustrates this where the pre-mapped semantic point cloud on the right is aligned with an image at runtime with other dynamic objects. They have mapped majority of European cities and they provide an accuracy of 10 cm assuming a coarse location from GPS. Landmark based maps are based on semantic objects instead of generic 3D point clouds. Thus it works primarily for camera data. Mobileye and HERE follow this strategy. This can be viewed as a simple form of the 3D point cloud where a subset of objects is mapped using a 2D map. In this method, object detection is leveraged to provide a HD map and the accuracy is improved by aggregating over several observations from different cars.

In case of a good localization, HD maps can be treated as a dominant cue and semantic segmentation algorithm greatly simplifies to be a refinement algorithm of priors obtained by

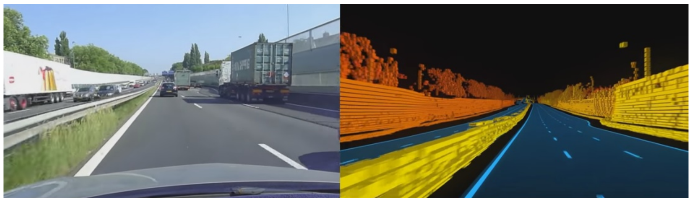

Fig. 3: Example of High Definition(HD) map from TomTom RoadDNA (Reproduced with permission of the copyright owner)

HD maps. In Figure 3, the semantic point cloud alignment provides an accurate semantic segmentation for static objects. Note that it does not cover distant objects like sky. This would need a good confidence measure for localization accuracy, typically some kind of re-projection error is used. HD maps can also be used for validation or post-processing the semantic segmentation to eliminate false positives. For this, both landmark maps and semantic point cloud maps could be used.

C. Localization

Localization or depth estimation is very critical for automated driving. Having image semantics without localization is not very useful.

1) Depth using Structure from Motion(SFM): The straight forward approach to augment localization is to have a parallel independent path for computing dense depth using a standard method like structure from motion (SFM) and then augmenting the depth to localize the objects. Dense depth is computed to understand the spatial geometry of the scene. Accurate Depth should help in semantic segmentation and could be passed on as an extra channel. However, SFM estimates are quite noisy and also the algorithm variations over time could affect the training of the network. But in [18] some cues from the noisy point-cloud was inferred to act as features for segmentation. The cues proposed were: height above the camera, distance to the camera path, projected surface orientation, feature track density, and residual reconstruction error. The work in[4] proposed a way of jointly estimating the semantic segmentation and structure from motion in a conditional random field formulation.

2) LIDAR sensors: LIDAR sensors provide very accurate depth estimation. However, they are not dense in the image lattice. This leads to problems in learning a dense convolutional neural networks features. But it can provide a way to fuse semantic segmentation with depth information in a probabilistic framework. In [41] the method fused a map built using elastic fusion [42] and semantic segmentation from convolutional neural networks termed as semantic fusion. The class probabilities were maintained for each pixel in the map and updated in an incrementally bayesian method. The images used in this work wer from RGB-D cameras, but it provided potential use of depth from LIDAR sensors. Generally, this is a good research problem to be pursued as LIDAR is becoming a standard sensor for next generation automated driving systems.

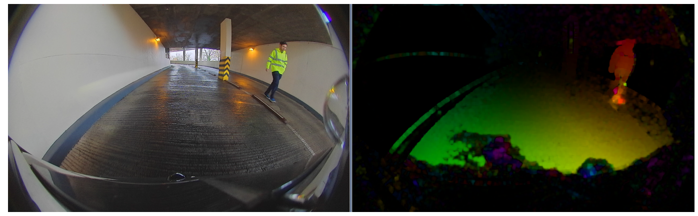

Fig. 4: SFM point cloud

3) Joint In-the-Network Localization: There exists promising algorithms on using convolutional neural networks to estimate structure and camera motion. A recent work in [43] proposed depth and motion network for learning monocular stereo. As far as the authors are aware, there is no work on jointly estimating depth and semantics with in a network. This can synergize and potentially aid in the estimation of each other. It can also be trained simultaneously in an end-to-end fashion. This problem can be of potential future direction for further research.

IV. Challenges

A. Computational Bound in Embedded Systems

On a high end automotive platform like Nvidia Tegra X1, Enet achieves around 4 fps and the proposed algorithm in [44] achieves around 3 fps at a slightly higher accuracy. This benchmark is for a 720P resolution and the current generation cameras are around 2 Megapixel which will reduce the runtime by another factor of 3X. This is clearly not acceptable for a commercial solution to handle high speed objects for highway driving. Reducing the resolution to VGA (640x480) brings it close to 10 fps which is still not reasonable and reducing resolution degrades accuracy and misses small objects which might be critical. Additionally, for full surround view sensing at least 5 cameras need to be employed which adds in another factor of 5X. However the industry is moving towards customer hardware accelerators for CNNs which will enable the possibility of doing multi-camera semantic segmentation at a higher frame rate, Nvidia Xavier for instance supports 30 tera-ops. There is also active research on efficient network design which will improve the performance.

B. Availability of large annotated datasets

The real potential of deep learning was unveiled because of the large dataset Imagenet[45]. The functional complexity of semantic segmentation is much higher and it requires a significantly larger dataset relative to Imagenet. Annotation for semantic segmentation is time consuming, typically it can take around an hour for annotating a single image. It can be speeded up by the availability of other cues like LIDAR or exploiting temporal propagation and bootstrapping classifier.

The popular semantic segmentation automotive datasets are camvid [18] and the more recent cityscapes [11]. The latter has a size of 5000 annotation frames which is relatively small. The algorithms trained on this dataset do not generalize well to data tested on other cities and with unseen objects like tunnels.

To compensate for that, synthetic datasets like Synthia [9] and Virtual KITTI [46] were created. There is some literature which demonstrates that a combination produces reasonable results in small datasets. But they are still limited for a commercial deployment of an automated driving system.

Hence there is a recent effort to build massively larger semantic segmentation datasets like Berkley Deep Drive(BDD) [47] and Toronto City [48]. In Berkeley Deep Drive Dataset they are working on a massive semantic segmentation dataset of 100,000 hours annotated for bounding boxes and semantic segmentation. Building a semi-automated annotation tool for bootstrapping, temporal propagation, HD maps, etc. First release is to be published by the end of 2017. Toronto City is a massive semantic segmentation, mapping and 3D reconstruction dataset covering 712 km2 of land, 8439 km of road and around 400,000 buildings. The annotation is completely automated with some manual verification and tuning. Clever algorithms that use Aerial Drone data, HD maps, city maps and LIDARs are deployed.

C. Learning Challenges

1) ***Class imbalance:*** There is severe class imbalance due to the fact that important objects like pedestrians are under represented unlike sky and building. This could also create a bias to ignore small objects. This could be handled by a weighting scheme in the error function. Another potential solution is to use mask predictions on detected bounding boxes of these small objects as in [49][50].

2) ***Unobserved Objects:*** Because the soft-max classifier is normalized to probability one, it doesn't allow room for unknown objects. The algorithm tries to allow one of the classes seen before for an unobserved object. Daimler tried to solve this issue outside the semantic segmentation framework by making use of dense depth. This could be handled by measuring uncertainty of the output classification, similar to Bayesian Segnet [51].

3) ***Complexity of Output:*** The output representation of semantic segmentation is a bunch of complex contours and can be overwhelming in very high textured scenes. The post processing modules like mapppin or maneuvering need simpler representation from this. This brings about a question of why not classify this simpler representation directly instead of semantic segmentation.

4) ***Recovering individual objects:*** Pixel-wise Semantic segmentation produces regions of same object and hence does not provide individual objects in a segment. This might be needed for tracking applications which tend to track objects like pedestrians individually. One solution is to use post processing classifier to further sub-divide the regions but this could be directly classified instead. However, a recent instance level segmentation paradigm can segment different instances of the same class as in [49] without the need for post processing.

5) ***Goal Orientation:*** Semantic segmentation is a generic problem and at the moment there is no goal orientation towards the end goal of automated driving. For instance, there may not be a need for accurate contour of objects or in detecting irrelevant objects like sky for end driving goal. This could be achieved to some extent by weighting of objects but a modular end to end system will be scalable to automatically perform it.

6) ***Varying object complexity:*** Automotive scene has varying complexity with simple structures like road or sky and complex structures like pedestrians. Pedestrians have higher complexity due to large appearance variations and articulations. Thus instead of using a small complexity network across the image, a varying complexity network like a cascaded CNN might be better.

7) ***Corner Case Mining:*** As the object detection parts are tightly coupled, it is difficult to do hard negative mining and to analyze corner cases. Even when known, it is hard to record use cases for the same. But, it can be done in synthetic sequences.

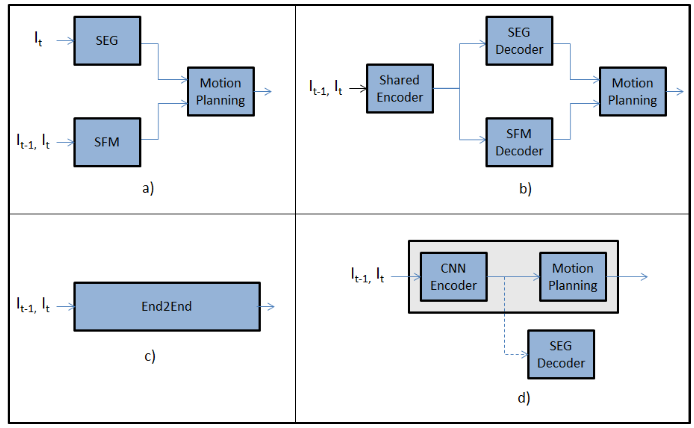

Fig. 5: Different application architectures - a) Classical architecture b) Shared encoder for multi-task learning c) End to end learning d) Modular end to end learning

V. Alternative Application architectures

In this section different alternatives to pure semantic segmentation are discussed. We present it with other possibilities where it can be coupled.

Multi-task Learning: Since the same CNN features generalize well for various tasks beyond object detection like flow estimation, depth, correspondence, and tracking. Thus a common CNN feature pipeline can be harmonized to be used for various tasks. In [50], they propose a CNN encoder and decoder for various tasks like scene classification and vehicle detection. A joint flow estimation and semantic segmentation in [52] was presented.

End to end learning: Human beings perform soft computing and they do not perform an accurate object detection when driving. They are goal oriented and an accurate object detection is not necessary for safe driving. End to end has the big advantage of not having to do any annotation as the driving signal outputs are obtained directly from the Controller Area Network (CAN) signals. Companies like Uber are betting this away as they can collect loads of driving data through their taxi fleet.

The output is of fewer dimensions (brake, steering, acceleration) and also temporally smooth. Hence for the same input, mathematically this function should have a simpler functional complexity relative to the complex output structure of semantic segmentation. The work in [13][14] is in that direction.

Modular End to End learning: We use the term modular end to end learning when there are auxiliary losses to ensure safety and interpret ability. For instance, segmentation loss can be added as an auxiliary loss for an end to end driving CNN [53]. Using this auxiliary loss, the CNN loosely learns to semantically segment, but it is also learns to have a better representation for the intermediate features. It was shown in that work that using auxillary loss outperforms the vanilla end to end learning. The work also uses recurrent gated unit after the CNN to model temporal information.

VI. Benchmarking and Design Exploration

A. Benchmarking

In this section a comparative evaluation of different semantic segmentation architectures is presented. Although there has been numerous work showing evaluation of different architectures on Camvid [18]. However the previous work was concerned only with the accuracy of the segmentation. We present an evaluation of both shallower and state of the art work in terms of mean intersection over union and speed. The comparison is shown in Table II. Some networks that has not been evaluated on the semantic segmentation for automated driving are also presented. Thus covering a wider range of potential efficient architectures. This can guide further decisions on what would best fit in the automated driving system. Although other architectures such as DeepLab[23] show much better accuracy and are the state of the art in segmentation, but are computationally inefficient. Thus, these architectures are not included in the comparison. Evaluation metrics used are mean intersection over union(IoU) and per class IoU. The running time for inference is computed in seconds. The different architectures are evaluated on a GTX TITAN GPU with images of resolution 480x360.

The architectures that are primarily evaluated are : (1) Unet [39]. (2) Xception [54] which is a classification network that was not used in the segmentation problem before. (3) Dilated FCN16s, an architecture that was designed to be computationally and memory efficient with reasonable accuracy. (4) FCN8s [28]. (5) Segnet Basic [30]. (6) Dilation8 [36]. (7)Enet [55], which is the most efficient architecture for semantic segmentation. A unified framework with the first five architectures is going to be publicly available to help further research. While the results of the last two architectures are reported from their work. Note that the mean class IoU is computed over all classes even the ones not included in the Table II. But, only the classes of interest were the ones mentioned in Table II.

Although Dilation8 outperforms all previous architectures in mean IoU it has the largest running time. This renders it as an inefficient solution to semantic segmentation for automated driving. However, the dilated convolution idea can be adapted in a shallower network. It uses dilated convolution to increase receptive field while maintaining the resolution of the segmentation. Dilated FCN16s is an adapted version of FCN-16s as originally introduced in [28]. Two pooling layers are removed along with the convolutional layers in between them and conv4/conv5 layers are reduced to two dilated convolution layers with dilation factors of 2 and 4 respectively. This leads to a decrease in the size of the network and its running time for real-time applications. Another architecture used for medical image segmentation was experimented on Camvid which is called Unet. It turned to work second best on Camvid, but the running time is still not practical for real deployment. Xception [54] is an architecture that is mainly relying on depthwise separable convolution, that separates the spatial convolution from depthwise convolution. Although the network is designed for classification, it has been transformed to a fully convolutional network for the purpose of segmentation. The network mean IoU was much lower than other architectures, with a very small improvement in the running time against Segnet. Although Segnet is not considered as the state of the art in segmentation, but it turned out to provide a good balance between mean IoU and speed. In our experiments using batch normalization [56] turned to be effective in training both Segnet and Unet. It turned to converge faster, and it got better mean IoU of 47.3% in case of Segnet. It is worth noting that in case of FCN8s, we were able to reproduce similar results to the work in[32]. But this is less by 2% than what was reported in [51].

B. Design Exploration

Deep learning is a rapidly progressing area of research. Most of the research is disparate in which the various ideas developed in different architectures are not formalized because of the lack of theory. Hence it is hard to combine ideas from two top networks from an application development perspective. Additionally the main area of active research in deep learning is on image recognition problems (as in ImageNet challenge) and the ideas trickle down to semantic segmentation. Additionally efficiency is typically not a design criteria in academic research as majority of leading networks are very large comprising of hundreds of layers and employ ensemble of several networks. The work in [57] compared various networks' accuracy normalized to the amount of computation and shows ResNet and GoogleNet are efficient architectures. This suffers from the same problem of treating the different networks independently and hence does not formalize and combine ideas.

Some good design choices that are accepted with-in the community are presented:

(1) The use of 3x3 convolutions similar to VGG architectures [58] turned to be useful experimentally. Especially in scenarios where you care about the resolution of your input such as segmentation. Since larger filter size will cause reduction in the image resolution.

(2) The dilated convolution is considered to be the best practice in segmentation as it increases the receptive

TABLE II: Semantic Segmentation Results on Camvid. Running time in seconds, mean IoU, and perclass IoU is shown. Some of the 11 classes are shown due to limited space.

	Running Time	Mean Class IoU	Per-Class IoU						
			Sky	Building	Road	Sidewalk	Vegetation	Car	Pedestrian
FCN16s Dilated	0.07	46.7	86.3	69.1	87.8	63.7	60.8	63.6	21.4
Xception[54]	**0.02**	42.8	81.9	68.9	86.6	62.9	61.6	60.8	19.8
Segnet-Basic[30]	0.03	46.4	87.0	68.7	86.2	60.5	52.0	58.5	25.3
FCN8s[28]	0.33	49.7	87.6	75.5	87.2	67.2	70.6	76.4	27.7
Unet[39]+BN[56]	0.56	53.9	90.2	72.6	89.1	67.2	67.7	74.7	34.1
Dilation8[36]	0.6474	**65.3**	89.9	**82.6**	92.2	75.3	76.2	**84.0**	56.3
Enet[55]	0.047	51.3	**95.1**	74.7	**95.1**	**86.7**	**77.8**	82.4	**67.2**

field without downgrading resolution. Although in our comparative evaluation Dilation8 was not suitable for real-time applications. However that can be due to their use of a deep network to build upon.
(3) For real-time performance shallow networks can be useful for segmentation with a compromise in the accuracy.
(4) Batch normalization [56] turned to be a very useful trick for better convergence during training in our experiments. This is due to the reduction of change in distribution of network activations. They termed that as reduction of internal covariate shift. The covariate shift occurs due to change of networks parameters during training.
(5) The resolution of the input image largely affects the segmentation, although it seems as a tiny detail. We found that higher input image resolution can help with segmenting small objects like pedestrian. Also, using random crops to help reduce the class imbalance can further help the segmentaiton. This can be seen in Dilation8 results, they use random crops from the image that is then upsampled as input.
(6) Skip connections is widely used in segmentation architectures such as FCN8s [28] and U-net [39]. However, the extensive use of these skip connections can lead to overhead in memory bandwidth.

VII. Conclusion

In this paper we have conducted a thorough review of deep semantic segmentation. Including fully convolutional network architectures and different variants that work patchwise, pixelwise, or that support multiscale aggregation. Then utilizing structure and spatio-temporal features in the segmentation problem is discussed. One important aspect that is missing from various surveys is introducing the problem of automated driving and the potential use of semantic segmentation with-in this area. The different datasets that can be used in automated driving setting is summarized. A set of challenges are also presented to guide both the research and industrial community to what are the current bottlenecks in such systems. A comparative evaluation is conducted on different state of the art network architectures on an urban scene dataset. The comparison included architectures that have not been tried in this setting, along with comparison in terms of both accuracy and performance. Finally, some design choices are presented to be of huge benefit to the community. Due to page limitations, the contents of this survey are kept high level. The authors are working on a more detailed survey which will include more detailed mathematical formalism.

References

[1] A. Valada, G. L. Oliveira, T. Brox, and W. Burgard, "Deep multispectral semantic scene understanding of forested environments using multimodal fusion," in *The 2016 International Symposium on Experimental Robotics (ISER 2016)*, 2016.

[2] T. M. Bonanni, A. Pennisi, D. Bloisi, L. Iocchi, and D. Nardi, "Human-robot collaboration for semantic labeling of the environment," in *Proceedings of the 3rd Workshop on Semantic Perception, Mapping and Exploration*, 2013.

[3] V. Vineet, O. Miksik, M. Lidegaard, M. Nießner, S. Golodetz, V. A. Prisacariu, O. Kähler, D. W. Murray, S. Izadi, P. Perez, and P. H. S. Torr, "Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[4] A. Kundu, Y. Li, F. Dellaert, F. Li, and J. M. Rehg, "Joint semantic segmentation and 3d reconstruction from monocular video," in *European Conference on Computer Vision*. Springer, 2014, pp. 703–718.

[5] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: learning dense volumetric segmentation from sparse annotation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2016, pp. 424–432.

[6] W. Zhu and X. Xie, "Adversarial deep structural networks for mammographic mass segmentation," *arXiv preprint arXiv:1612.05970*, 2016.

[7] O. Miksik, V. Vineet, M. Lidegaard, R. Prasaath, M. Nießner, S. Golodetz, S. L. Hicks, P. Pérez, S. Izadi, and P. H. Torr, "The semantic paintbrush: Interactive 3d mapping and recognition in large outdoor spaces," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 3317–3326.

[8] H. Zhang, A. Geiger, and R. Urtasun, "Understanding high-level semantics by modeling traffic patterns," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3056–3063.

[9] 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*, 2016, pp. 3234–3243.

[10] 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, 2009.

[11] 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," *arXiv preprint arXiv:1604.01685*, 2016.

[12] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," DTIC Document, Tech. Rep., 1989.

[13] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[14] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. L. Cun, "Off-road obstacle avoidance through end-to-end learning," in *Advances in neural information processing systems*, 2005, pp. 739–746.

[15] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2722–2730.

[16] H. Zhu, F. Meng, J. Cai, and S. Lu, "Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation," *Journal of Visual Communication and Image Representation*, vol. 34, pp. 12–27, 2016.

[17] J. Shotton, M. Johnson, and R. Cipolla, "Semantic texton forests for image categorization and segmentation," in *Computer vision and pattern recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[18] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *European conference on computer vision*. Springer, 2008, pp. 44–57.

[19] 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, no. 8, pp. 1915–1929, 2013.

[20] C. Farabet, N. EDU, C. Couprie, L. Najman, and Y. LeCun, "Scene parsing with multiscale feature learning, purity trees, and optimal covers."

[21] D. Grangier, L. Bottou, and R. Collobert, "Deep convolutional networks for scene parsing," in *ICML 2009 Deep Learning Workshop*, vol. 3. Citeseer, 2009.

[22] G. Lin, C. Shen, I. Reid *et al.*, "Efficient piecewise training of deep structured models for semantic segmentation," *arXiv preprint arXiv:1504.01013*, 2015.

[23] 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," *arXiv preprint arXiv:1606.00915*, 2016.

[24] 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.

[25] E. Shelhamer, K. Rakelly, J. Hoffman, and T. Darrell, "Clockwork convnets for video semantic segmentation," *CoRR*, vol. abs/1608.03609, 2016. [Online]. Available: http://arxiv.org/abs/1608.03609

[26] P. Sturgess, K. Alahari, L. Ladicky, and P. H. Torr, "Combining appearance and structure from motion features for road scene understanding," in *BMVC 2012-23rd British Machine Vision Conference*. BMVA, 2009.

[27] C. Russell, P. Kohli, P. H. Torr *et al.*, "Associative hierarchical crfs for object class image segmentation," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 739–746.

[28] 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.

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

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

[31] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 41–48.

[32] M. Fayyaz, M. H. Saffar, M. Sabokrou, M. Fathy, and R. Klette, "STFCN: spatio-temporal FCN for semantic video segmentation," *CoRR*, vol. abs/1608.05971, 2016. [Online]. Available: http://arxiv.org/abs/1608.05971

[33] M. Siam, S. Valipour, M. Jagersand, and N. Ray, "Convolutional gated recurrent networks for video segmentation," *arXiv preprint arXiv:1611.05435*, 2016.

[34] D. Nilsson and C. Sminchisescu, "Semantic video segmentation by gated recurrent flow propagation," *arXiv preprint arXiv:1612.08871*, 2016.

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

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

[37] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," *arXiv preprint arXiv:1511.03339*, 2015.

[38] G.-J. Qi, "Hierarchically gated deep networks for semantic segmentation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[39] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[40] "Tomtom roaddna,," http://automotive.tomtom.com/products-services/hd-map-roaddna/.

[41] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfusion: Dense 3d semantic mapping with convolutional neural networks," *arXiv preprint arXiv:1609.05130*, 2016.

[42] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "Elasticfusion: Dense slam without a pose graph." in *Robotics: science and systems*, vol. 11, 2015.

[43] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox, "Demon: Depth and motion network for learning monocular stereo," *arXiv preprint arXiv:1612.02401*, 2016.

[44] M. Treml, J. Arjona-Medina, T. Unterthiner, R. Durgesh, F. Friedmann, P. Schuberth, A. Mayr, M. Heusel, M. Hofmarcher, M. Widrich *et al.*, "Speeding up semantic segmentation for autonomous driving."

[45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[46] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtual worlds as proxy for multi-object tracking analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4340–4349.

[47] "Berkley Deep Drive," http://bdd.berkeley.edu/, 2017.

[48] S. Wang, M. Bai, G. Mattyus, H. Chu, W. Luo, B. Yang, J. Liang, J. Cheverie, S. Fidler, and R. Urtasun, "Torontocity: Seeing the world with a million eyes," *arXiv preprint arXiv:1612.00423*, 2016.

[49] J. Dai, K. He, Y. Li, S. Ren, and J. Sun, "Instance-sensitive fully convolutional networks," *CoRR*, vol. abs/1603.08678, 2016. [Online]. Available: http://arxiv.org/abs/1603.08678

[50] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun, "Multinet: Real-time joint semantic reasoning for autonomous driving," *arXiv preprint arXiv:1612.07695*, 2016.

[51] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," *CoRR*, vol. abs/1511.02680, 2015. [Online]. Available: http://arxiv.org/abs/1511.02680

[52] J. Hur and S. Roth, "Joint optical flow and temporally consistent semantic segmentation," *CoRR*, vol. abs/1607.07716, 2016. [Online]. Available: http://arxiv.org/abs/1607.07716

[53] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-end learning of driving models from large-scale video datasets," *arXiv preprint arXiv:1612.01079*, 2016.

[54] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *arXiv preprint arXiv:1610.02357*, 2016.

[55] 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*, 2016.

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

[57] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," *arXiv preprint arXiv:1605.07678*, 2016.

[58] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.