Boundary Enhanced Semantic Segmentation for High Resolution Electron Microscope Images

Matthias Pollach*, Felix Schiegg*, Matthias Ludwig*†, Ann-Christin Bette*† and Alois Knoll*
*Technical University Munich, †Infineon Technologies AG

***Abstract*—This work proposes an automated semantic segmentation approach for high resolution scanning electron microscope images, which enables the detection of hardware Trojans and counterfeit integrated circuits. We evaluate state of the art segmentation approaches and leverage expert domain knowledge to propose a neural network architecture tailored for our use case. We further address the challenge of the limited availability of training images and evaluate which pre-trained encoder can be leveraged most effectively for the given use case. The proposed segmentation network uses expert domain knowledge to account for the importance of separating technology features on a fine-grain level by introducing a separate boundary stream. The test results compare our network to a baseline approach and to two state-of-the-art segmentation networks.**

***Index Terms*—counterfeit electronics, hardware Trojans, scanning electron microscope image segmentation, semantic segmentation, integrated circuits, machine learning, neural networks**

I. Introduction

The safe operation of semiconductor devices is essential when they are used in critical applications such as the medical or automotive sector. For this reason, integrated device manufacturers put a lot of effort into electrical testing and process control. Yet, in the horizontally distributed supply-chain, adversaries got into the market and counterfeit electronics are a multi-billion dollar market [9]. Detection schemes for forged electronics are in high demand. We present a novel approach building on the extension of an established analytic process: the inspection of scanning electron microscope (SEM) images of semiconductor device cross-sections. The proposed method can be used for internal process characterization - process stability, defect analysis, or root-cause-analysis - or as the enabler for a future counterfeit detection method.

An important aspect in this context is to measure the distance between technological features, which allow conclusions regarding the present technology and the production process of a microchip. The more precisely these features can be classified, the higher the quality of applications that use the segmentation as an input. Throughout this work, we focus on the specific challenges introduced by SEM images, which are addressed by a tailored model architecture. As part of this, we leverage certain properties of the images and expert domain knowledge on the microchips to address the high quality demands.

The SEM images used for the present use case contain metal layers and vertical interconnection accesses (VIAs). The selected field of views are between $4\,\mu\mathrm{m}$ and $70\,\mu\mathrm{m}$. In order to reduce the deviations within the data set, only images were chosen where the size ratios of metal layers and VIAs are in the same order of magnitude.

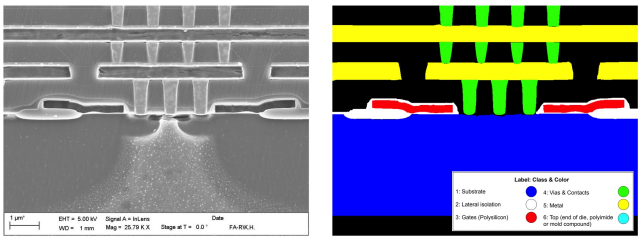


Fig. 1: Pixel wise labelled SEM image

All images were recorded using two microscopes with a resolution of either 1024×768 or 1280×960 pixels. To enable supervised learning for semantic image segmentation, the SEM images have to be annotated with respect to the relevant classes. This requirement drastically limits the number of available images, so the resulting data set is very small. This inherently causes challenges like overfitting, class imbalances and difficulties in optimization, which will be addressed throughout this work. In total seven classes are identified as being most relevant for the semantic segmentation which is shown in figure 1. The characteristics of the visualized structures complicate the automated reverse engineering in several aspects. In particular, several boundary conditions have to be taken into consideration, which are introduced by the production process. Furthermore, these conditions vary greatly depending on the chip technology that is to be analyzed. The appearance of the prepared sample varies due to the manual effort and the use of a wide variety of chemicals to expose the cross-section. In addition, the probe is manually cut which leads to damages and again a large variation among probes. Iterative image acquisition with constantly increasing magnification is performed with an electron microscope. Due to the electron reflection at edges, SEM images show very bright boundaries, while the majority of a component appears as a homogeneous unit. When looking at a metal layer in figure 1, this effect is well visible. This is a stark contrast to most semantic segmentation approaches, where objects have a more heterogeneous appearance.

II. Related Work

Machine learning methods have become increasingly influential in the field of hardware security. They are mainly used as defensive methods against hardware Trojans and IC counterfeits. Machine learning is also used for side-channel attacks and to launch Physical Unclonable Functions (PUFs)

clone models to enable IC overbuilding [5]. In the context of hardware reverse engineering, ML models are primarily used to analyze layout images of ICs. Botero et al. [2] list a survey of ML-based approaches published in reverse engineering. These mainly involve both supervised and unsupervised learning methods and aim to segment the materials of a layer, identify standard cells, and detect malicious modifications in the layout. For example, the authors of [11] published a fully convolutional network with VGG-16 encoder for segmentation of metal tracks and VIAs. In [4], the authors postulate an unsupervised K-means approach to the same task, but they acknowledge that this method is severely limited due to preparation shortcomings and image variations.

The development of applied methods for counterfeit detection has yet been exclusively limited to package analyses [1], [6], [10]. The authors have shown different approaches to distinguish an original sample against a counterfeit product through computer vision techniques. Our approach presents the first model towards an automated detection scheme on the technology level.

Two domains where semantic segmentation is widely used are autonomous driving and medical applications. In the context of autonomous driving, a large variety of objects in the environment are segmented. In the medical domain, various imaging technologies produce grayscale images of the human body. For cancer detection, segmentation is indispensable because of the ability to detect cancerous tissue, which is part of an organ while having different properties compared to the regular tissue.

Throughout this work, we select U-net as a baseline architecture, which has proven to work for small data sets across different domains and particularly, in the medical domain, as for example shown in the ISBI cell tracking challenge 2015 [16]. Additionally, more recent architectures like feature pyramid network (FPN) and pyramid scene parsing network (PSPNet), have evolved in recent years [13], [19]. Using a pyramidal hierarchy at multiple scales, FPN is a region proposal and classification network that creates a feature pyramid. FPN has outperformed other region proposal networks like DeepMask, SharpMask and InstanceFCN [13]. PSPNet enriches the feature space by extracting features at multiple scales and has shown its superiority in the ImageNet scene parsing challenge 2016 [19]. Another architecture that has shown great benefits in the automotive domain is the gated shape convolutional neural network (GSCNN). It uses a two-stream architecture that leverages shape information in a separate stream in addition to the traditional CNN feature stream [18].

III. Proposed Methodology

A. Inspection Framework

IC manufacturers have a strong need to characterize their internal processes to validate process stability, to detect defects, to execute root-cause-analysis or to perform counterfeit detection. Figure 2 shows the automated process from IC preparation to advanced analyses. In order to properly execute these analyses, the chip needs to be cut vertically in a first step. After polishing and preparing the surface, images are taken at distinct zoom levels to investigate the relevant area of the chip using a scanning electron microscope (SEM). To enable automated measurements, the SEM images are segmented and serve as the basis for all further stages. The proposed process described throughout this paper provides an automated semantic segmentation of images which is the enabler for in depth analysis and counterfeit detection analysis of ICs. However, a Reverse Engineer assesses the outcome of the image segmentation step and determines if the quality is sufficient to continue with additional analysis.

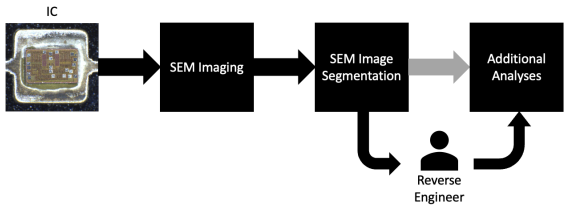


Fig. 2: Automated IC framework for SEM image segmentation enabling advanced analyses with humans in the loop.

B. Data Set

All SEM images used for the present use case have a resolution of either 1024×768 or 1280×960 pixels and are stored as 8-bit grayscale images. The images contain seven segment classes, which are shown in figure 1, that are relevant for the desired semantic segmentation use case. These classes are substrate, lateral isolation, polysilicon, VIA, metal, top of die and background. The limited data set and the structure of the probes directly results in a strong imbalance between classes. Background (38.1%) and substrate (29,6%) pixels are dominant with two-thirds of the overall pixels, whereas lateral isolation (1.3%), polysilicon (0.5%) and VIA (2.5%) occur much sparser. The arrangement and occurrence of the defined classes are determined by the technology. The different shades of gray result, in parts, from the different densities of the various materials. The bright edges result from the enhanced electron reflections at edges. The highly specialized application of segmenting SEM images requires data specifically labelled for the present use case. However, due to required expert domain knowledge for annotation, the creation of ground truth for this type of image is very resource intensive. For the present work, two labelling approaches were considered: A pixel level accurate annotation and a polygon approximated annotation. The pixel level accuracy requires a more time consuming annotation process but avoids wrongly assigned pixels. On the other hand, the polygon approximation is faster but pixels are falsely assigned to a class. Given the limited availability of images, a pixel level accurate annotation is chosen to achieve a more focused and robust training process resulting in a total number of 40 manually labelled images.

C. Pre-Processing

A very important aspect in machine learning applications is data pre-processing. Neural networks create a high dimensional feature space, which is used to determine the different classes. In theory, the higher dimensional the feature space is, the easier the separation of classes gets. The learning capacity is directly influenced by the number of nodes and the number of layers present in the network. An increase in the number of layers and nodes, leads directly to an increase in learning capacity. This enables the network to learn more complex transfer functions. The capacity impacts the scope and types of functions that can be approximated by the model. When evaluating the models capacity, it is crucial to consider the concept of overfitting and underfitting.

In our use case, we aim to exploit our expert domain knowledge so that the to be approximated transfer function of the neural network becomes less complex. For the present use case, this means that we process the data before proving it to the neural network in such a way, that we reduce the complexity of the to be approximated transfer function. For the present work, the images are then resized to 224×224 pixels so that the images can be directly used by the neural network. This happens to ensure feature transferability and to reduce the memory footprint. The latter is particularly important to reduce the training time and to manage the required computational resources.

D. Baseline Classifier

1) Data Augmentation: A common method is to simply use more data to allow for generalization [7]. However, this is very costly for most applications and often data is strongly limited. Especially in our use case, it is very expensive to generate training data because of very complex data acquisition, which is very human-labor-intensive and requires special equipment. Another key element that allows for generalization is data augmentation. Thereby, artificial training data is created by slightly altering the original images. There is a large variety of these techniques which have different benefits and drawbacks. For the present work we consider the following techniques:

Optical distortion, grid distortion, elastic transformation, median blurring, Gaussian noise injection, adaptive histogram equalization (CLAHE), random cropping and horizontal flips. Given the boundary conditions of SEM images, which are used as domain knowledge to simplify the classification problem, vertical flips are not considered because it would violate some key assumptions regarding the structure of SEM images and the relation of different segments to each other.

All of the above listed data augmentation techniques were evaluated using the the dice score as a performance metric, which measures how similar a target A is to an output B:

$$Dice \quad score = \frac{2\,|A \cap B|}{|A| + |B|} \tag{1}$$

The dice score penalizes false positive class detection, which is beneficial for our use case, due to the imbalanced data set. To our surprise, neither Gaussian noise injection, nor using CLAHE pre-processed images led to an improvement. Consequently, all other listed techniques are used for data augmentation with the exception of Gaussian noise injection and CLAHE.

2) Encoder Selection: Due to the nature of our use case, data is very limited and this results in a very small data set, compared to most other supervised learning problems. Consequently, it is not feasible to train a classifier from scratch, yet it would lead to very poor classification results due to overfitting to the training data set. We are able to address some of the challenges by leveraging pre-trained encoders, which were trained on much larger data sets. More specifically, we leverage encoders that were trained on the ImageNet data set [17]. Essentially, we adapt the encoder so that it matches our use case of image segmentation. The underlying hypothesis is, that we will be able to adapt an encoder to our problem leveraging transfer learning and data augmentation. Literature shows that neural networks leveraging transfer learning are better performing and faster converging when compared to similar network architectures that are trained from scratch. For our use case, it is essential to make use of this because of the limited data set as shown in [12], while simultaneously aiming to use an encoder that has demonstrated high performance as shown in [3], [14]. For the given use case, we reevaluated the following encoders trained on the 2012 ILSVRC ImageNet data set [17]: ResNet18, ResNet50, ResNet101, ResNet152, SE-ResNeXt50, SE-ResNeXt101, VGG11, VGG19, Densenet161, Densenet201 and DPN131.

For evaluation purposes, we used a fix learning rate of 1e-4 while ensuring that the same training and validation split is used for all networks. Cross-validation is applied by executing three training runs per encoder. The performance was evaluated based on the dice score, while considering the total number of parameters of an encoder. The best performance on average was achieved by SE-ResNeXt50, which is used as an encoder throughout this work.

3) Network Architecture Evaluation: An important aspect of selecting the adequate architecture and the most suitable encoder is the ability to compare ourselves to a baseline. We choose to compare U-net, FPN, PSPNet in the first step, while ensuring that the same encoder is used for the various network architectures to allow for a fair comparison. Based on the dice score, multi-scale and pyramid based networks perform better, while the performance of FPN and PSP is comparable.

4) Batch size: The batch size influences the required computational resources, generalization capabilities and mainly the training dynamics. Large batch sizes allow for parallelization during training and a more accurate estimate of the gradient [8]. However, memory requirements increase linearly with batch size, which is a limiting factor for many applications. The authors of [15] showed, that a small batch size has the ability to improve generalization capabilities while introducing much looser memory constraints. Nevertheless, due to the limited number of available training images, we evaluate batch sizes between 1 and 32.

Our evaluation shows, that a batch size of four performs well,

while smaller and larger batch sizes decrease the performance. Due to the small data set, a small batch size leads to the network not learning properly and a larger batch size leads to overfitting.

E. Boundary Enhancement

A major disadvantage of using CNNs is the loss of spatial resolution which leads to suppressing high frequency components, which directly results in blurry edges for segmentation. One way to address this is the use of skip connections and feature map concatenations. This preserves more of the high frequency components while introducing a small overhead by passing additional components through the entire network. However, this also leads to undesired information being passed through the network which increases the complexity of the segmentation problem. This results in further inefficiencies because a large variety of features is processed within one deep CNN [18]. A robust boundary detection is very important for our use case, but challenging because of blurry and noisy artifacts at edges in SEM images.

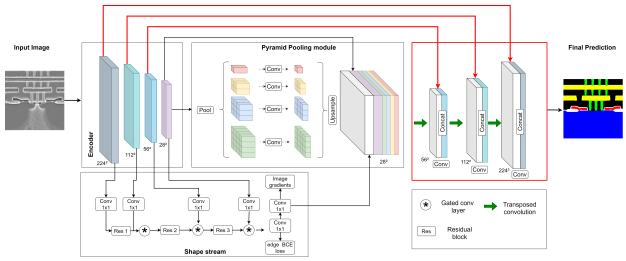


Fig. 3: PUBNet: The outputs of each stage of the encoder are passed forward to the shape stream and to the corresponding level in the upsampling path (highlighted in red). The size of each feature map is indicated bellow each feature map.

1) GSCNN: Leveraging a two stream architecture for semantic segmentation, the shape information propagates through a separate processing branch in the network. The focus on shape information in this branch allows for an improved boundary detection, especially when high frequency components are of high importance. The approach of using a separate shape stream is described in more detail in [18]. The authors also describe why using cross entropy on its own is not sufficient to train the network. There are two separate predictions for segmentation and boundary, which have to be jointly supervised. For the segmentation map, cross entropy is used, whereas binary cross entropy is used for the boundaries. This results in a loss $\mathcal{L}_{JMTL}$ that combines both components.

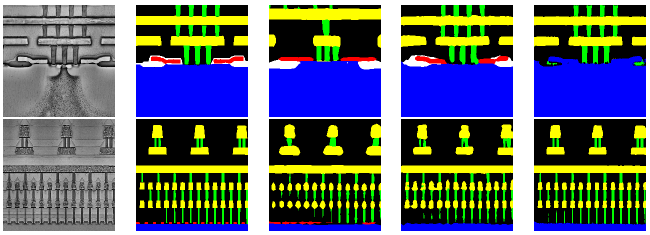
$$\mathcal{L}_{JMTL} = \lambda_1 \mathcal{L}_{\mathcal{BCE}}\left(s, \hat{s}\right) + \lambda_2 \mathcal{L}_{\mathcal{CE}}\ \left(\hat{y}, f\right) \tag{2}$$

The Sobel filter in x and y direction serve as the ground truth boundary maps $\hat{s} \in \mathbb{R}^{HxW}$, whereas $\hat{y} \in \mathbb{R}^{HxW}$ is the ground truth for the semantic map. For the hyperparameters λ_1 and λ_2 we follow the suggestion of the authors of [18] to set them to λ_1=20 and λ_2=1. A major disadvantage of this network is the number of total parameters, that need to be trained. Consequently, we evaluate GSCNN and an adapted lightweight version of GSCNN (L-GSCNN), using SE-ResNeXt50 as an encoder, which has fewer parameters.

2) PUBNet: We introduce our own network architecture PSPNet with U-Net like upsampling and boundary enhancement (PUBNet) shown in 3, which is tailored to our use case and compensates some of the known shortcomings of the previously discussed networks. Upsampling the low dimensional feature map outputs of PSPNet in a single step would result in a loss of information eliminating the desired advantages introduced by a separate shape stream. To address this, we use an upsampling approach comparable to U-Net. The expansive path of the network uses 3x3 convolutional, batch normalization and ReLu layers. Upsampling is implemented using 2x2 transposed convolutions with a stride of 2 and the output of each upsampling step is concatenated with the respective feature map from the encoder through skip connections. Consequently, our network is based on a PSPNet architecture that uses an usampling approach comparable to U-Net, while leveraging a separate shape stream to preserve the desired boundary information. Based on our evaluations and to allow comparability, SE-ResNeXt50 is used as an encoder.

IV. Results

The baseline classifier uses a PSPNet with a SE-ResNeXt50 encoder and is compared to architectures, which specifically aim at improving boundary detection. The results are presented in table I. All approaches improve the average performance of the network when compared to the baseline. When comparing GSCNN and L-GSCNN we observe a smaller variance for L-GSCNN results, which is an indication that fewer parameters are better suited for our use case. In addition to the dice score, we also evaluate the networks based on mean intersection over union, as shown in table I. We see very similar results, which are comparable to the dice score and indicate the same trends. In addition to the purely analytical driven evaluation, it is important to qualitatively evaluate how well the segments are separated. The SEM images and the according ground truth with the resulting segmentation are shown in figure 4. We clearly see that the baseline network and the L-GSCNN network fail at the task of separating individual fain grained segments, whereas PUBNet is able to provide a clear separation.



(a) Input (b) Target (c) Baseline (d) GSCNN (e) PUBNet

Fig. 4: Input-target pairs and exemplary output masks of the Baseline, lightweight GSCNN (L-GSCNN) and PUBNet.

Architecture	Parameters	Dice Score				mIoU			
		T1	T2	T3	Avg	T1	T2	T3	Avg
Baseline	26 331 511	0.852	0.855	0.840	0.849	0.372	0.361	0.355	0.363
GSCNN	137 275 118	0.889	0.850	0.927	0.889	0.38	0.366	0.439	0.395
L-GSCNN	42 525 534	0.910	0.876	0.929	0.905	0.415	0.373	0.448	0.412
PUBNet	31 073 009	0.894	0.917	0.91	**0.907**	0.429	0.416	0.42	**0.422**

TABLE I: Different shape enhancing architectures evaluated based on their number of parameters, dice score and mIoU shown per training run (T1, T2 and T3) and average (Avg) performance.

V. Conclusion and Outlook

This work contributes to enabling automated in depth analysis and counterfeit detection analysis that goes beyond packaging analyses. Our results indicate, that our own tailored network architecture outperforms state-of-the-art approaches for our given use case. This is achieved by leveraging state-of-the-art approaches from the medical and the autonomous driving domain, while taking into account the specific challenges of our application. The knowledge of the importance of image boundaries was exploited by introducing the separate shape stream, so that individual components can be separated. Our final results allow measurements, which are the basis for more advanced hardware Trojan and counterfeit analyses. The approach of exploiting domain knowledge offers the potential to extend this work to other use cases in other domains.

The present work only includes a limited number of classes that define a semiconductor device technology. The available data for some classes is extremely limited and results in weak performance for strongly underrepresented classes. Missing classes not represented in the data set include for example deep trench geometries, characteristics of the package, or gate oxide geometries. Furthermore, the variance between different microscope imaging settings and preparation techniques was not addressed. Additionally, the limited number of available annotated images did not allow for a more extensive evaluation.

It is a question of further research to design experiments which will enable the detection of counterfeits on the technological level. The presented approach might be extended towards multiple imaging technologies, like optical or transmission electron microscopes. For future efforts, it is essential to generate more labelled images with a focus on having sufficient examples of all relevant classes.

Acknowledgment

AI4DI receives funding within the Electronic Components and Systems for European Leadership Joint Undertaking (ECSEL JU) in collaboration with the European Union's Horizon2020 Framework Programm and National Authorities, under grant agreement n° 826060.

References

[1] Navid Asadizanjani, Mark Tehranipoor, and Domenic Forte, 'Counterfeit electronics detection using image processing and machine learning', *Journal of Physics: Conference Series*, **787**, 012023, (jan 2017).

[2] Ulbert J Botero, Ronald Wilson, Hangwei Lu, Mir Tanjidur Rahman, Mukhil A Mallaiyan, Fatemeh Ganji, Navid Asadizanjani, Mark M Tehranipoor, Damon L Woodard, and Domenic Forte, 'Hardware trust and assurance through reverse engineering: A survey and outlook from image analysis and machine learning perspectives', *arXiv preprint arXiv:2002.04210*, (2020).

[3] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, 'Encoder-decoder with atrous separable convolution for semantic image segmentation'.

[4] Deruo Cheng, Yiqiong Shi, Tong Lin, Bah-Hwee Gwee, and Kar-Ann Toh, 'Hybrid k-means clustering and support vector machine method for via and metal line detections in delayered ic images', *IEEE Transactions on Circuits and Systems II: Express Briefs*, **65**(12), 1849–1853, (2018).

[5] Rana Elnaggar and Krishnendu Chakrabarty, 'Machine learning for hardware security: opportunities and risks', *Journal of Electronic Testing*, **34**(2), 183–201, (2018).

[6] P. Ghosh and R. S. Chakraborty, 'Recycled and remarked counterfeit integrated circuit detection by image-processing-based package texture and indent analysis', *IEEE Transactions on Industrial Informatics*, **15**(4), 1966–1974, (2019).

[7] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep learning*, MIT Press, Cambridge, Massachusetts and London, England, 2016.

[8] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He, 'Accurate, large minibatch sgd: Training imagenet in 1 hour'.

[9] Ujjwal Guin, Daniel Dimase, and Mark Tehranipoor, 'Counterfeit integrated circuits: Detection, avoidance, and the challenges ahead', *Journal of Electronic Testing: Theory and Applications*, **30**, 9–23, (02 2014).

[10] Robb Hammond. Counterfeit electronic component detection.

[11] Xuenong Hong, Deruo Cheng, Yiqiong Shi, Tong Lin, and Bah Hwee Gwee, 'Deep learning for automatic ic image analysis', in *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, pp. 1–5. IEEE, (2018).

[12] Vladimir Iglovikov and Alexey Shvets, 'Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation'.

[13] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, 'Feature pyramid networks for object detection'.

[14] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, 'Learning and transferring mid-level image representations using convolutional neural networks', in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1717–1724, (2014).

[15] Dominic Masters and Carlo Luschi, 'Revisiting small batch training for deep neural networks'.

[16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, 'U-net: Convolutional networks for biomedical image segmentation'.

[17] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, 'Imagenet large scale visual recognition challenge'.

[18] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler, 'Gated-scnn: Gated shape cnns for semantic segmentation'.

[19] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, 'Pyramid scene parsing network'.