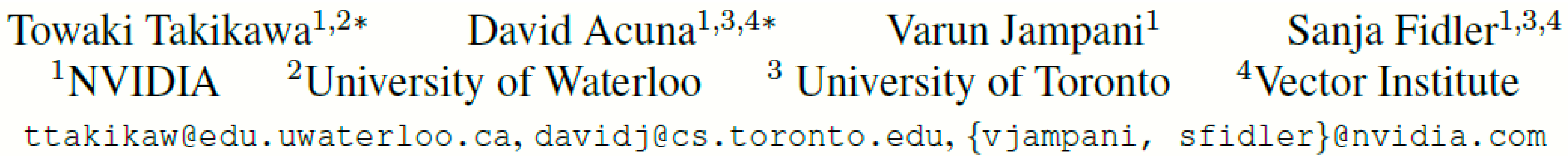
Gated-SCNN: Gated Shape CNNs for Semantic Segmentation (ICCV, 2019)



Towaki Takikawa[1,2*] David Acuna[1,3,4*] Varun Jampani[1] Sanja Fidler[1,3,4]
[1]NVIDIA [2]University of Waterloo [3] University of Toronto [4]Vector Institute
ttakikaw@edu.uwaterloo.ca, davidj@cs.toronto.edu, {vjampani, sfidler}@nvidia.com

Compiled by Jiajin Zhang

07/01/2020

Overview

- The authors proposed a new two-stream CNN architecture for semantic segmentation that explicitly wires shape information as a separate processing branch (shape stream)
- They use the higher-level activations in the classical stream to gate the lower-level activations in the shape stream, effectively removing noise and helping the shape stream to only focus on processing the relevant boundary-related information
- This method achieves SOTA performance on the Cityscapes benchmark

  mask (mIoU) improved by 2%

  boundary (F-score) quality improved by 4%

Gated-SCNN

Architecture overview

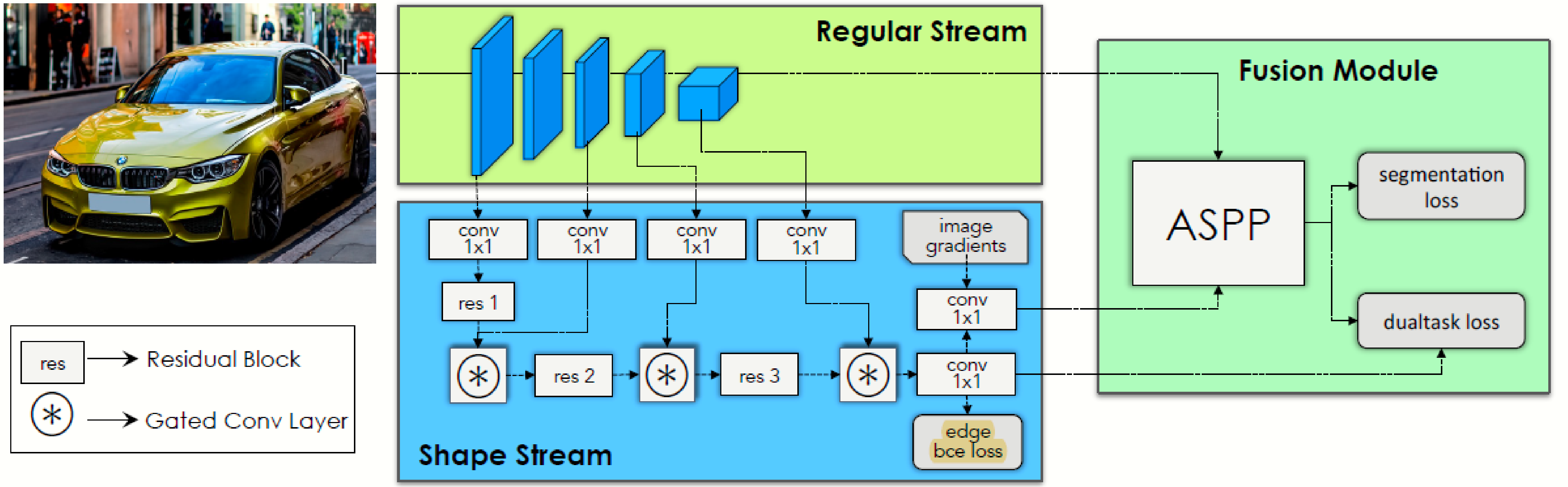
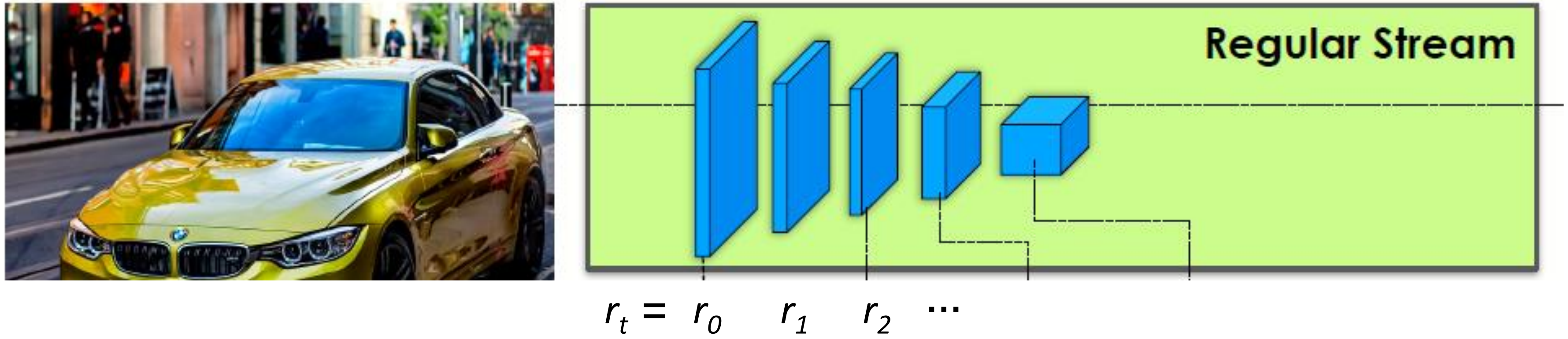


Figure 2: **GSCNN architecture.** Our architecture constitutes of two main streams. The regular stream and the shape stream. The regular stream can be any backbone architecture. The shape stream focuses on shape processing through a set of residual blocks, Gated Convolutional Layers (GCL) and supervision. A fusion module later combines information from the two streams in a multi-scale fashion using an Atrous Spatial Pyramid Pooling module (ASPP). High quality boundaries on the segmentation masks are ensured through a Dual Task Regularizer .

Chen et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs, IEEE TPAMI, 2017

Gated-SCNN

Regular stream



The regular stream can be any feedforward fully-convolutional network such as ResNet based or VGG based semantic segmentation network.

The authors make use of ResNet-like architecture such as ResNet-101 and WideResNet for the regular stream.

This stream, denoted as $\mathcal{R}_\theta(I)$

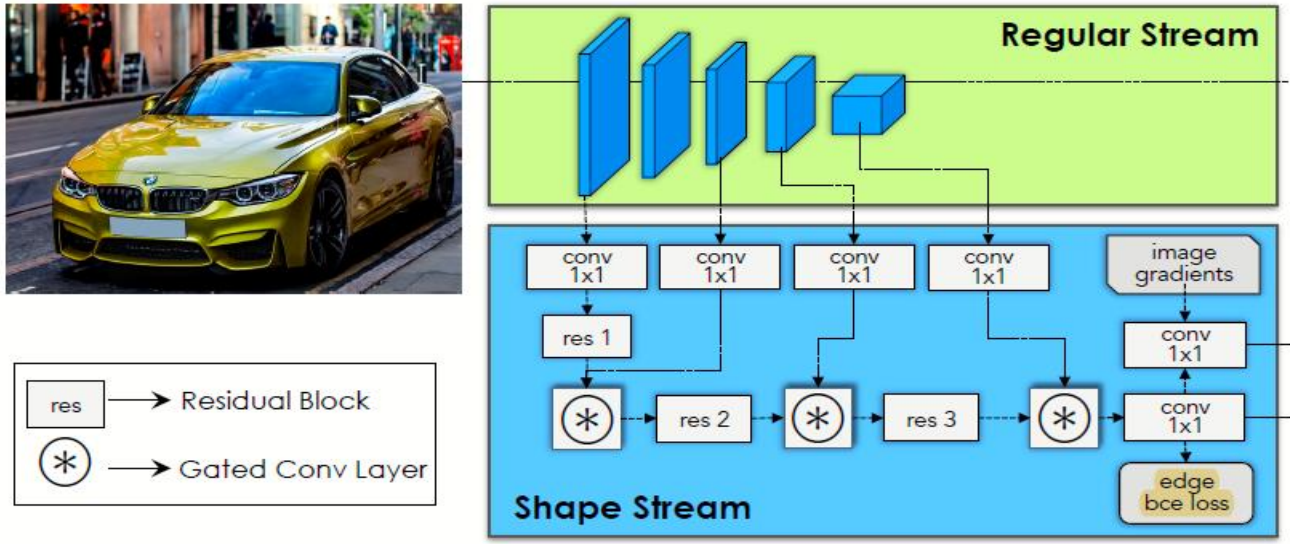
Input image: $I \in \mathbb{R}^{3\times H\times W}$

Output feature $r \in \mathbb{R}^{\dot{C}\times\frac{H}{m}\times\frac{W}{m}}$ where m is the stride of the regular stream

Intermediate representation of the regular stream: r_t

Gated-SCNN

Shape stream



The network architecture is composed of a few residual blocks interleaved with gated convolution layers (GCL)

Use supervised binary cross entropy loss on output boundaries to supervise the shape stream

This stream, denoted as $\mathcal{S}_\phi$.

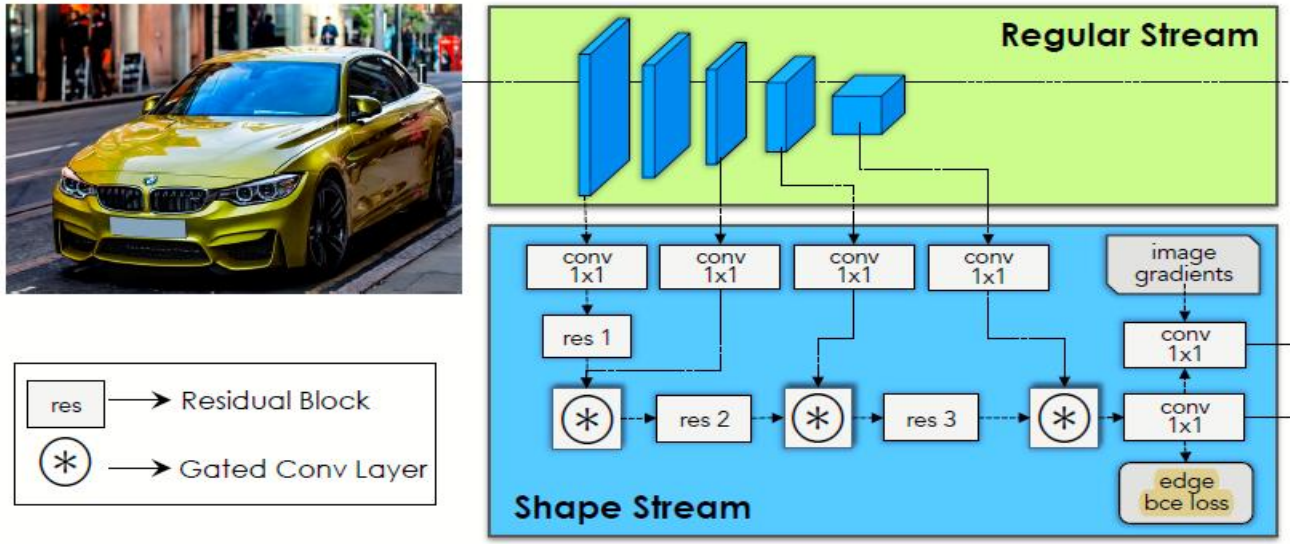
Input: (1) image gradients ∇I (2) Intermediate representation of the shape stream: r_t

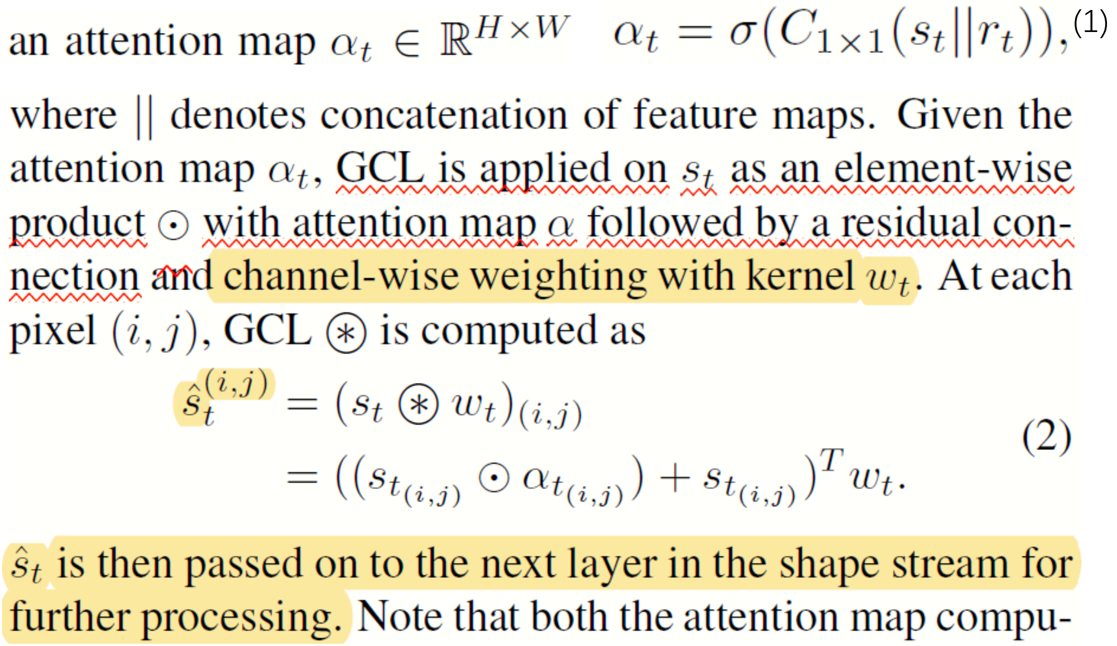
Output feature $s \in \mathbb{R}^{H \times W}$

Intermediate representation of the shape stream: S_t

Gated-SCNN

Shape stream
Gated convolutional layer (GCL)





an attention map $\alpha_t \in \mathbb{R}^{H\times W}$ $\quad \alpha_t = \sigma(C_{1\times 1}(s_t||r_t)),$ (1)

where || denotes concatenation of feature maps. Given the attention map α_t, GCL is applied on s_t as an element-wise product $\odot$ with attention map α followed by a residual connection and channel-wise weighting with kernel w_t. At each pixel (i, j), GCL $\circledast$ is computed as

$$\begin{aligned} \hat{s}_t^{(i,j)} &= (s_t \circledast w_t)_{(i,j)} \\ &= ((s_{t_{(i,j)}} \odot \alpha_{t_{(i,j)}}) + s_{t_{(i,j)}})^T w_t. \end{aligned} \qquad (2)$$

$\hat{s}_t$ is then passed on to the next layer in the shape stream for further processing. Note that both the attention map compu-

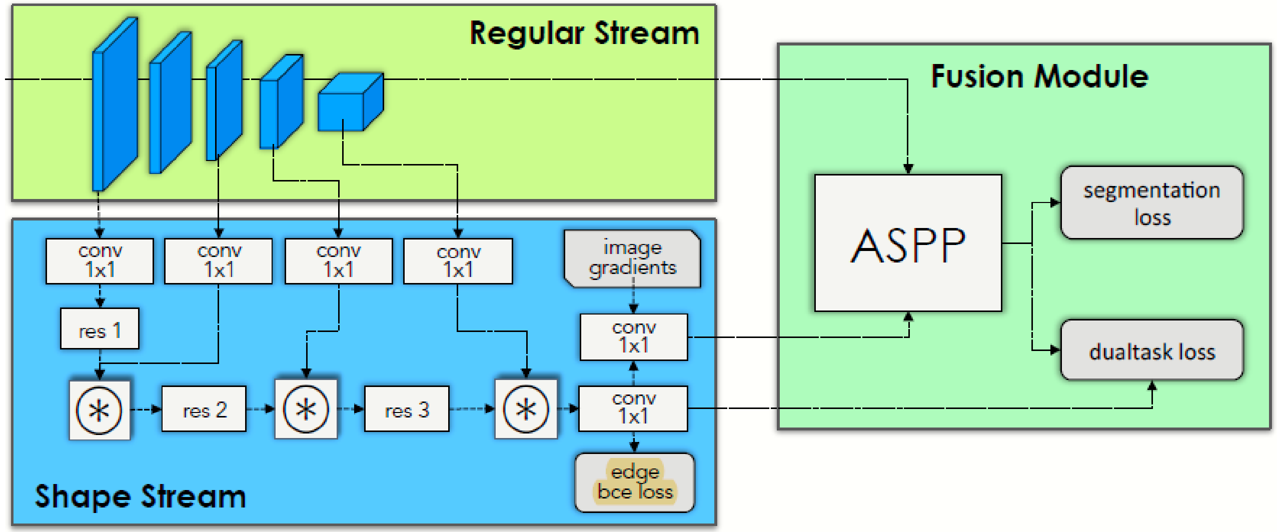
GCL ensures that the shape stream only processes boundary relevant information.

The shape stream does not incorporate features from the regular stream.
Rather, it uses GCL to deactivate its own activations that are not deemed relevant by the higher-level information contained in the regular stream.

GCL can be regarded as a collaboration between two streams, where the more powerful one, which has formed a higher-level semantic understanding of the scene, helps the other stream to focus only on the relevant parts since start.

Gated-SCNN

Fusion module



This module, denoted as $\mathcal{F}_\gamma$

Input:

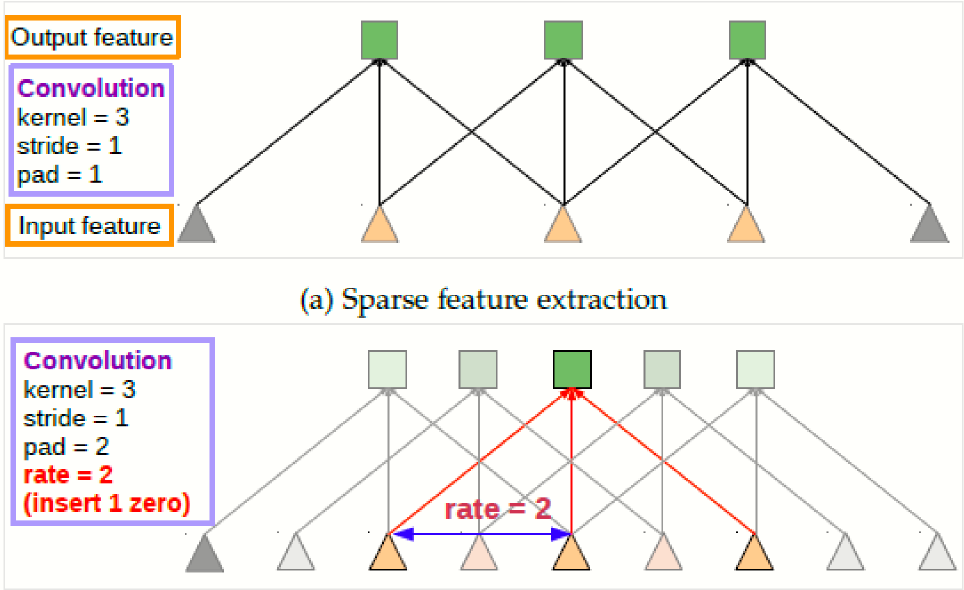
the dense feature representation ***r*** from regular stream + the boundary map ***S*** from shape stream

ASPP module Outputs a categorical distribution

$$f = p(y|s,r) = \mathcal{F}_\gamma(s,r) \in \mathbb{R}^{K\times H\times W}$$

Chen et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs, IEEE TPAMI, 2017

Gated-SCNN



(a) Sparse feature extraction

(b) Dense feature extraction

Fig. 2: Illustration of atrous convolution in 1-D. (a) Sparse feature extraction with standard convolution on a low resolution input feature map. (b) Dense feature extraction with atrous convolution with rate $r = 2$, applied on a high resolution input feature map.

The rate parameter r corresponds to the stride with which we sample the input signal. Standard convolution is a special case for rate r = 1.

Fusion module

ASPP(Atrous Spatial Pyramid Pooling)

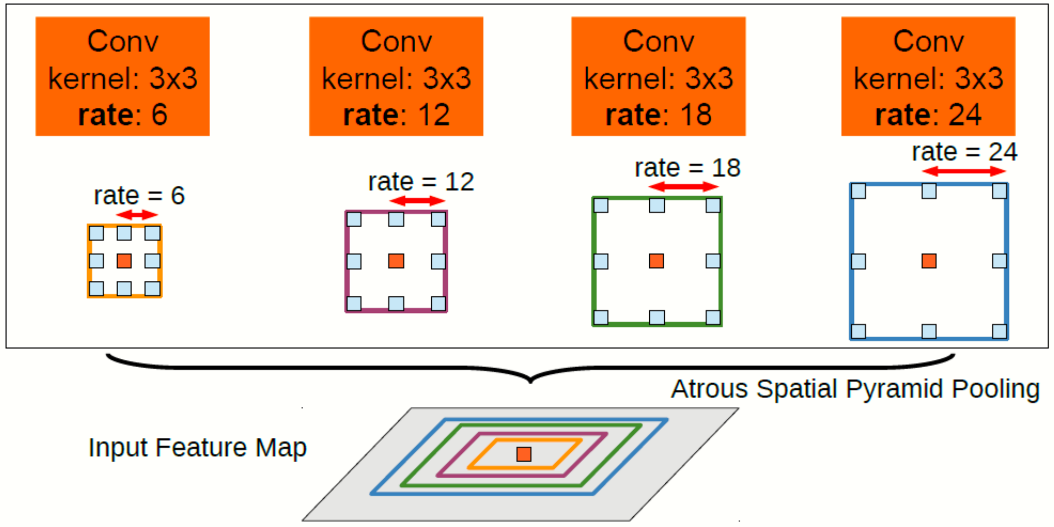


Fig. 4: Atrous Spatial Pyramid Pooling (ASPP). To classify the center pixel (orange), ASPP exploits multi-scale features by employing multiple parallel filters with different rates. The effective Field-Of-Views are shown in different colors.

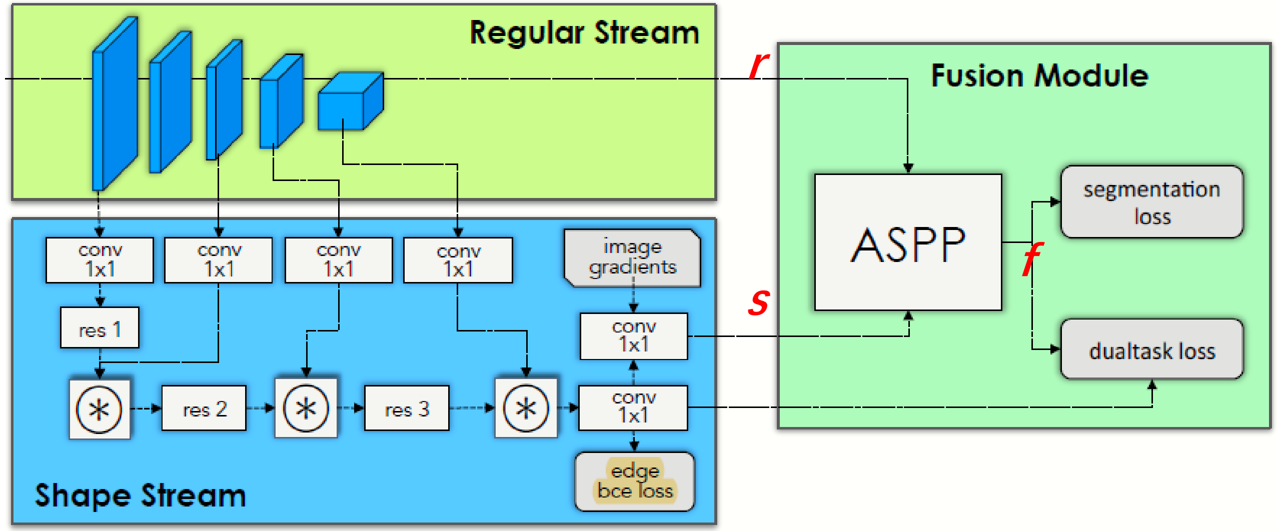
Combination everal parallel **atrous convolution** with different rates is called Atrous Spatial Pyramid Pooling or ASPP.

Atrous convolution allows us to arbitrarily enlarge the field-of-view of filters at any DCNN layer.

Chen et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs, IEEE TPAMI, 2017

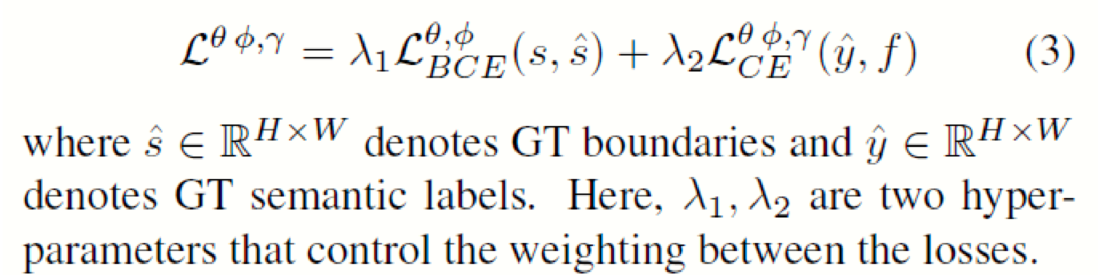
Gated-SCNN

Fusion module
Joint Multi Task Learning



The BCE supervision on boundary maps s is performed before feeding them into the fusion module. Thus *the BCE loss updates the parameters of both the regular and shape streams*.

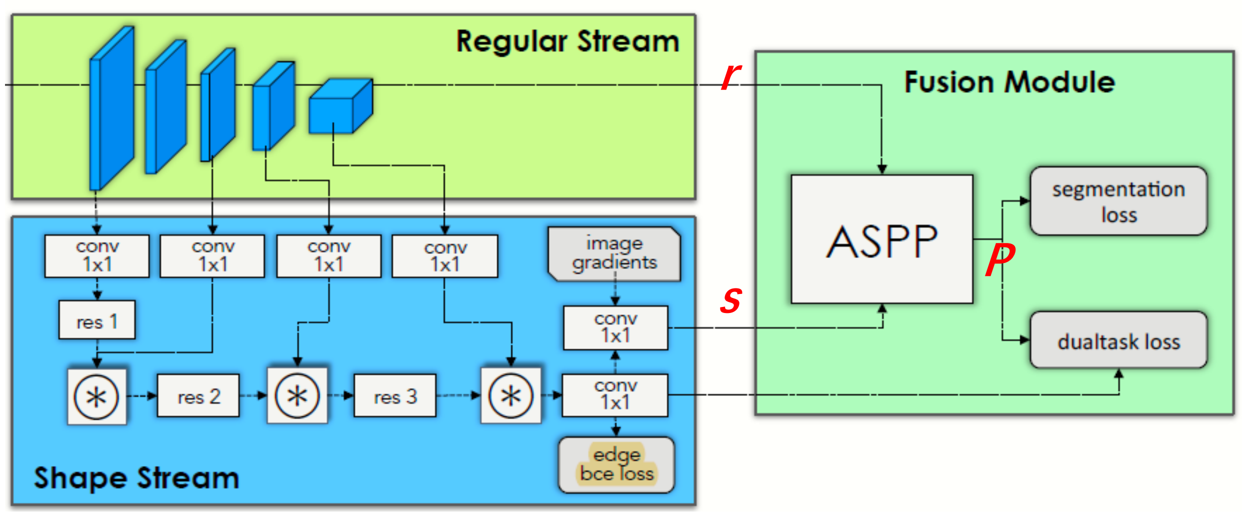
The final categorical distribution *f* of semantic classes is supervised with CE loss at the end as in standard semantic segmentation networks, *updating all the network parameters*



$$\mathcal{L}^{\theta\phi,\gamma} = \lambda_1 \mathcal{L}^{\theta,\phi}_{BCE}(s, \hat{s}) + \lambda_2 \mathcal{L}^{\theta\phi,\gamma}_{CE}(\hat{y}, f) \qquad (3)$$

where $\hat{s} \in \mathbb{R}^{H \times W}$ denotes GT boundaries and $\hat{y} \in \mathbb{R}^{H \times W}$ denotes GT semantic labels. Here, λ_1, λ_2 are two hyper-parameters that control the weighting between the losses.

Gated-SCNN



Regularizer 1

$\hat{\zeta}$ is a GT binary mask

$$\mathcal{L}^{\theta\,\phi,\gamma}_{reg_{\rightarrow}} = \lambda_3 \sum_{p^+} |\zeta(p^+) - \hat{\zeta}(p^+)| \qquad (5)$$

where p^+ contains the set of all non-zero pixel coordinates in both ζ and $\hat{\zeta}$.

→ ensure that boundary pixels are penalized when there is a mismatch with GT boundaries

→ to avoid non-boundary pixels to dominate the loss function.

Fusion module
Dual Task Regularizer

$$p(y|r,s) \in R^{K\times H\times W}$$

→ A categorical distribution output of the ASPP module

$$\zeta \in R^{H\times W} \quad \zeta = \frac{1}{\sqrt{2}}||\nabla(G * \arg\max_k p(y^k|r,s))|| \qquad (4)$$

where *G* denotes Gaussian filter

→ A potential that represents whether a particular pixel belongs to a semantic boundary in the input image *I*

Regularizer 2

$$\mathcal{L}^{\theta\,\phi,\gamma}_{reg_{\leftarrow}} = \lambda_4 \sum_{k,p} \mathbb{1}_{s_p}[\hat{y}^k_p \log p(y^k_p|r,s)], \qquad (6)$$

$$\mathbb{1}_s = \{1 : s > thrs\}$$

corresponds to the indicator function and thrs as a confidence threshold

→ ensure consistency between the binary boundary prediction ***s*** and the predicted semantics ***p(y|r; s)***.

The total dual task regularizer $\mathcal{L}^{\theta\,\phi,\gamma} = \mathcal{L}^{\theta\,\phi,\gamma}_{reg_{\rightarrow}} + \mathcal{L}^{\theta\,\phi,\gamma}_{reg_{\leftarrow}} \qquad (7)$

Experiments

Configuration

Experiment Details

Labeled dataset: Cityscapes dataset

This dataset contains images from 27 cities in Germany and neighboring countries. It contains 2975 training, 500 validation and 1525 test images.

Architecture: DeepLabV3+[1] as the main baseline.

Specifically, use ResNet-50, ResNet-101 and WideResNet as the backbone architecture for their version of DeeplabV3+.

Evaluation Metrics:

1 intersection over union (IoU) → evaluate whether the network accurately predicts regions $\mathcal{J} = \frac{|M \cap G|}{|M \cup G|}$

2 boundary metric F-score [2] → $\mathcal{F} = \frac{2P_c R_c}{P_c + R_c}$ (contour-based precision and recall) In experiments, they use thresholds correspond to 3, 5, 9, and 12 pixels respectively

3 distance-based evaluation in terms of IoU → evaluate the performance of the segmentation models at varying distances from the camera.

Training details:

We use 800*800 as thetraining resolution and synchronized batch norm.
For our joint loss, we set λ1 = 20, λ2 = 1, λ3 = 1 and λ4 = 1.

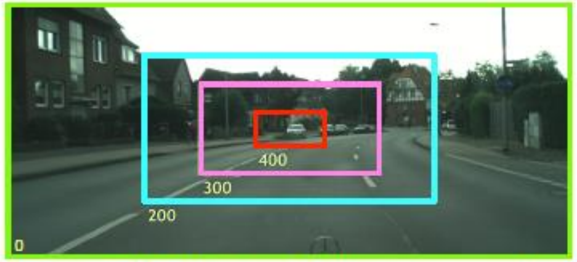


Figure 3: Illustration of the crops used for the distance-based evaluation.

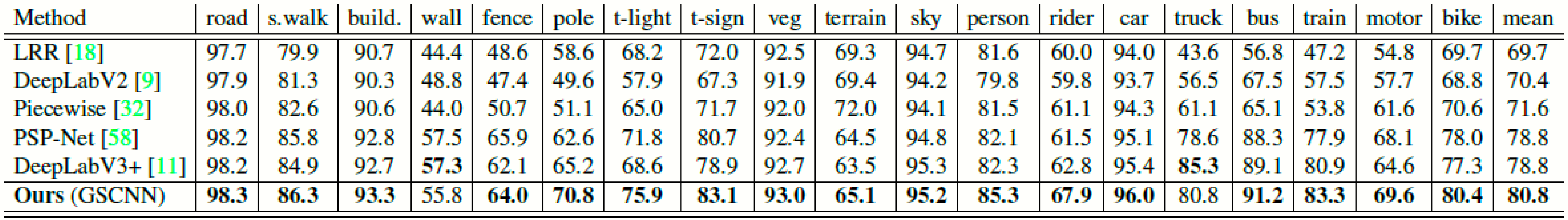
[1] L.-C. Chen, et al. Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. In ECCV, 2018

[2] F. Perazzi et al. A benchmark dataset and evaluation methodology for video object segmentation. In CVPR, 2016

Experiments

Quantitative Evaluation

Metric 1

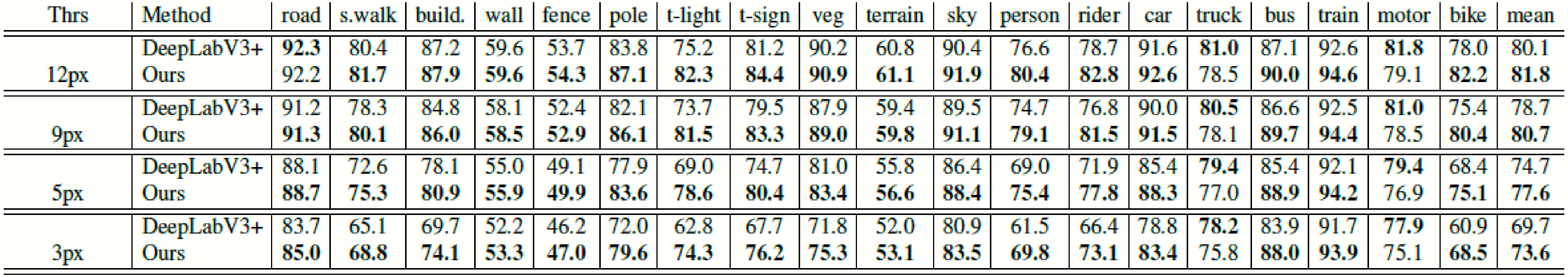


Method	road	s.walk	build.	wall	fence	pole	t-light	t-sign	veg	terrain	sky	person	rider	car	truck	bus	train	motor	bike	mean
LRR [18]	97.7	79.9	90.7	44.4	48.6	58.6	68.2	72.0	92.5	69.3	94.7	81.6	60.0	94.0	43.6	56.8	47.2	54.8	69.7	69.7
DeepLabV2 [9]	97.9	81.3	90.3	48.8	47.4	49.6	57.9	67.3	91.9	69.4	94.2	79.8	59.8	93.7	56.5	67.5	57.5	57.7	68.8	70.4
Piecewise [32]	98.0	82.6	90.6	44.0	50.7	51.1	65.0	71.7	92.0	72.0	94.1	81.5	61.1	94.3	61.1	65.1	53.8	61.6	70.6	71.6
PSP-Net [58]	98.2	85.8	92.8	57.5	65.9	62.6	71.8	80.7	92.4	64.5	94.8	82.1	61.5	95.1	78.6	88.3	77.9	68.1	78.0	78.8
DeepLabV3+ [11]	98.2	84.9	92.7	**57.3**	62.1	65.2	68.6	78.9	92.7	63.5	95.3	82.3	62.8	95.4	**85.3**	89.1	80.9	64.6	77.3	78.8
Ours (GSCNN)	**98.3**	**86.3**	**93.3**	55.8	**64.0**	**70.8**	**75.9**	**83.1**	**93.0**	**65.1**	**95.2**	**85.3**	**67.9**	**96.0**	80.8	**91.2**	**83.3**	**69.6**	**80.4**	**80.8**

Table 1: Comparison in terms of IoU vs state-of-the-art baselines on the Cityscapes val set.

In Table 1, we compare the performance of our GSCNN against the baselines in terms of region accuracy (measured by mIoU). The numbers are reported on the validation set, and computed on the full image. **In this metric, we achieve a 2% improvement, which is a significant result at this level of performance.**

Metric 2



Thrs	Method	road	s.walk	build.	wall	fence	pole	t-light	t-sign	veg	terrain	sky	person	rider	car	truck	bus	train	motor	bike	mean
12px	DeepLabV3+	**92.3**	80.4	87.2	59.6	53.7	83.8	75.2	81.2	90.2	60.8	90.4	76.6	78.7	91.6	**81.0**	87.1	92.6	**81.8**	78.0	80.1
	Ours	92.2	**81.7**	**87.9**	**59.6**	**54.3**	**87.1**	**82.3**	**84.4**	**90.9**	**61.1**	**91.9**	**80.4**	**82.8**	**92.6**	78.5	**90.0**	**94.6**	79.1	**82.2**	**81.8**
9px	DeepLabV3+	91.2	78.3	84.8	58.1	52.4	82.1	73.7	79.5	87.9	59.4	89.5	74.7	76.8	90.0	**80.5**	86.6	92.5	**81.0**	75.4	78.7
	Ours	**91.3**	**80.1**	**86.0**	**58.5**	**52.9**	**86.1**	**81.5**	**83.3**	**89.0**	**59.8**	**91.1**	**79.1**	**81.5**	**91.5**	78.1	**89.7**	**94.4**	78.5	**80.4**	**80.7**
5px	DeepLabV3+	88.1	72.6	78.1	55.0	49.1	77.9	69.0	74.7	81.0	55.8	86.4	69.0	71.9	85.4	**79.4**	85.4	92.1	**79.4**	68.4	74.7
	Ours	**88.7**	**75.3**	**80.9**	**55.9**	**49.9**	**83.6**	**78.6**	**80.4**	**83.4**	**56.6**	**88.4**	**75.4**	**77.8**	**88.3**	77.0	**88.9**	**94.2**	76.9	**75.1**	**77.6**
3px	DeepLabV3+	83.7	65.1	69.7	52.2	46.2	72.0	62.8	67.7	71.8	52.0	80.9	61.5	66.4	78.8	**78.2**	83.9	91.7	**77.9**	60.9	69.7
	Ours	**85.0**	**68.8**	**74.1**	**53.3**	**47.0**	**79.6**	**74.3**	**76.2**	**75.3**	**53.1**	**83.5**	**69.8**	**73.1**	**83.4**	75.8	**88.0**	**93.9**	75.1	**68.5**	**73.6**

Table 2: Comparison vs baselines at different thresholds in terms of boundary F-score on the Cityscapes val set.

Table 2, on the other hand, compares the performance of our method against the baseline in terms of boundary accuracy (measured by F score). **Similarly, our model performs considerably better, outperforming the baseline by close to 4% in the strictest regime**.

Experiments

Quantitative Evaluation

Metric 3

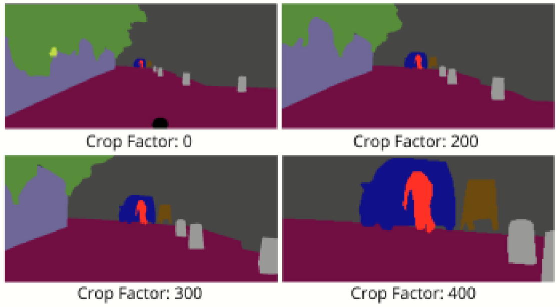


Figure 4: Predictions at diff. crop factors.

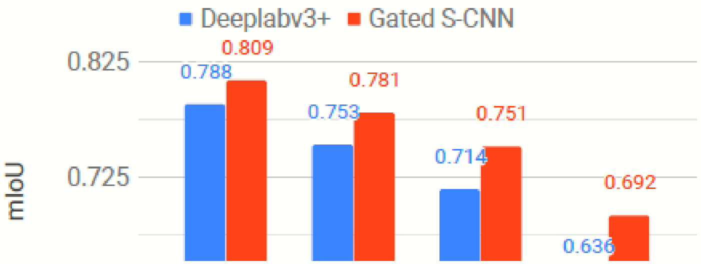
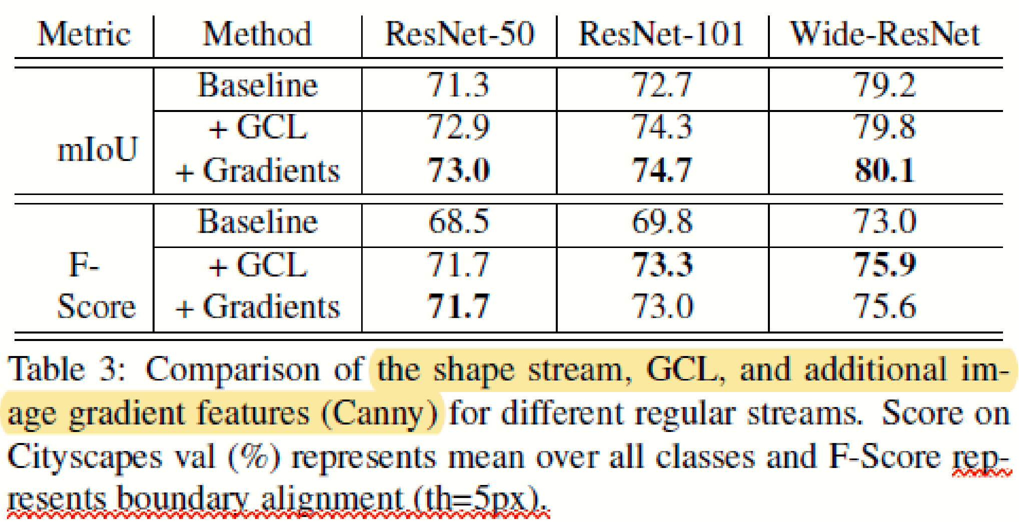


Figure 5: **Distance-based evaluation**: Comparison of mIoU at different crop factors.

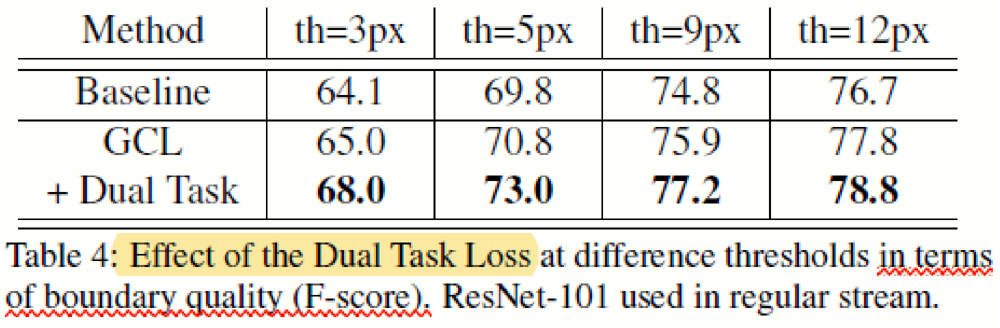
In Fig 5, the authors show the performance of theor method vs baseline following the proposed distance-based evaluation method. Here, they find that GSCNN performs increasingly better as the crop factor increases.

Ablation Study



Metric	Method	ResNet-50	ResNet-101	Wide-ResNet
mIoU	Baseline	71.3	72.7	79.2
	+ GCL	72.9	74.3	79.8
	+ Gradients	**73.0**	**74.7**	**80.1**
F-Score	Baseline	68.5	69.8	73.0
	+ GCL	71.7	**73.3**	**75.9**
	+ Gradients	**71.7**	73.0	75.6

Table 3: Comparison of the shape stream, GCL, and additional image gradient features (Canny) for different regular streams. Score on Cityscapes val (%) represents mean over all classes and F-Score represents boundary alignment (th=5px).



Method	th=3px	th=5px	th=9px	th=12px
Baseline	64.1	69.8	74.8	76.7
GCL	65.0	70.8	75.9	77.8
+ Dual Task	**68.0**	**73.0**	**77.2**	**78.8**

Table 4: Effect of the Dual Task Loss at difference thresholds in terms of boundary quality (F-score). ResNet-101 used in regular stream.

Table 3 **the effectiveness of each component of the method using different encoder networks for the regular stream**.
Here, **GCL** denotes a network trained with the shape stream with dual task loss, and **Gradients** denotes the network that also adds image gradients before the fusion layer. Adding GCL achieve between 1 to 2 % improvement in performance in terms of mIoU, and around 3 % for F-score(boundary alignment)

Table 4 **shows the effect of the Dual Task loss in terms of F-score for boundary alignment**.
Here, GCL denotes the network with the GCL shape stream trained without Dual Task Loss.
Concretely, by adding back the Dual-Task loss, we see up to 3% improvement in terms of F-socre.

Qualitative Results

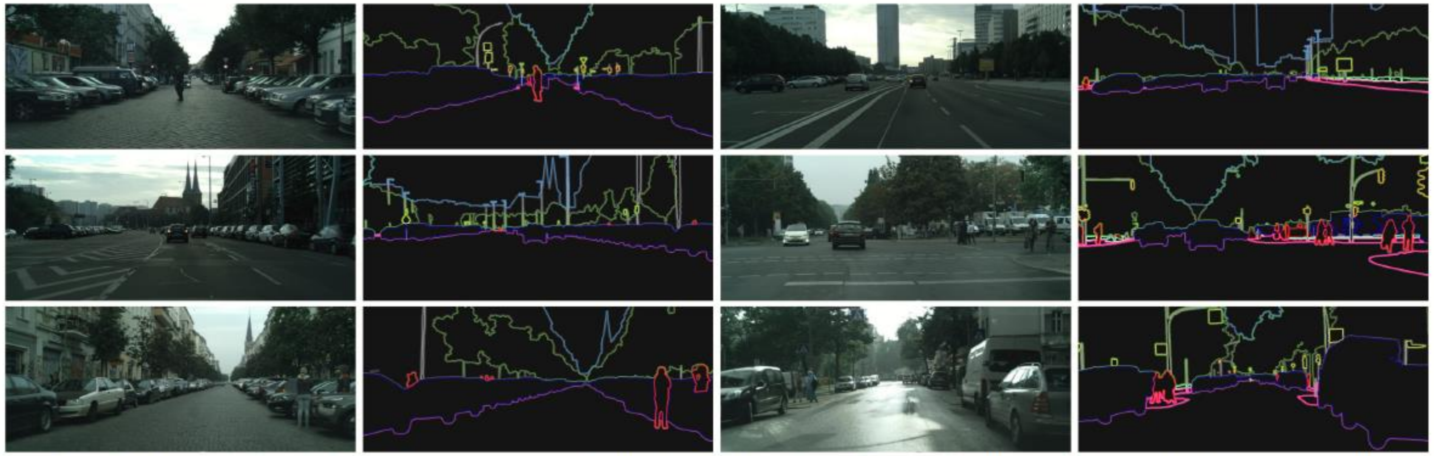


Figure 9: Qualitative results on the Cityscapes test set showing the high-quality boundaries of our predicted segmentation masks. Boundaries are obtained by finding the edges of the predicted segmentation masks.

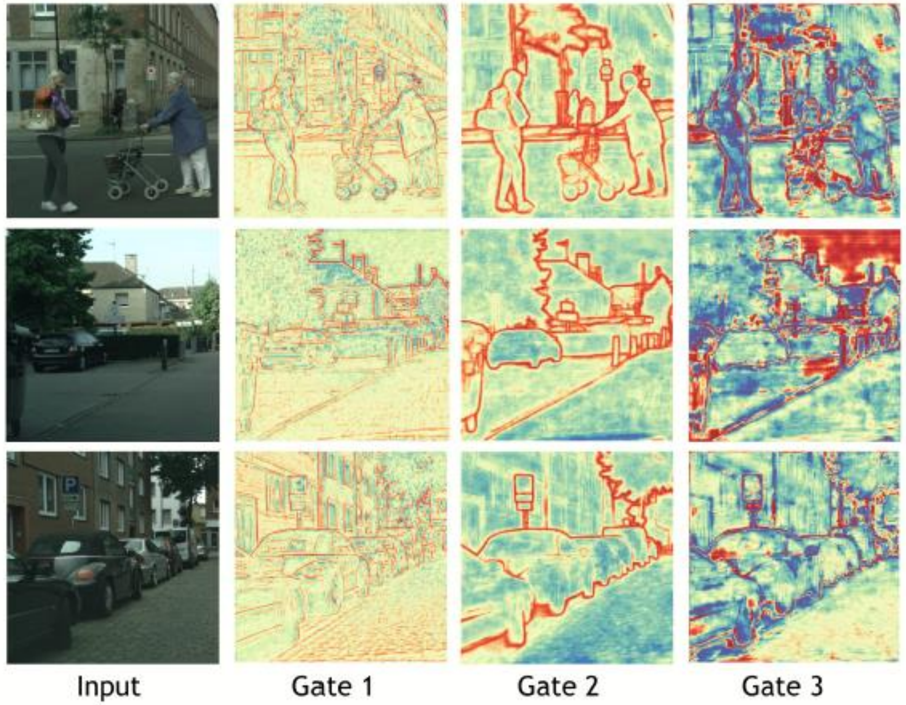


Figure 10: Visualization of the alpha channels from the GCLs.

Figure 9 shows the boundaries obtained from the final segmentation masks. High accuracy especially on the thinner and smaller objects.

Fig 10 provides a visualization of the alpha channels from the GCL.

For example, the first gate emphasized very low-level edges while the second and third focus on object-level boundaries

Thanks！