

Multi-Scale Feature Aggregation by Cross-Scale Pixel-to-Region Relation Operation for Semantic Segmentation

Yechao Bai¹, Ziyuan Huang¹,
 Lyuyu Shen¹, Hongliang Guo², Marcelo H. Ang Jr¹ and Daniela Rus³

Abstract—Exploiting multi-scale features has shown great potential in tackling semantic segmentation problems. The aggregation is commonly done with sum or concatenation (concat) followed by convolutional (conv) layers. However, it fully passes down the high-level context to the following hierarchy without considering their interrelation. In this work, we aim to enable the low-level feature to aggregate the complementary context from adjacent high-level feature maps by a cross-scale pixel-to-region relation operation. We leverage cross-scale context propagation to make the long-range dependency capturable even by the high-resolution low-level features. To this end, we employ an efficient feature pyramid network to obtain multi-scale features. We propose a Relational Semantics Extractor (RSE) and Relational Semantics Propagator (RSP) for context extraction and propagation respectively. Then we stack several RSP into an RSP head to achieve the progressive top-down distribution of the context. Experiment results on two challenging datasets Cityscapes and COCO demonstrate that the RSP head performs competitively on both semantic segmentation and panoptic segmentation with high efficiency. It outperforms DeeplabV3 [1] by 0.7% with 75% fewer FLOPs (multiply-adds) in the semantic segmentation task.

Index Terms—Deep Learning for Visual Perception, Semantic Scene Understanding, Automation Technologies for Smart Cities.

I. INTRODUCTION

Semantic segmentation is a fundamental task in computer vision that has various important applications in self-driving car, robotics, etc. Great advancement has been achieved since the advent of deep neural networks. Lots of works have shown that effective integration of contextual information plays a central role in pushing forward the segmentation performance [2], [3], [4], [5], [6], [7], [8], [9]. Contextual information implies the relational connection between an object and a region which facilitates the classification of the object.

As the output of the layers of the CNN backbone encodes different scales and levels of contextual information which

*This work was supported by the National Research Foundation, Prime Minister's Office, Singapore, under its CREATE program, Singapore-MIT Alliance for Research and Technology (SMART) Future Urban Mobility (FM) IRG. (Yechao Bai and Ziyuan Huang are co-first authors.) (Corresponding authors: Yechao Bai.)

¹Yechao Bai, Ziyuan Huang, Lyuyu Shen and Marcelo H. Ang Jr are with the Department of Mechanical Engineering, National University of Singapore, Singapore. {yechao.bai, ziyuan.huang, e0444150}@u.nus.edu, mpeangh@nus.edu.sg

²Hongliang Guo is with the Singapore-MIT Alliance for Research and Technology (SMART) Centre, Singapore. hongliang@smart.mit.edu

³Daniela Rus is with the Massachusetts Institute of Technology, Cambridge, MA, USA. rus@csail.mit.edu

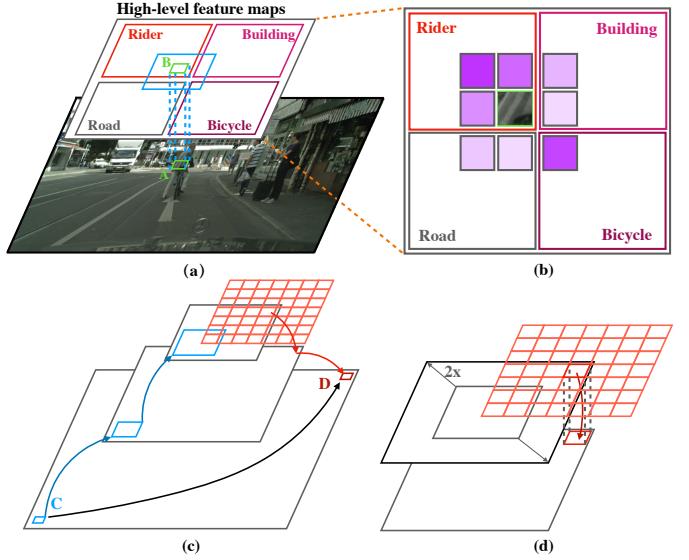


Fig. 1: (a) Cross-scale pixel-to-region relation. This demonstrates that not all context from high-level feature maps is beneficial to the classification of the small portion of a rider in A. (b) The proposed relation operation emphasizes the related context (deeper purple) and suppresses the unrelated context (lighter purple) from the corresponding region of the high-level feature map. (c) The blue arrows represent the context extraction. The pink grids indicate the region that the feature on the adjacent low-level feature maps search and aggregate complementary context. The red arrows represent the propagation of the context. The black arrow implies that feature D in essence captures long-range dependencies from feature C. (d) The high-level feature map is firstly upsampled to the same spatial dimension as the adjacent low-level feature map. The region on the upsampled high-level feature map is centered at the feature at the same spatial position as the low-level feature.

combine to form a feature pyramid, it emerges as a natural choice to leverage this multi-scale feature pyramid to achieve a high quality yet efficient context fusion. The multi-scale feature aggregation is commonly done with sum or concat followed by conv layers with a pixel-to-pixel correspondence. However, it fully passes down the high-level context to the following hierarchy without considering their interrelation. For example in Fig. 1 (a) and (b), not all context information in the predefined vicinity of the corresponding high-level feature B is beneficial to the classification of the low-level feature A (a portion of a rider). Ideally, feature A should

discriminately aggregate feature that contains the high-level context, emphasize the semantically related and spatially close features from Rider and Bicycle, and suppress the others.

To this end, we propose a Relational Semantics Extractor (RSE) inspired by [10] to enable the low-level feature to extract the complementary relational context from adjacent high-level feature maps by using a cross-scale pixel-to-region relation operation. The key insight is that the proposed local relation operation essentially learns the composability between objects in the adjacent feature maps. The spatial relation is established by adding a positional embedding [11]. On top of RSE, we present the Relational Semantics Propagator (RSP) to propagate the extracted relational context. To progressively propagate the high-level context in a top-down manner, we construct an RSP head by stacking several RSP modules as in Fig. 2. As illustrated in Fig. 1(c) and (d), our simple and efficient model architecture allows each low-level feature to search and aggregate context information from a large region in the high-level feature map. The blue arrow and red arrow indicate the context extraction and context propagation respectively, together with a top-down progressive contextual information propagation and a large relation operation region. We essentially enable the low-level feature D to capture long-range dependencies from another low-level feature C. In summary, our contributions are:

- 1) Propose a cross-scale pixel-to-region relation operation as an effective solution to multi-scale feature aggregation.
- 2) Propose a Relational Semantics Extractor (RSE) and Relational Semantics Propagator (RSP) for context extraction and propagation respectively.
- 3) Introduce a simple, light-weight yet effective RSP head for semantic segmentation which performs competitively on Cityscapes and COCO.

II. RELATED WORK

Multi-scale feature aggregation. Following the earlier work [2], various successful approaches have been developed based on exploiting multi-scale feature aggregation. Methods like [3], [13], [14], [1] and [4] extract multi-scale features with pyramid pooling and atrous spatial pyramid pooling respectively. On the other hand, Lin et al. [15] exploit the natural structure of the deep networks for the construction of multi-scale semantics. A recently proposed network [12] upsamples the multi-scale feature pyramid to the same spatial dimension through lateral paths and fuse them by element-wise summation. Chen et al. [16] use image pyramids of different scales as input, then use a CNN trunk to fuse multi-scale information by weighted summation. These methods use either concatenation or element-wise summation during fusion which propagates all high-level contextual information to the lower level. Besides, they conduct multi-scale feature aggregation based on a pixel-to-pixel correspondence and do not consider their interrelation.

Attention. Attention-based methods have shown great potential in computer vision. Wang et al. [17] demonstrate that long-range dependencies are beneficial to classification. Parmar et al. [18] takes one step further and shows that in the image

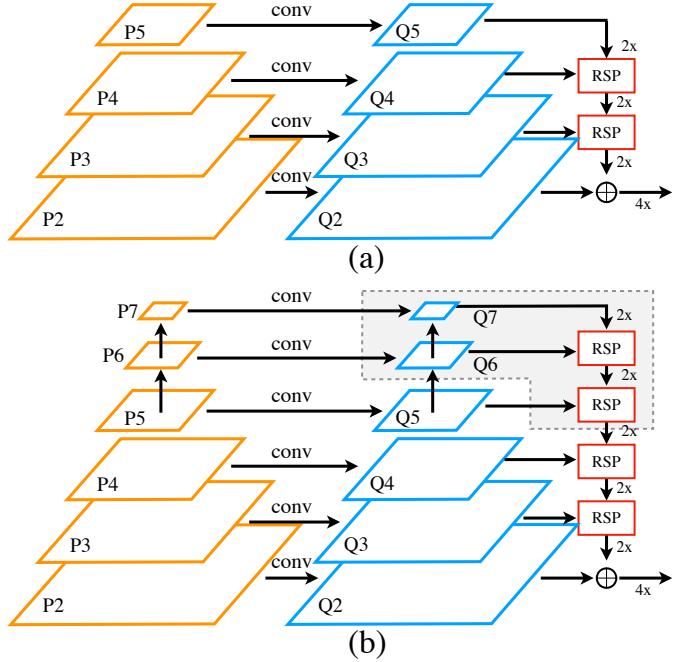


Fig. 2: **The structure of RSP head.** (a) RSP-2 head. (B) RSP-4 head. To fully exploit the semantic propagation ability of the RSP, we additionally use $\{P_6, P_7\}$ in the FPN. Four RSP modules are connected to aggregate the features from Q_3 to Q_7 , which are transformed from $\{P_3, \dots, P_7\}$ as in [12]. We do not apply RSP to the fusion of Q_2 and Q_3 for efficiency.

recognition task, the convolutional kernel can be replaced by a form of self-attention operation. Compositional relationship between pixels in a local neighborhood is exploited in [10], [19] to meaningfully join elements together, and it highlights that a meaningful fusion is determined by the similarity of two pixels' feature projections into a learned embedding space [11]. Recent work [20] applies a non-local operation to compare feature maps from two scale levels for feature enhancement. Our work extends local relation operation to cross-scale settings to learn the multi-scale composability to achieve a meaningful aggregation of information from multiple scale levels. A coarse-to-fine approach [5] exploits the coarse prediction to obtain a class center feature as context and then use it to enhance the coarse prediction. Yuan et al. [6] aggregates context from object regions in an image through a coarse prediction and distributes it back to all spatial positions based on the relationships between the feature position and the context representation [21]. These approaches leverage the pixel-to-region relation to extracting the context but are restrained to a single scale level whereas our method aggregates the related context in a cross-scale setting. Ding et al. [8] produce a context map for each pixel with a paired convolution and Gaussian kernel in a large predefined region. Then apply the mask to the weights of conv operations to make it shape-variant. Due to the computation cost, the shape-variant context-mask restrained to a single layer of low-resolution.

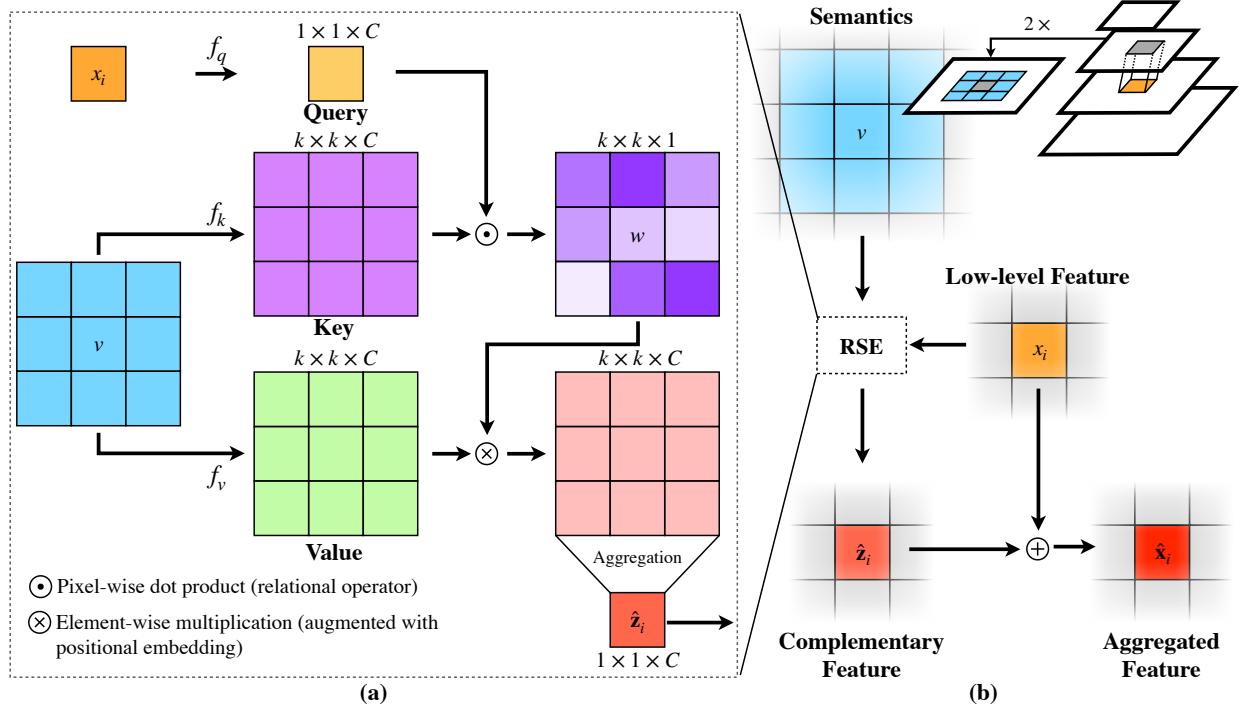


Fig. 3: (a) Cross-scale Relational semantics extractor (RSE) (b) Relational semantics propagator (RSP). The key insight is that the relation operation extracts complementary features from the key and the value and passes them to the query. We exploit this property to extract complementary information from the corresponding region in the high-level feature map w.r.t. to the low-level feature.

III. APPROACH

A. Relational Semantics Extractor

To address the inefficiency of the convolutional layer in modeling the compositional relationships, [10] propose to explicitly exploit relations between different pixels to extract meaningful features with relation operation. One key insight is that the proposed local relation operation essentially learns the composability between objects in the key map and the query map. The local relation operation obtains the key and value from the same region. In contrast, we propose a relational semantics extractor (RSE) to exploit the property of relation operation to enable the low-level feature map to selectively extract complementary context from its adjacent high-level feature map with a pixel-to-region correspondence as shown in Fig. 3. Formally, the operations in relational semantics extractor are defined as follows, given the upsampled high level feature map $\mathbf{z} \in \mathbb{R}^{H \times W \times C}$ and the low level feature map $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$:

$$\hat{\mathbf{z}}_i = \Phi(f_q(\mathbf{x}_i), f_k(\mathcal{L}(\mathbf{x}_i, \mathbf{z})) \otimes f_v(\mathcal{L}(\mathbf{x}_i, \mathbf{z})) \quad (1)$$

where $\hat{\mathbf{z}} \in \mathbb{R}^{H \times W \times C}$ is the output feature map, \mathbf{x}_i and $\hat{\mathbf{z}}_i \in \mathbb{R}^{1 \times 1 \times C}$ is the feature of a specific pixel at location i in \mathbf{x} and $\hat{\mathbf{z}}_i$ respectively. Φ is the relation operator, which looks for composability between the input pixel \mathbf{x}_i and the defined adjacent region of \mathbf{x}_i . $\mathcal{L}(a, b)$ extracts adjacent region of pixel a in feature map b . f_q, f_k and f_v denotes linear transformations that project the features into the embedding space. If we define the kernel size of RSE as k , $\mathbf{v} = \mathcal{L}(\mathbf{x}_i, \mathbf{z})$ extracts the feature matrix $\mathbf{v} \in \mathbb{R}^{k \times k \times C}$, with the center of \mathbf{v} at the

same location with \mathbf{x}_i , as visualized in Fig. 3(b). \odot is a pixel-wise dot product with broadcasting in channel dimension if required. Following [10], [11], we denote $f_q(\mathbf{x}_i)$, $f_k(\mathcal{L}(\mathbf{x}_i, \mathbf{z}))$ and $f_v(\mathcal{L}(\mathbf{x}_i, \mathbf{z}))$ as the *query*, *key* and *value* respectively in Fig. 3(a). To reduce the computation overhead, we reduce the channel number of the key and query in f_q and f_k by a factor of d . The relation operator Φ computes appearance compositability and is defined as the dot product of the feature pairs:

$$\mathbf{w} = \Phi(\mathbf{x}_i, \mathcal{L}(\mathbf{x}_i, \mathbf{z})) = \text{Concat}_{\forall \mathbf{x}_j \in \mathcal{L}(\mathbf{x}_i, \mathbf{z})} (\mathbf{x}_i \cdot \mathbf{x}_j) \quad (2)$$

where the output of Φ is a weight matrix $\mathbf{w} \in \mathbb{R}^{k \times k \times 1}$. For simplicity, we omit the linear transformation in the formula. There are other forms of relation modeling but their performance is similar [10], [19], therefore we adopt the dot product by default for implementation efficiency.

Since the current formulation does not encode positional information and is thus permutation invariant, additional positional embedding is required. We follow a similar strategy for positional embedding in [18]. In our case, a normalized 2D relative position map goes through its own linear transformation before the embedding is included in the relation operation. The 2D relative position is generated as $\mathbf{p} \in \mathbb{R}^{k \times k \times C}$, where the first $\frac{C}{2}$ channels are row offset and the second half is column offset. The normalization process projects the values between -1 and 1 . The relation operator Φ' with positional embedding can be now defined as:

$$\Phi'(f_q(\mathbf{x}_i), f_k(\mathcal{L}(\mathbf{x}_i, \mathbf{z})) = \Phi(f_q(\mathbf{x}_i), f_k(\mathcal{L}(\mathbf{x}_i, \mathbf{z}))) + f_p(\mathbf{p}) \quad (3)$$

where $f_p(\mathbf{p})$ denotes the linear transformation of the relative position map \mathbf{p} .

B. Relational Semantics Propagation Head

With the RSE that extracts complementary semantic information to the low-level feature maps, RSP propagates the information to the low-level feature map. With the element-wise addition shown in Fig. 3(b), we achieve scale fusion with only selected semantics. Specifically, the aggregation process can be expressed as:

$$\hat{\mathbf{x}}_i = \mathbf{x}_i + \hat{\mathbf{z}}_i \quad (4)$$

where the $\hat{\mathbf{z}}_i$ is the extracted relational semantics as in Eq. 1.

Compared to performing element-wise summation for multi-scale feature aggregation, the proposed RSE has two advantages. (A) During aggregation, summation only considers pixel-wise correspondence, while in RSP, information in a larger semantics region is aggregated to one pixel location from the low-level feature map. (B) Instead of propagating all contextual information from the high level semantic features, RSE selectively extracts useful features with respect to the low-level features.

We construct the RSP head by stacking a number of RSP modules. The overall structure of the RSP head can be seen in Fig. 2. Since the RSP is able to propagate the high level information to the low level feature maps, we follow [22], [23] and leverage a 7-level FPN structure [15]. Specifically, $\{P_2, P_3, P_4, P_5\}$ are generated by connecting a 1×1 convolution to the feature maps of different stages in ResNet [24], and $\{P_6, P_7\}$ are obtained by applying strided 3×3 convolution to P_5 and P_6 respectively. For more details please refer to [23].

We denote the transformed feature map from $\{P_2, \dots, P_7\}$ as $\{Q_2, \dots, Q_7\}$, and progressively aggregate the feature maps from the highest level to the lowest level. The high level feature map is first upscaled by a factor of two before it is fed into the RSP. For clarity, we denote the basic version of the RSP head without $\{Q_6, Q_7\}$ as RSP-2. The element-wise summations between level Q_5, Q_4 and level Q_4, Q_3 are replaced by the proposed RSP module. The full RSP head with 4 RSP modules is denoted as RSP-4. All the fusion between two scale levels are replaced with the RSP module except for the one between Q_3 and Q_2 , where we perform only simple summation for avoiding high computations. In our experiments, we also show that the aggregation of higher scale features using RSP yields better results.

IV. EXPERIMENTS

A. Implementation Details

Baseline Network. The baseline network adopts FPN as the backbone for multi-scale feature extraction. The baseline for RSP-2 uses $\{Q_2, Q_3, Q_4, Q_5\}$ with strides of $\{4, 8, 16, 32\}$ pixels with respect to the input image. Additional $\{Q_6, Q_7\}$ are used in the baseline for RSP-4 with strides of $\{64, 128\}$ pixels. Our baseline networks aggregate the features with a pixel-to-pixel correspondence. It starts from the highest

level $Q_5(\text{RSP-2})/Q_7(\text{RSP-4})$ and gradually approaches Q_2 by upsampling the high level feature map to match the spatial dimension of the following low-level feature map with bilinear upsampling and then apply element-wise summation. A final 1×1 convolution, $4 \times$ bilinear upsampling, and softmax are used to generate the per-pixel class labels at the original image resolution.

Cityscapes. The Cityscapes dataset [25] is tasked for urban scene understanding with 19 categories for semantic segmentation evaluation. The dataset contains 5,000 high resolution pixel-level finely annotated images and 20,000 coarsely annotated images. The finely annotated images are divided into 2,975/500/1,525 images for training, validation and testing.

COCO. The COCO dataset [26] is challenging large scale dataset for computer vision tasks. The panoptic segmentation task [27] uses all 2017 COCO images with 80 thing and 53 stuff classes annotated. As we integrate the proposed semantic segmentation head to the panoptic FPN, we evaluate our approach in the panoptic segmentation task. We use mIoU as the evaluation metric for semantic segmentation and also report PQ, Mask AP and Box AP.

Training details. On Cityscapes, we follow [12] and use SGD with 0.9 momentum with 32 images per mini-batch cropped to a fixed 512×1024 size; the training schedule is 40K/15K/10K updates at learning rates of 0.01/0.0001/0.0001 respectively; a linear learning rate warmup [28] over 1000 updates starting from a learning rate of 0.001 is applied; a weight decay of 0.0001 is applied; horizontal flipping, color augmentation [29], and crop bootstrapping [30] are used during training; scale train-time data augmentation resizes an input image from $0.5 \times$ to $2.0 \times$ with a 32 pixel step; BN layers are frozen; no test-time augmentation is used. The evaluation metric is mIoU (mean Intersection-over-Union). On COCO dataset, we use the default Mask R-CNN $1 \times$ training setting [31] with scale jitter (shorter image side in [640, 800]).

Loss function. For semantic segmentation, we use the per-pixel cross entropy loss. For panoptic segmentation, we follow [12] and use the weighted sum of the instance segmentation loss and the semantic segmentation loss, $L = \lambda_i(L_c + L_b + L_m) + \lambda_s L_s$. The semantic segmentation loss weight is set to be $\lambda_s = 0.5$ and instance segmentation loss weight is set to be $\lambda_i = 1$.

B. Performance Comparisons.

Semantic segmentation. We compare RSP with existing semantic segmentation methods on Cityscapes *val* set. Only *fine* annotation is used for training and the mIoU is evaluated *without* using flip and multi-scale testing. We first compare with Semantic FPN [27] on Cityscapes *val* as our RSP head is most similar to the Semantic FPN [12]. The results are shown in Table I. 'D' in model name indicates use of dilated kernel of size 3 and dilation 3, the detail is in Section IV-C. RSP-2 outperforms Semantic FPN with $\sim 15\%$ fewer FLOPs. It is worth noting that RSP-4 with ResNet-50-FPN backbone already achieves 77.5% mIoU, which is very close to the result of Semantic FPN 77.7% mIoU with the heavier ResNet-101 backbone. Next, we compare RSP-4 with other top-performing

TABLE I: **Semantic segmentation results on Cityscapes *val* set.** Only *fine* Cityscapes annotations are used for training. ‘D’ in the model name indicates the use of dilated kernel of size 3 and dilation 3. The median and standard deviation of 5 random runs are reported and the best results are in bold. Note that RSP-4 with ResNet-50-FPN backbone achieves a very close performance to Semantic FPN[12] with ResNet-101-FPN backbone. FLOPs (multiply-adds $\times 10^9$) and the number of parameters are only calculated for the *head* i.e. backbone excluded.

Model	Backbone	mIoU	FLOPs	# param.
Baseline	ResNet-50-FPN	74.8	51.7G	4.7M
RSP-2	ResNet-50-FPN	76.1 ± 0.2	53.4G	5.1M
RSP-4	ResNet-50-FPN	77.5 ± 0.2	53.7G	7.8M
Baseline	ResNet-101-FPN	76.7	51.7G	4.7M
RSP-2-D	ResNet-101-FPN	77.9 ± 0.2	53.4G	5.1M
RSP-4-D	ResNet-101-FPN	78.5 ± 0.2	53.7G	7.8M
Semantic FPN[12]	ResNet-50-FPN	75.8	62.5G	6.5M
Semantic FPN[12]	ResNet-101-FPN	77.7	62.5G	6.5M

methods. The results are shown in Table II. Note that, RSP-4 outperforms DeeplabV3 [1] by 0.7% with 75% fewer FLOPs when using the same backbone. Our approach, which is simple in design, is able to perform on par with DeepLabV3+ which have undergone many design iterations. RSP-4 achieves strong results compared to state-of-the-art method OCR [6] with lighter FLOPs.

Panoptic segmentation. Next, we conduct experiments to compare with the semantic segmentation branch in the panoptic segmentation task on COCO *val* set by replacing the semantic segmentation branch with the RSP head. The results are shown in Table III. RSP improves the semantic segmentation performance mIoU by a large margin and this also leads to improvement in the panoptic segmentation metric PQ.

C. Ablation Study on Cityscapes

Ablation study of the RSP-2 head. We break down the improvements of RSP-2 over the baseline, by adding RSP modules to the baseline one-by-one. The results are shown in Table IV. Adding RSP module consistently improves the

TABLE II: **Performance comparisons on Cityscapes *val* set.** Only *fine* Cityscapes annotations are used for training. The mIoU is evaluated w/o using flip and multi-scale testing. ‘D’ in model name indicates use of dilated kernel of size 3 and dilation 3. The backbone notation includes the dilated resolution ‘D’. FLOPs (multiply-adds) and memory (# activations) are calculated for the *whole* model i.e. includes backbone and head. Memory are approximate but informative.

Model	Backbone	mIoU	FLOPs	memory.
Semantic FPN[12]	ResNet-101-FPN	77.7	0.5T	0.8G
DeeplabV3 [1]	ResNet-101-D8	77.8	1.9T	1.9G
PSANet101 [32]	ResNet-101-D8	77.9	2.0T	2.0G
SETR-PUP [33]	T-Large	79.3	1.0T	2.7G
Mapillary [30]	WideResNet-38-D8	79.4	4.3T	1.7G
DeeplabV3+ [4]	X-71-D16	79.6	0.5T	1.9G
OCR [6]	HRNetV2	80.8	1.3T	1.4G
RSP-4-D	ResNet-101-FPN	78.5	0.5T	0.8G
RSP-4-D	ResNeXt-101-FPN	79.5	0.8T	1.4G

TABLE III: **Panoptic segmentation results on COCO *val*.** The backbone is ResNet-50-FPN. ‘D’ in model name indicates use of dilated kernel of size 3 and dilation 3. The backbone notation includes the dilated resolution ‘D’. In the second row and third row we replace the original semantic segmentation branch in the Panoptic FPN with our RSP-2-D and RSP-4-D head respectively. The FLOPs (multiply-adds $\times 10^9$) and number of parameters are calculated for the head. only.

Model	mIoU	PQ	Mask AP	Box AP	FLOPs	# param
Panoptic FPN [12]	41.3	39.4	34.6	37.5	62.5G	6.5M
RSP-2-D head	41.9	40.1	34.6	37.5	53.1G	5.1M
RSP-4-D head	42.7	40.2	34.5	37.5	53.4G	7.8M

baseline. With 2 RSP modules ($\sim 3\%$ computation increment), the RSP head achieves a 1.3 mIoU improvement over the baseline. In the experiment + *SELF*, we replace all the cross-scale relation operations with local relation operation [10], and the input is only the high-level feature map. Compared to + *SELF*, RSP achieves much better performance because the cross-scale setting of our relation operation enables the low-level feature to access context from a much larger region. In the experiment + *CONTEXT*, we propagate high-level semantic information by simply aggregating high-level features in a local receptive field by average pooling and add it to the low-level feature. This outperforms the baseline but not our RSP-2. It proves the superiority of our proposed relation operation in extracting meaningful context information from the high-level feature map.

Ablation study of RSP Module. We analyze the effect of kernel sizes in the RSP module, as shown in Table V. The RSP-2 achieves the best performance when the kernel size is 7 and dilation is 1. Meanwhile, RSP obtains a similar result for kernel size 3 and dilation 3, where the effective kernel size is also 7. Therefore, we decide to adopt kernel size 7 and dilation 1 when using backbone ResNet-50 for a better performance, and kernel size 3 and dilation 3 when using larger backbone ResNet-101 since using the dilation reduces the number of computation as well as the GPU memory. For the number of middle channels, we choose the dimension reduction factor as

TABLE IV: **Effect of the number of RSP modules in the RSP-2 head with backbone ResNet-50-FPN.** $\times 1$ and $\times 2$ indicates the number of RSP modules employed. In column RSP and Sum, (54,43) means employing RSP/Sum between level Q_5, Q_4 and Q_4, Q_3 . In + *SELF* $\times 2$, we replace the cross-scale relation operation with the local relation operation [10]. In + *CONTEXT* $\times 2$ we replace the pair-wise relation operation in RSE with the averaging operation. *For reference purpose. We re-train the semantic FPN [12] with the same training settings as our RSP head.

Model	RSP	Sum	mIoU	FLOPs	# param.
BASELINE	-	(54, 43)	74.8	51.7G	4.7M
+ RSP $\times 1$	54	43	75.2(+0.4)	52.0G	4.9M
+ RSP $\times 1$	43	54	75.6(+0.8)	53.1G	4.9M
+ RSP $\times 2$	(54, 43)	-	76.1(+1.3)	53.4G	5.1M
+ SELF $\times 2$	-	-	75.5(+0.7)	53.4G	5.1M
+ CONTEXT $\times 2$	-	-	75.2(+0.4)	51.7G	4.7M

TABLE V: The effect of the kernel sizes and dimension reduction factor d in the RSP module. Experiments are conducted with RSP-2 head on the backbone ResNet-50-FPN.

(a) **Performance with different RSE kernel size.** We alter the kernel size and the dilation of the kernels in the RSE to discover the optimal setting. K and D indicates the kernel size and dilation respectively.

(K, D)	mIoU	FLOPs	# param.
(3, 1)	75.5	53.1G	5.1M
(5, 1)	75.5	53.2G	5.1M
(7, 1)	76.1	53.4G	5.1M
(3, 2)	75.6	53.1G	5.1M
(3, 3)	76.0	53.1G	5.1M

(b) **Performance with different number of middle channels.** The output channels of the value transformation f_v are 128. As mentioned before, dimension reduction is applied in f_q and f_v to reduce the channel number by a factor of d before further operations. We indicate the number of middle channels in brackets.

d	mIoU	FLOPs	# param.
1 (128)	75.9	53.8G	5.3M
2 (64)	76.1	53.4G	5.1M
4 (32)	75.6	53.0G	5.0M
8 (16)	76.0	52.8G	4.9M

2 for its better performance.

Ablation study of RSP-4 head. We study effect of the number of RSP modules in the RSP-4 head, results are in Table VI. We have three observations. 1) Increasing the depth improves the performance even the additional higher-level features are aggregated by element-wise summation. This confirms that high-level semantics is beneficial to classification. 2) Increasing the number of RSP modules consistently improves the performance. 3) With the same number of RSP modules employed, start adding RSP modules from the highest-level generally gives a better result than from the lowest level. This proves that the proposed RSP successfully meets our design goal to propagate the complementary contextual information in a top-down manner.

D. Qualitative Evaluation.

Complementary information in relation operation. As shown in [10], the feature extracted by the query and key trans-

TABLE VI: Effect of the number of RSP module in the RSP-4 head with backbone ResNet-50-FPN. $\times 1$ -4 indicates the number of RSP module employed. In column RSP and Sum, (54,43) means employing RSP/Sum between level Q_5 , Q_4 and Q_4 , Q_3 .

Model	RSP	Sum	mIoU	FLOPs	# param.
BASELINE	-	(54, 43)	74.8	51.7G	4.7M
+ Q6, Q7	-	(76, 65, 54, 43)	76.0(+1.2)	51.9G	7.1M
+ RSP \times 1	43	(76, 65, 54)	76.3(+1.5)	53.2G	7.3M
+ RSP \times 1	76	(65, 54, 43)	76.2(+1.4)	51.9G	7.3M
+ RSP \times 2	(54, 43)	(76, 65)	76.4(+1.6)	53.6G	7.4M
+ RSP \times 2	(76, 65)	(54, 43)	76.9(+2.1)	52.0G	7.4M
+ RSP \times 3	(65, 54, 43)	76	77.1(+2.3)	53.7G	7.6M
+ RSP \times 3	(76, 65, 54)	43	77.2(+2.4)	52.3G	7.6M
+ RSP \times 4	(76, 65, 54, 43)	-	77.5(+2.7)	53.7G	7.8M

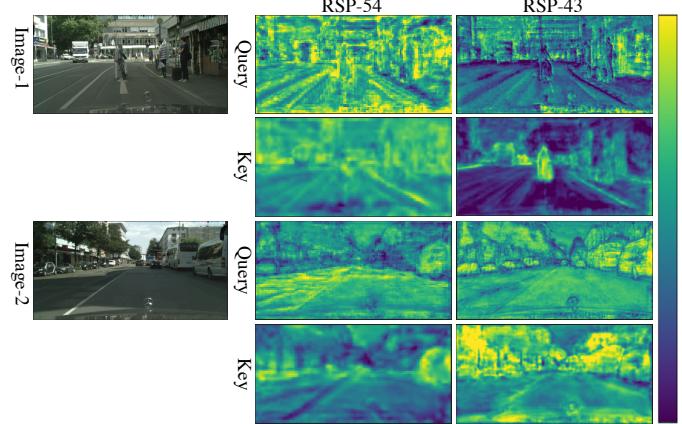


Fig. 4: **Illustration of learnt key and query.** RSP-AB indicates RSP module placed between Q_A and Q_B . The complementary property between the query and key features visualised here is the core insight that we leverage to extract complementary information w.r.t the query map from the value map.

formation complement each other. In our case, we demonstrate that in cross-scale relation operation, this observation stands. As is visualized in Fig. 4, both key-query feature pairs in RSP-54 and RSP-43 complement each other.

Qualitative results on Cityscapes. We provide the qualitative comparisons between the RSP-4 and the baseline network with ResNet-50-FPN in the upper part of Fig. 5(a). We use the red box to mark those challenging regions. The baseline model misclassifies the sidewalk near the crowd as the road in the first image, the portion of the rider far from the motorcycle as a person in the second image, and pixels at the boundary of a bus as the car in the last image. In contrast, the proposed RSP head classifies all those areas correctly. The rider case demonstrates that the RSP head enables long-range dependencies to be captured. The sidewalk and bus case confirms that the RSP head allows the pixel at the boundary to select the helpful high-level context.

Visualization of the context propagation. We visualize and compare the feature maps from the same channel produced by RSP-4 and the baseline during the whole feature aggregation process on two images Fig. 5(b). In the left image, half part of the rider not near the bicycle is misclassified as the person. In the right image, the pixels at the boundary of the car and bus are misclassified. The feature maps that display the context propagation show that the baseline model fully passes down the high-level context which includes wrong or incomplete context whereas the RSP-4 successfully reject those context and aggregate the complementary and informative context. In the last row, we use white circles to highlight the features produced by the RSP-4 and baseline model that represents rider and car in the red box. The RSP-4 produces complete and clear features which is much easier to be discriminated against.

V. CONCLUSION

In this work, we propose a relational multi-scale feature aggregation approach for semantic segmentation. The multi-

scale feature aggregation is achieved through the proposed relational semantics propagator (RSP) head, where the high-level context is selectively propagated to the low-level feature maps with a pixel-to-region correspondence. We propose a cross-scale relation operation named relational semantics extractor (RSE) to extract complementary contextual information w.r.t. the low-level feature from the corresponding region of adjacent high-level feature maps. The cross-scale setting also enables the low-level features to capture long-range dependency in a compute-efficient way. Extensive experiments show the effectiveness of the RSP module and the consistent improvement by adding multiple RSP modules.

ACKNOWLEDGMENT

We thank Feng Xue and Guirong Zhuo for their helpful discussion and generous support and the Institute of Intelligent Vehicles, School of Automotive Studies, Tongji University for providing the GPUs for experiments.

REFERENCES

- [1] L. Chen, G. Papandreou, F. Schroff, and H. Adam, “Rethinking atrous convolution for semantic image segmentation,” *arXiv preprint arXiv:1706.05587*, 2017. [1](#), [2](#), [5](#)
- [2] W. Liu, A. Rabinovich, and A. Berg, “ParseNet: Looking wider to see better,” *arXiv preprint arXiv:1506.04579*, 2015. [1](#), [2](#)
- [3] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890. [1](#), [2](#)
- [4] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818. [1](#), [2](#), [5](#)
- [5] F. Zhang, Y. Chen, Z. Li, Z. Hong, J. Liu, F. Ma, J. Han, and E. Ding, “ACFNet: Attentional class feature network for semantic segmentation,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 6798–6807. [1](#), [2](#)
- [6] Y. Yuan, X. Chen, and J. Wang, “Object-contextual representations for semantic segmentation,” *arXiv preprint arXiv:1909.11065*, 2019. [1](#), [2](#), [5](#)
- [7] A. Tao, K. Sapra, and B. Catanzaro, “Hierarchical multi-scale attention for semantic segmentation,” *arXiv preprint arXiv:2005.10821*, 2020. [1](#)
- [8] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang, “Semantic correlation promoted shape-variant context for segmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8885–8894. [1](#), [2](#)
- [9] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, “CCNet: Criss-cross attention for semantic segmentation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. [1](#)
- [10] H. Hu, Z. Zhang, Z. Xie, and S. Lin, “Local relation networks for image recognition,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 3464–3473. [2](#), [3](#), [5](#), [6](#)
- [11] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zamzad, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, *et al.*, “Relational inductive biases, deep learning, and graph networks,” *arXiv preprint arXiv:1806.01261*, 2018. [2](#), [3](#)
- [12] A. Kirillov, R. Girshick, K. He, and P. Dollár, “Panoptic feature pyramid networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6399–6408. [2](#), [4](#), [5](#)
- [13] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017. [2](#)
- [14] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Semantic image segmentation with deep convolutional nets and fully connected CRFs,” *arXiv preprint arXiv:1412.7062*, 2014. [2](#)
- [15] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125. [2](#), [4](#)
- [16] L. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, “Attention to Scale: Scale-aware semantic image segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3640–3649. [2](#)
- [17] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7794–7803. [2](#)
- [18] N. Parmar, P. Ramachandran, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, “Stand-alone self-attention in vision models,” in *Advances in Neural Information Processing Systems*, 2019, pp. 68–80. [2](#), [3](#)
- [19] H. Zhao, J. Jia, and V. Koltun, “Exploring self-attention for image recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10076–10085. [2](#), [3](#)
- [20] D. Zhang, H. Zhang, J. Tang, M. Wang, X. Hua, and Q. Sun, “Feature pyramid transformer,” *arXiv preprint arXiv:2007.09451*, 2020. [2](#)
- [21] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, “A²-Nets: Double attention networks,” in *Advances in neural information processing systems*, 2018, pp. 352–361. [2](#)
- [22] Z. Tian, C. Shen, H. Chen, and T. He, “FCOS: Fully convolutional one-stage object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2019, pp. 9627–9636. [4](#)
- [23] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988. [4](#)
- [24] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. [4](#)
- [25] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, “The cityscapes dataset for semantic urban scene understanding,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223. [4](#)
- [26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in *European conference on computer vision*. Springer, 2014, pp. 740–755. [4](#)
- [27] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, “Panoptic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 9404–9413. [4](#)
- [28] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, “Accurate, Large Minibatch SGD: Training imagenet in 1 hour,” *arXiv preprint arXiv:1706.02677*, 2017. [4](#)
- [29] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in *European conference on computer vision*. Springer, 2016, pp. 21–37. [4](#)
- [30] S. Rota Bulò, L. Porzi, and P. Kotschieder, “In-place activated batch-norm for memory-optimized training of DNNs,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5639–5647. [4](#), [5](#)
- [31] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He, “Detectron,” <https://github.com/facebookresearch/detectron>, 2018. [4](#)
- [32] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia, “PSANet: Point-wise spatial attention network for scene parsing,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 267–283. [5](#)
- [33] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr, *et al.*, “Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers,” *arXiv preprint arXiv:2012.15840*, 2020. [5](#)

void	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation
terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle

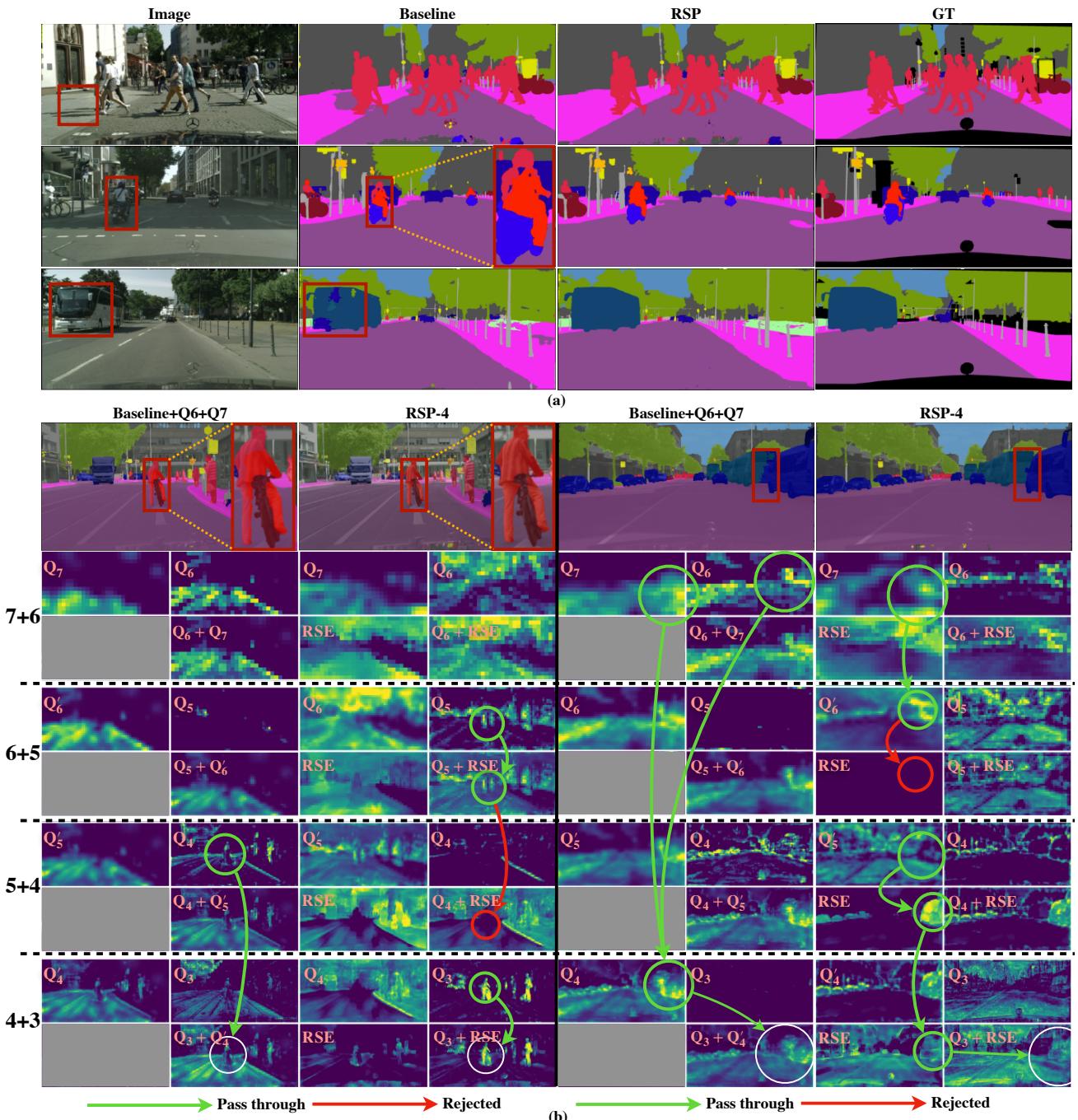


Fig. 5: (a) **Qualitative results on Cityscapes.** The challenging area for the baseline model is at the places where a transition from one class to another class occurs consequently multiple context information is available in those areas. The ability to select the right context in our proposed RSP-4 plays a key role in making the correct classification. (b) **Visualization of the context propagation.** We visualize and compare the feature maps from the same channel produced by RSP-4 and the baseline during the whole feature aggregation process on two images. Q_x follows the notation defined before IV-A. RSE box shows the features extracted by the RSE operation. Gray box means no relational features are extracted. $Q_x + Q_{x+1}/\text{RSE}$ box displays the aggregated cross-scale features. Q'_x box shows the upsampled aggregated features, which means $Q'_x = \text{upsampling}(Q_x + Q_{x+1}/\text{RSE})$. The fusion between Q_3 and Q_2 is not visualized as the structures of RSP and the baseline are the same in this part. The green path indicates the passage of features, while the red path indicates the areas where the high-level semantic features are rejected by RSE.