Motion and Depth Augmented Semantic Segmentation for Autonomous Navigation

Hazem Rashed[1], Ahmad El Sallab[1], Senthil Yogamani[2] and Mohamed ElHelw[3]
[1]Valeo R&D Cairo, Egypt , [2]Valeo Vision Systems, [3]Nile University, Cairo, Egypt
{hazem.rashed, ahmad.el-sallab@valeo.com, senthil.yogamani}@valeo.com, melhelw@nu.edu.eg

Abstract

Motion and depth provide critical information in autonomous driving and they are commonly used for generic object detection. In this paper, we leverage them for improving semantic segmentation. Depth cues can be useful for detecting road as it lies below the horizon line. There is also a strong structural similarity for different instances of different objects including buildings and trees. Motion cues are useful as the scene is highly dynamic with moving objects including vehicles and pedestrians. This work utilizes geometric information modelled by depth maps and motion cues represented by optical flow vectors to improve the pixel-wise segmentation task. A CNN architecture is proposed and the variations regarding the stage at which color, depth, and motion information are fused, e.g. early-fusion or mid-fusion, are systematically investigated. Additionally, we implement a multimodal fusion algorithm to maximize the benefit from all the information. The proposed algorithms are evaluated on Virtual-KITTI and Cityscapes datasets where results demonstrate enhanced performance with depth and flow augmentation.

1. Introduction

Recently, semantic segmentation has gained huge attention especially after the emergence of deep learning. Semantic segmentation aims at associating a certain class to each pixel in the surrounding scene. For an autonomous vehicle, this task is crucial to fully perceive the surrounding environment and react accordingly. With the significant progress in embedded devices and their computation power, semantic segmentation is gaining more attention in practical applications. The majority of semantic segmentation algorithms rely only on color information captured by a camera sensor to label each pixel with a specific class. However, such algorithms do not make use of the specific conditions associated with autonomous driving scenes with regard to motion or geometry. In this work, we make use of depth maps and optical flow and study the impact of these signals on semantic segmentation task through constructing fusion CNNs.

The main contributions of this paper include:

1. Implementation of CNN based fusion architecture to study the benefit of depth and flow signals over standard RGB images for semantic segmentation.
2. Comparison between early-fusion and mid-fusion algorithms using both synthetic and real automotive datasets, namely Virtual-KITTI and Cityscapes.
3. Empirical study to evaluate the proposed approach using different depth and optical flow estimation algorithms.
4. Proposing a multimodal fusion CNN to utilize the three modalities for semantic segmentation.

The rest of the paper is organized as follows: review of the related work is presented in Section 2, the network architecture is detailed in Section 3, the experimental setup and results are illustrated in Section 4 and finally Section 5 concludes the paper.

2. Related Work

2.1. Semantic Segmentation

Semantic segmentation has been a challenging problem in recent years and several approaches were investigated for this task. In [3] [2], patch-wise training was adopted to perform classification. In [3] a Laplacian pyramid was used while in [7] a deep network was utilized to avoid post-processing. [12] exploited end-to-end training using fully-convolutional network to learn heatmaps that were upsampled to finally reach the original image size at the classification stage. In SegNet [1], deconvolution networks were used for upsampling while keeping the corresponding max-pooling indices from the encoder. Several CNN approaches used only images as input for the segmentation task.

2.2. Depth Estimation for Autonomous Driving

Depth is a very important cue that implicitly encodes scene and object geometry information. Knowing only color semantics without depth is not enough for full environment perception. In addition to depth sensors such as LIDAR, depth can be estimated using several camera-only approaches. The most common way for camera-based depth estimation is based on stereo images using the SGM [8] algorithm where pixels from the left and right image are matched across epipolar lines. Disparity is subsequently computed according to a cost function that guarantees the smoothness of the output surface. Structure from Motion (SfM) approaches use monocular camera but suffer from limited performance when dealing with moving objects. Several CNN methods were used to estimate depth using monocular cameras such as in [19][11]. Unsupervised approaches [6][5] are very beneficial for autonomous driving applications due to the lack of annotated datasets that generalize to multiple scene structures.

2.3. Flow Estimation for Autonomous Driving

Optical flow estimation is a standard module in recent commercial vehicles and with the rapid progress of embedded computing power, dense optical flow is currently being computed instead of sparse flow maps. However, optical flow estimation is still a challenging task due to the highly dynamic environment of autonomous driving. In this case, the scene exhibits two types of motion, ego-motion and object motion. For accurate optical flow estimation, ego-motion compensation has to be carried out. CNN methods have also been introduced to learn scene motion such as [10] which performs generic foreground object segmentation, [14] which performs pixel-wise segmentation and [17][15] which carries out motion segmentation. CNNs have also been used to estimate optical flow [9][13] [20].

3. Methodology

A novel CNN architecture for semantic segmentation is proposed in this section. The variations regarding the stage at which color, depth and motion information are fused together result in a slightly different network for which performance is investigated. To this end, four network architectures were implemented:

- Unimodal architecture which provides baseline segmentation results for each information signal separately.
- Early-Fusion architecture which fuses two signals prior to feature extraction and extracts joint features via CNN.
- Mid-Fusion architecture which extracts features for two modalities using a CNN extractor, then fuse the information on the feature level.
- Multimodal Mid-Fusion architecture which fuses color, motion, depth information altogether using the Mid-Fusion approach.

3.1. Unimodal Architecture

Our baseline network is based on the encoder-decoder architecture of FCN8s [12] which utilizes the VGG [18] for feature extraction in the encoder part. The fully connected layers of the VGG are removed and replaced by three upsampling layers to reconstruct the original image size. Skip connections are used to extract high resolution features and avoid losing information while reducing size of the image due to maxpooling layers. 1x1 convolutions are used to adjust depth of feature maps to be able to add to deeper layers. Softmax is used to determine the likelihood for the output to belong to a certain class and cross-entropy is utilized for loss function. We make use of this network to evaluate the performance of semantic segmentation using a single piece of information, i.e. RGB, depth or flow-only.

3.2. Early-Fusion Architecture

This architecture fuses raw signals prior to feature extraction. The network input in this case is 3D volume comprising four layers. Three of them are RGB layers and the fourth layer is optical flow or a depth map. Several optical flow representations have been studied experimentally using this network, namely Colorwheel representation in 3 channels, magnitude and direction in 2 channels and magnitude only in 1 channel. The depth map consists of 1 layer which is concatenated directly to RGB image. The input channels are normalized from 0 to 255 to have the same values as the RGB image. The first layer of the network is adapted so that it accepts an input of four channels and the corresponding weights are initialized randomly while the pre-trained VGG weights are used in the rest of the network. This architecture allows us to perform fusion for the input information without increase in complexity.

3.3. Mid-Fusion Architecture

A methodology similar to [17] is adopted where we construct a mid-fusion network that performs feature extraction for each modality separately then uses feature-level fusion to combine both cues together. Fusion is done at the last layer of the VGG while skip connections are utilized to avoid losing information due to down sampling. This network provides the best results in fusing two signals however, it is also more computationally expensive because number of parameters in the encoder part is doubled. The network still has the advantage of utilizing pretrained VGG weights without modifications.

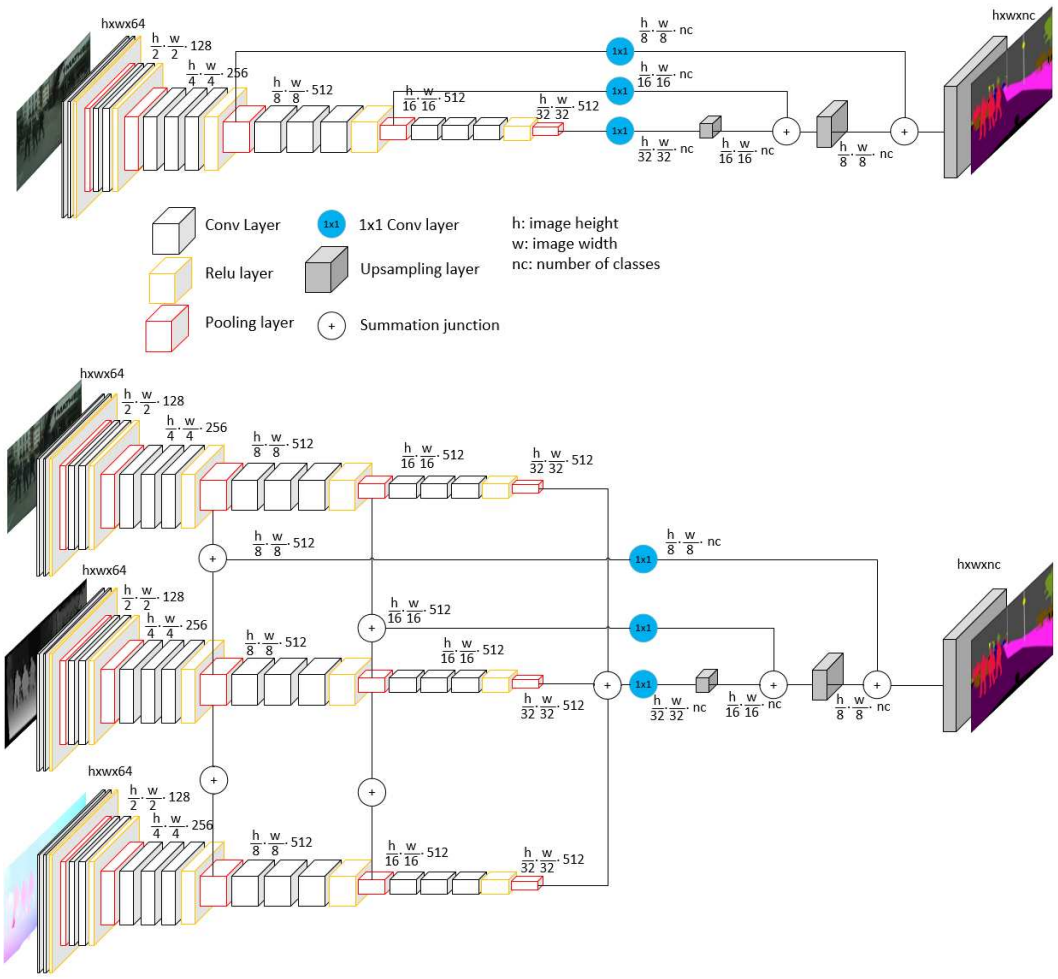

Figure 1: **Top:** Unimodal Architecture. **Bottom:** Multimodal-Mid-Fusion Architecture

3.4. Multimodal Mid-Fusion Architecture

In this architecture, we introduce multimodal fusion using three modalities. We construct a CNN that extracts features from three input signals then fusion is done on feature-level using the same approach as Mid-Fusion network. The output feature maps are fused using a summation junction, and finally, encoder output is upsampled to reach the original image size as illustrated in Figure 1.

4. Experiments

4.1. Experimental Setup

The VGG encoder network is initialized with the pretrained weights and dropout with probability 0.5 is used. Input resolution for Virtual-KITTI is 375x1242 while Cityscapes images are downsized to half-size to be 512x1024. The metric used for evaluation is class-wise Intersection over Union (IoU) and mean IoU over all classes.

4.2. Datasets

Two well-known datasets, namely Virtual-KITTI and Cityscapes, are used in the experiments. Virtual-KITTI provides perfect annotations while Cityscapes provides more realistic data. Virtual-KITTI consists of 21,260 frames that are generated using the Unity game engine. They include 5 different environments under different weather conditions. Virtual-KITTI provides semantic annotation for 14 classes which we train our network to predict. The dataset is split into 80% for training and 20% for testing. On the other hand, Cityscapes dataset consists of 5,000 real images that have fine semantic segmentation annotations. The split provided by this dataset is used and results are reported on the validation set which is not used in training. For Virtual-KITTI dataset, the ground truth data provides perfect depth and optical flow estimations that are used to train our networks. Additionally, experimental results using CNN-based depth and flow estimators, namely monoDepth

Table 1: Quantitative results on Virtual-KITTI dataset using IoU metric.

Type	Mean	Truck	Car	Van	Road	Sky	Building	Guardrail	TrafficSign	TrafficLight	Pole
RGB	66.47	33.66	85.69	29.04	95.91	93.91	68.15	81.82	66.01	65.07	40.91
D (GT)	55	67.68	58.03	56.3	73.81	94.38	43.95	14.61	53.97	56.51	**42.67**
RGBD (GT)	66.76	65.34	91.74	56.93	95.46	94.41	54.91	73.42	60.21	46.09	30.46
RGB+D (GT)	**72.13**	62.84	**93.32**	38.42	**96.33**	94.2	**79.04**	**90.85**	**72.22**	**67.83**	34.4
D (monoDepth)	46.1	36.05	75.46	33.2	77.3	87.3	32.3	6.8	42.14	45.9	15.9
RGB+D (monoDepth)	68.92	40.57	86.1	50.3	95.95	93.82	70.43	86.3	68.66	67.58	35.94
F (GT)	42	36.2	55.2	20.7	62.6	93.9	34	15.23	51.5	33.2	29.3
RGBF (GT)	65.328	70.74	80.2	48.34	93.6	93.3	62.05	67.86	55.14	55.48	31.9
RGB+F (GT)	70.52	**71.79**	91.4	56.8	96.19	93.5	66.53	82.6	64.69	64.65	26.6
F (Flownet)	28.6	24.6	47.8	14.3	57.9	68	4.9	0.8	31.8	18.5	6.6
RGB+F (Flownet)	68.84	60.05	90.87	40.54	96.05	91.73	68.52	82.43	65.2	63.54	26.54
RGB+D+F (GT)	71.88	71.688	92.08	**61.44**	95.85	**94.83**	71.86	83.42	64.69	60.67	31.08

Table 2: Quantitative results on Virtual-Kitti different flow representations using IoU metric.

Type	Mean	Truck	Car	Van	Road	Sky	Building	Guardrail	TrafficSign	TrafficLight	Pole
RGBF (GT-Color Wheel)	59.88	41.7	84.44	40.74	93.76	93.6	49.43	52.18	62.21	49.61	21.52
RGBF (GT-Mag & Dir)	58.85	45.12	82.3	30.04	90.25	94.1	56.48	51.48	58.74	49.7	26.01
RGBF (GT-Mag only)	65.32	70.73	80.16	48.33	93.59	93.3	62.04	67.86	55.13	55.48	31.92
RGB+F (GT-3 layers Mag)	67.88	35.7	91.02	24.78	96.47	**94.06**	**74.4**	**84.5**	**69.48**	**68.95**	**34.28**
RGB+F (GT-Color Wheel)	**70.52**	**71.79**	**91.4**	**56.8**	**96.19**	93.5	66.53	82.6	64.69	64.65	26.6

[6] and FlowNet2[9] are used to assess the impact of estimated depth and flow maps instead of perfect ground truth. For Cityscapes, we provide results of using classical depth and flow estimators, namely SGM [8] and Farneback [4], and compare results with CNN estimators monoDepth, and FlowNet2 to evaluate the impact of noisy input on segmentation. Classical approaches provide noisy disparity and flow maps while CNN-based estimators provide smoother output and hence less noisy, however less accurate due to tendency of CNNs for over-smoothing. We provide experimental comparison for using both approaches.

4.3. Results

Several tables are provided in this section for quantitative comparison of semantic segmentation results. Table 1 illustrates that depth and flow augmentation improve results in Virtual KITTI dataset. For depth, we adopted two methods for usage of depth maps. One is the ground truth, and the other is monoDepth [6]. With ground truth we obtained 5.7% improvement in IoU which is the best-case scenario. Higher improvement was obtained for specific classes, for example, 32%, 28%, 9% and 8% for Truck, Van, Building and Traffic Lights classes. With non-perfect depth maps we obtained improvement over RGB-only by 3% for mean IoU. For optical flow, we used ground truth as perfect estimator, and flowNet [9] as a realistic flow estimate. Fusion with ground truth improved mean IoU by 4% while moving classes were improved significantly. For example, improvements of 38%, 28% and 6% were obtained for Truck, Van and Car classes. Fusion with FlowNet improved mean IoU by 2.37%. We refer to early fusion with Depth by RGBD, mid-fusion by RGB+D while GT denotes Ground Truth. The same strategy is applied to fusion with optical flow.

Table 3 shows results on more challenging dataset Cityscapes. Cityscapes is more complex with very high level of detail compared to Virtual-KITTI. Moreover, it only has a total of 5k images for training and testing compared to 21k images in Virtual-KITTI. We make use of two depth maps, namely SGM which is obtained using stereo setup and provides noise disparity map and the other one is obtained using monoDepth estimator. Using SGM we obtained 1% increase in mean IoU while larger improvement for objects like Buses and Trains with 6.3% and 9% improvement. Using monoDepth, results were very close to SGM which shows the networks capability to fuse the information even with noise input. For optical flow, we used classical Farneback algorithm which improved moving objects like Motorcycle for 14%. Fusion with flowNet maps surprisingly provided slightly lower results than Farneback, however there is improvement for large

Table 3: Quantitative results on Cityscapes depth and Flow fusion using IoU metric.

Type	Mean	Bicycle	Person	Rider	Motorcycle	Bus	Car	Train	Truck	Building	Road	Sky	TrafficSign
RGB	62.47	63.52	67.93	40.49	29.96	62.13	89.16	44.19	48.54	**87.86**	96.22	89.79	59.88
D (SGM)	47.8	39.84	54.99	29.04	11.29	48.1	82.36	27.17	35.3	78.42	95.15	81.18	27.96
RGBD (SGM)	55.5	56.68	60.27	34.64	21.18	58	86.94	36.54	37.64	84.7	94.84	84.64	45.48
RGB+D(SGM)	**63.13**	65.32	67.79	39.14	37.27	**69.71**	**90.06**	53	45.5	87.44	**96.6**	91.06	59.44
D(monoDepth)	40.89	36.63	44.6	18.5	7.3	37.5	77.78	21.6	31.1	77.01	92.83	89.33	24.67
RGB+D (monoDepth)	63.03	65.85	67.44	41.33	46.24	66.5	89.7	55.1	49.7	87.25	96.01	90.3	59.8
F (Farnebaeck)	34.7	34.48	37.9	12.7	7.39	31.4	74.3	11.35	19.42	72.77	91.2	79.6	11.4
RGBF (Farnebaeck)	47.8	52.6	55.8	31.1	22.4	39.34	82.75	22.8	20.7	80.43	92.24	81.87	44.08
RGB+F (Farnebaeck)	62.56	**63.65**	66.3	39.65	**47.22**	66.24	89.63	51.02	36.1	87.13	96.4	90.64	**60.68**
F (Flownet)	36.8	32.9	50.9	26.8	5.12	25.99	75.29	15.1	25.46	65.16	90.75	50.16	29.14
RGBF (Flownet)	52.3	54.9	58.9	34.8	26.1	53.7	83.6	40.7	28.1	79.4	94	79.4	45.5
RGB+F (Flownet)	62.43	64.2	66.32	40.9	40.76	66.05	90.03	41.3	**54.7**	87.3	95.8	**91.07**	58.21
RGB+D+F	62.58	65.46	**68.18**	**43.09**	37.5	64.4	88.34	**57.13**	41.45	86.86	96.13	87.96	59.23

Table 4: Quantitative results on Cityscapes using different flow representations using IoU metric.

Type	Mean	Bicycle	Person	Rider	Motorcycle	Bus	Car	Train	Truck	Building	Road	Sky	TrafficSign
RGBF (Mag only)	47.8	52.63	55.82	31.08	22.38	39.34	82.75	22.8	20.7	80.43	92.24	81.87	44.08
RGBF (Mag & Dir)	54.6	57.28	58.63	33.56	18.49	56.44	84.6	41.15	31.8	84.41	95.5	87.86	44.26
RGBF (Color Wheel)	57.2	61.47	62.18	35.13	22.68	54.87	87.45	36.69	40.2	86.28	95.94	90.07	51.64
RGB+F (3 layers Mag)	62.1	**65.15**	65.44	32.59	33.19	63.07	89.48	43.6	57.2	**87.88**	96.17	**91.48**	55.76
RGB+F (Color Wheel)	**62.56**	63.65	**66.3**	**39.65**	**47.22**	**66.24**	**89.63**	**51.02**	36.11	87.13	**96.4**	90.64	**60.68**

objects like Truck class in which we obtained 6% increase in IoU. We argue that this is due to over smoothing of CNN algorithms to the optical flow map while Farneback provide noisy map however with higher level of detail.

Table 2 and Table 4 show comparative study for different flow representations. Different representations include using optical flow magnitude only, magnitude and direction concatenated together , and finally color wheel representation. Color wheel is showed to provide good representation for both magnitude and direction in a color map which is also consistent with the input expected from VGG pretrained weights. In both Table 1 and Table 3 we report results of fusing all the information together. In this experiment we constructed a CNN network with three encoders, and we fuse the information on the feature level. RGB+D+F provide significantly higher performance than standard RGB. However, results do not show the benefit of fusing the three modalities together. This might be due to the network learning one modality implicitly without explicitly using it. Another reason is that standard network architecture we use may not be able to fuse different modalities together. In future work, we plan to maximize the benefit from these crucial signals using more complex network architectures.

RGB+D and RGB+F obtain the best result where depth and motion significantly improved segmentation. This can be confirmed visually in Figure 2 through the white van with no texture in the background. RGB-only was not able to correctly classify it, and considered it as part of the building behind it. However, when Depth and motion are added, the van is segmented better. The same comment applies for fusion with FlowNet, however, we still need more robust algorithms to get more benefit from this information. Qualitative result on Cityscapes are illustrated on Figure 2 where it is shown that RGB+D significantly improved the bus classification compared to RGB-only. It is shown that the network was not able to classify the bus using color information only. Using depth signal alone provides better result than RGB and consequently contributes to improving the bus classification which is seen in RGB+D result. On the other hand, fusion with monoDepth improved result of RGB however, with less accuracy compared to SGM. It can be seen that the SGM depth map is more accurate than monoDepth map despite the noisy pixels as the bus is better perceived visually in SGM map than monoDepth. Usually, CNN algorithms tend to oversmooth the output, which in this case provided less contrast between the bus and background.

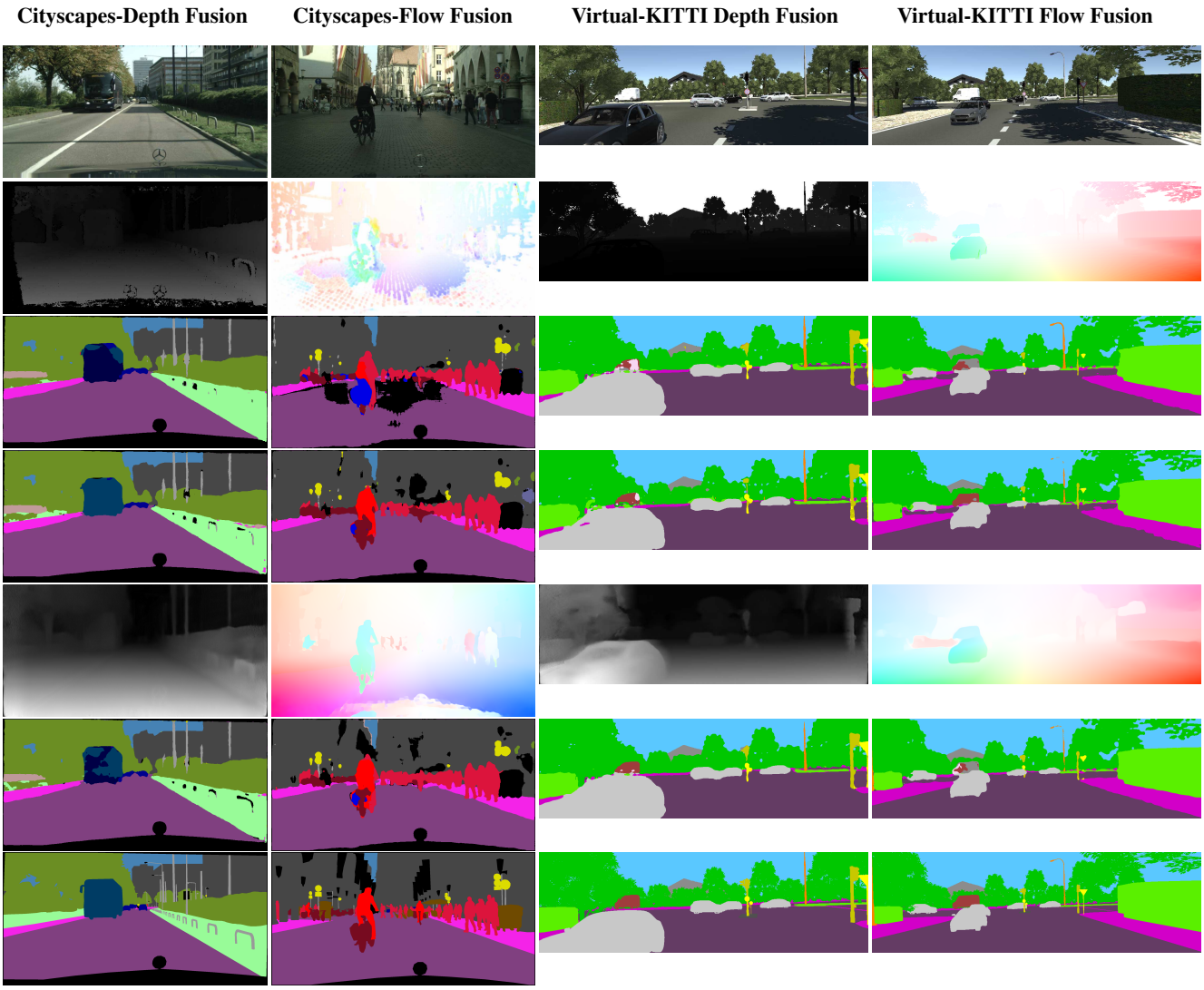

Figure 2: Qualitative results on Cityscapes fusion. **First Row:** Input RGB images. **Second Row:** Input Depth and flow maps. **Third Row:** RGB-only output. **Fourth Row:** Fusion output. **Fifth Row:** CNN-based Depth and flow maps. **Sixth Row:** Fusion with CNN-based maps. **Seventh Row:** Ground Truth

5. Conclusions and Future Work

This work provides a CNN-based architecture that uses multimodal information fusion for sematic segmentation. The network focuses on autonomous driving applications where prior information is exploited to enhance the segmentation task. The utilization of depth information implicitly captures the strong geometric similarity found in structures in most autonomous driving scenes whereas the use of optical flow facilitate handling moving objects in the highly dynamic environment. An ablation study is provided to evaluate the impact of fusing color, depth and motion information at different stages of the network including early-fusion, mid-fusion and multimodal-mid-fusion stages. It was found out that fusing depth and optical flow separately provided the best results. Multimodal fusion provided improved performance over baseline RGB-only segmentation network. Future work may include proposing deeper more complex network architecture that accommodates all three information cues for improved performance. Further work can also investigate utilizing structural constraints [16] to further enhance segmentation results.

Acknowledgements

Authors would like to thank their employer for supporting fundamental research.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 1

[2] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Scene parsing with multiscale feature learning, purity trees, and optimal covers. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1857–1864. Omnipress, 2012. 1

[3] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1915–1929, 2013. 1

[4] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer, 2003. 4

[5] R. Garg, V. K. B. G, and I. D. Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. *CoRR*, abs/1603.04992, 2016. 2

[6] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. *CoRR*, abs/1609.03677, 2016. 2, 4

[7] D. Grangier, L. Bottou, and R. Collobert. Deep convolutional networks for scene parsing. In *ICML 2009 Deep Learning Workshop*, volume 3. Citeseer, 2009. 1

[8] H. Hirschmüller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *null*, pages 807–814. IEEE, 2005. 2, 4

[9] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *IEEE conference on computer vision and pattern recognition (CVPR)*, volume 2, page 6, 2017. 2, 4

[10] S. D. Jain, B. Xiong, and K. Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2117–2126. IEEE, 2017. 2

[11] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. *CoRR*, abs/1606.00373, 2016. 2

[12] 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*, pages 3431–3440, 2015. 1, 2

[13] S. Meister, J. Hur, and S. Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 2

[14] H. Rashed., S. Yogamani., A. El-Sallab., P. Křek, and M. El-Helw. Optical flow augmented semantic segmentation networks for automated driving. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, pages 165–172. INSTICC, SciTePress, 2019. 2

[15] M. Siam, S. Eikerdawy, M. Gamal, M. Abdel-Razek, M. Jagersand, and H. Zhang. Real-time segmentation with appearance, motion and geometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5793–5800. IEEE, 2018. 2

[16] M. Siam and M. Elhelw. Enhanced target tracking in uav imagery with pn learning and structural constraints. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 586–593, 2013. 6

[17] M. Siam, H. Mahgoub, M. Zahran, S. Yogamani, M. Jagersand, and A. El-Sallab. Modnet: Motion and appearance based moving object detection network for autonomous driving. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2859–2864. IEEE, 2018. 2

[18] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[19] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. Demon: Depth and motion network for learning monocular stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5038–5047, 2017. 2

[20] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis. Ev-flownet: self-supervised optical flow estimation for event-based cameras. *arXiv preprint arXiv:1802.06898*, 2018. 2