STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LXV**, Number 1, 2020
DOI: 10.24193/subbi.2020.1.03

INTUITIVE ESTIMATION OF SPEED USING MOTION AND MONOCULAR DEPTH INFORMATION

RÓBERT ADRIAN RILL[1,2]

Abstract. Advances in deep learning make monocular vision approaches attractive for the autonomous driving domain. This work investigates a method for estimating the speed of the ego-vehicle using state-of-the-art deep neural network based optical flow and single-view depth prediction models. Adopting a straightforward intuitive approach and approximating a single scale factor, several application schemes of the deep networks are evaluated and meaningful conclusions are formulated, such as: combining depth information with optical flow improves speed estimation accuracy as opposed to using optical flow alone; the quality of the deep neural network results influences speed estimation performance; using the depth and optical flow data from smaller crops of wide images degrades performance. With these observations in mind, a RMSE of less than 1 m/s for ego-speed estimation was achieved on the KITTI benchmark using monocular images as input. Limitations and possible future directions are discussed as well.

1. Introduction

The significant progress in recent years makes deep learning solutions attractive in automotive industry applications [24]. In intelligent transportation systems, self-driving cars, advanced driver-assistance systems (ADAS) vehicle speed is one of the most important parameters for vehicle control and safety considerations. It can be measured using wheel sensor, Inertial Navigation System (INS) or Global Positioning System (GPS). Although these methods achieve high accuracy, they have limitations. Speed sensors involve a trade-off

Received by the editors: 10 January 2020.

2010 *Mathematics Subject Classification.* 68T45, 97R40.

1998 *CR Categories and Descriptors.* I.5.4 [**Computing Methodologies**]: Applications – *Computer Vision*; I.4.8 [**Computing Methodologies**]: Scene Analysis – *Motion, Depth cues*.

Key words and phrases. monocular vision, speed estimation, deep learning, optical flow, single-view depth.

This paper is based on the preprint: R.A. Rill. Speed estimation evaluation on the KITTI benchmark based on motion and monocular depth information. arXiv e-prints arXiv:1907.06989 [cs.CV] (2019).

between accuracy and cost. INS suffers from integration drift, i.e. small errors accumulate over time. GPS is prone to signal interference and loss in blocked areas, and may provide unreliable data when accelerating or decelerating. Some researchers also propose model-based approaches (see, e.g., [22]), which, however, may be significantly affected by incorrect parameter estimates.

Other methods for automotive sensing technologies, including speed estimation, are Radio Detection and Ranging (RADAR) or Light Detection and Ranging (LiDAR) systems that use radio frequency or laser signals, respectively [1]. Although both systems are popular in the industry, they have deficiencies. LiDAR systems can identify fine details of the 3D environment, but are greatly affected by unfavorable weather conditions and more importantly they are costly to produce and maintain. In contrast, RADAR systems are more robust, lightweight and cheap, but have lower accuracy and resolution.

To address the limitations of conventional speed estimation methods, computer vision based approaches have become attractive alternatives in recent years. Powerful hardware is available for autonomous vehicles to run deep learning algorithms in real-time [16]. In this work a simple monocular vision-based speed estimation approach is presented that exploits the recent advances in deep learning-based optical flow and monocular depth prediction methods. Optical flow is the pattern of apparent motion of objects in a visual scene caused by the relative motion between an observer and the scene. Monocular depth estimation aims to obtain a representation of the spatial structure of a scene by determining the distance of objects from a single image. The two problems are fundamental in computer vision and represent highly correlated tasks (see, e.g., [25]). The proposed method relies on the intuition that the magnitude of optical flow is positively correlated with the moving speed of the observer and that objects closer to the camera appear to move faster than the more distant ones. Different schemes of the proposed approach are investigated on a representative subset of the KITTI dataset [6, 5], and a RMSE of less than 1 m/s is achieved.

The rest of the paper is organized as follows. Section 2 presents related works providing a background and motivation for this study. Section 3 introduces the KITTI dataset and the deep learning methods used, and details the speed estimation pipeline. The quantitative and qualitative results are presented in Section 4. Section 5 discusses the results and limitations, highlights future directions and finally Section 6 concludes the paper.

2. Related work

Vision based approaches represent a promising direction for vehicle speed estimation that may replace or complement traditional methods. Most of the works are concerned with estimating the speed of the vehicles in traffic using a camera mounted for traffic surveillance. These methods involve different image processing techniques: background extraction [3, 20, 23, 14], image rectification [3, 20, 23, 12], detecting and tracking reference points [3, 20, 14] or centroids [23] over successive frames, converting the displacement vectors from the image to the real-world coordinate system. The state-of-the-art results of deep learning in vision tasks makes object detection and tracking [12], locating license plates on vehicles [14], 3D convolutional networks [4] other promising directions in the task of speed estimation. Disadvantages of these approaches and of the traffic enforcement solutions already in use, such as speed or point-to-point cameras, include the need for calibration processes, meticulous positioning of the devices at predefined locations, investment in infrastructure and maintenance.

As opposed to traffic surveillance purposes, few studies address the problem from an ADAS perspective, namely estimating speed using monocular images from a camera mounted on the car. Some works estimate the relative speed of other participants in traffic (see, e.g., [18] or [11]). The present study is concerned with estimating the absolute speed of the car the camera is mounted on, also called the ego/forward/longitudinal speed.

In [17] the authors used sparse optical flow to track feature points on images from a downward-looking camera mounted on the rear axle of the car and achieved a mean error relative to GPS measurement of 0.121 m/s. However, the method works only in restricted conditions and was evaluated on self-collected data at low speed values. Han [8] used projective geometry concepts to estimate relative and absolute speed in different case studies. Using black box footages, a maximum of 3% difference was reported for higher ego-speed values when compared to GPS measurements. The major limitation of this study is the assumption of known distances between stationary objects such as lane markings. Banerjee et al. [2] used a rather complicated neural network architecture trained on self-collected data and reported an RMSE of 10 mph on the KITTI benchmark [5].

In this work a simple intuitive approach for ego-speed estimation is investigated using state-of-the-art deep neural network-based optical flow and monocular depth prediction methods. The proposed method achieves a RMSE of less than 1 m/s on recordings from the KITTI dataset.

3. Methods

3.1. **Dataset and deep neural networks.** The KITTI Vision Benchmark Suite[1] [6, 5] is a popular real-world dataset consisting of 6 hours of traffic

[1] http://www.cvlibs.net/datasets/kitti

scenario recordings captured while driving in and around a mid-size city. The traffic situations range from highways over rural areas to inner-city scenes with many static and dynamic objects. To evaluate speed estimation 15 recordings of rectified images were manually selected from the left input color camera of the KITTI dataset. The list of drive numbers are: 1, 2, 5, 9, 14, 19, 27, 48, 56, 59, 84, 91, 95, 96, 104, all from 2011.09.26. These are representative videos where the car is moving almost always.

In the experiments presented in this paper two optical flow estimation methods are compared: FlowNet2[2] [9] and PWC-Net[3] [19]. FlowNet2 is a consolidation of the original FlowNet idea that proposed to use convolutional neural networks to learn optical flow and poses the problem as an end-to-end supervised learning task. FlowNet2, compared to its initial versions, shows large improvements in quality and speed. While it achieves impressive performance by stacking basic models into a large capacity model, the much smaller and easier to train PWC-Net obtains similar or better results by embedding classical and well-established principles into the network itself.

Similarly, two single-view depth estimation methods are examined: MonoDepth[4] [7] and MegaDepth[5] [13]. MonoDepth innovates beyond existing learning based single image depth estimation methods by replacing the use of large quantities and difficult to obtain quality training data with easier to obtain binocular stereo footage. It poses the task as an image reconstruction problem. On the other hand, MegaDepth refers to a large depth dataset generated via modern structure-from-motion and multi-view stereo methods from Internet photo collections. The models trained on MegaDepth exhibit high accuracy and strong generalization ability to novel scenes. One important difference is that while MonoDepth predicts disparity values, MegaDepth models predict ordinal depth defined up to a scale factor.

For more details about the four deep learning algorithms investigated in this work and related methods please see the cited works and the references therein. Before applying the methods for speed estimation, their performance was evaluated quantitatively on test data from KITTI. Note, however, that in order to obtain dense optical flow and depth information the 1242 × 375 resolution input images in the KITTI dataset need to be resized accordingly. The output of the deep neural networks was resized back to the original resolution as summarized in Table 1. To run the neural networks pre-trained weights were used, provided in the corresponding Github repositories. The quantitative results obtained are reported in Section 4.

[2] `https://github.com/lmb-freiburg/flownet2`

[3] `https://github.com/sniklaus/pytorch-pwc`

[4] `https://github.com/mrharicot/monodepth`

[5] `https://github.com/lixx2938/MegaDepth`

Table 1. **Deep neural network technicalities.** *For details see the references and/or the Github repositories.

Method	Input resolution	Input resize	Output resize	Model used*
FlowNet2	divisible by 64	pad with zeros	trim zeros	FlowNet2
PWC-Net	1024×436	bilinear interpolation	anti-aliasing	default network
MonoDepth	512×256	anti-aliasing	anti-aliasing	city2kitti
MegaDepth	512×384	anti-aliasing	anti-aliasing	best generalization

3.2. **Speed estimation pipeline.** In order to estimate speed from a moving camera, two observations should be made: (i) optical flow is expected to highly correlate with moving speed, and (ii) the apparent motion of objects closer to the camera is faster than those of more distant ones. Therefore, to obtain the speed of a given object on the image – that is the moving speed of the camera – it is reasonable to multiply optical flow magnitude by depth.

After extensively considering variations of the above intuitive idea with respect to deep learning based optical flow and single-view depth estimation methods, the following base multistep process is proposed, and some modifications will be inspected as well. In a first step the results of one optical flow and one depth estimation algorithm are retrieved for a given image frame. The magnitude of optical flow vectors will be denoted by OF, and the disparity by DISP in the following. In the second step, the OF and DISP values are considered at valid pixels from a predefined crop of the original image, and the mean OF is divided by the mean DISP to get a scaled speed estimate. The valid pixels are obtained by imposing thresholds: OF > 0.1 and DISP > 0.02, which were selected after extensive experiments. Note, however, that selecting other close values gave similar results and does not affect the conclusions of the paper. The next step is the concatenation of the scaled speed estimates over the temporal dimension, i.e. over frames of a video. The aggregated vectors are temporally smoothed using a 1D convolution of size 25 with equal weights. Finally, the resulting smoothed lists are taken for multiple recordings and a scaling factor is approximated that minimizes the ratio between the ground truth and predicted speed. This scaling factor is used to convert speed from the image domain to real-world units.

To summarize, the steps of the base speed estimation algorithm (denoted by A) are as follows:

(1) Run optical flow and depth estimation methods on a given image.
(2) Compute the scaled speed for the given frame: consider the OF and DISP values at valid pixels from a predefined image crop, and compute the quotient between their means.

(3) Repeat the previous steps for all the frames from a video and apply temporal smoothing.

(4) Repeat previous step for multiple videos and determine scaling factor.

In Section 4 several modifications of the above base pipeline are experimented with. These are summarized below.

(A_1) Neglect depth information completely, and use optical flow alone.

(A_2) Replace OF by the magnitude of horizontal optical flow only (which is expected to highly correlate with the moving speed especially towards the edges of the image).

(A_3) Apply temporal smoothing at the pixel-level separately for OF and DISP, before computing their means.

(A_4) Run the neural networks directly on the image crop, as opposed to using the full frame as input first and extracting OF and DISP for the crop after.

(A_5) Investigate different image crops in the base algorithm, including the full wide frame.

In experiment A_5 defined above the modification of the base speed estimation pipeline is evaluated on three different image crops and the full frame. They are defined in Table 2. Reasons for using crops only include the possible unavailability of wide images, or memory and run-time considerations.

Table 2. **Definition of image crops used in the experiments.** (x, y) defines the upper left corner and w, h the width and height of the bounding boxes in pixels.

Image crop	bounding box			
	x	y	w	h
crop_1	720	180	200	120
crop_2	700	100	400	240
crop_3	640	20	580	340

4. Experiments and results

Figure 1 shows visualizations of the deep neural network methods on one sample frame from the KITTI dataset. Both optical flow estimation methods produce smooth flow fields with sharp motion boundaries. PWC-Net seems to be more robust against shadow effects. The depth prediction methods also show good visual quality, with MonoDepth being able to capture object boundaries and thin structures more reliably.

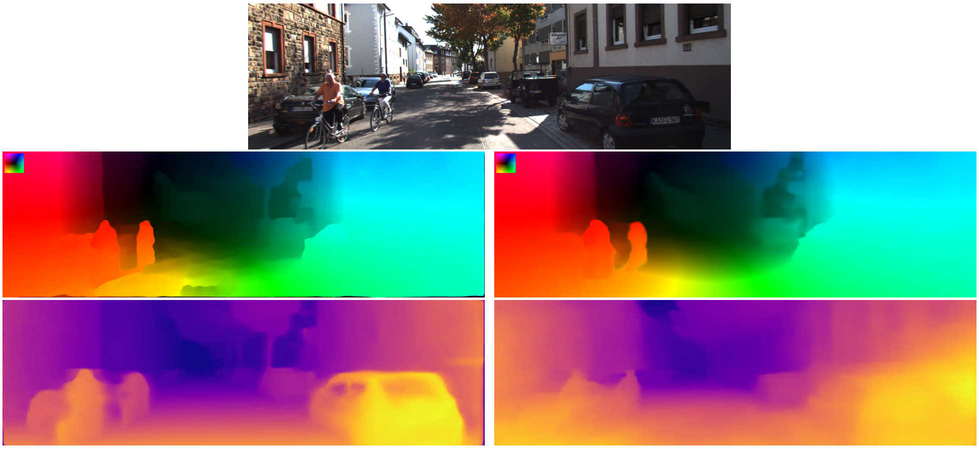


FIGURE 1. **Sample visualisation of network results.** From left to right and top to bottom: frame 71 of *drive_0095*, FlowNet2, PWC-Net, MonoDepth, MegaDepth. The colored square represents the color coding of optical flow.

The methods were evaluated quantitatively as well. The optical flow estimation algorithms are evaluated on the 200 training images from the KITTI 2015 benchmark [15]. The results displayed in Table 3 show that PWC-Net outperforms FlowNet2. The depth prediction algorithms are evaluated on the given 1000 manually selected images from the full validation split of the derived depth prediction and completion KITTI dataset [21]. According to Table 4, MonoDepth achieves better performance in several metrics compared to MegaDepth. In all four cases the results are in correspondence with those reported in the references presenting the methods.

TABLE 3. **Evaluation of optical flow estimation methods.** The KITTI 2015 benchmark [15] was used. AEPE: average endpoint error; Fl-all: Ratio of pixels where flow estimate is wrong by both ≥ 3 pixels and $\geq 5\%$.

Method	AEPE	Fl-all
FlowNet2	11.686	32.183%
PWC-Net	2.705	9.187%

Table 5 shows the speed estimation results for the experimental algorithms $A_1 - A_4$. Inspecting the values in detail allows to draw the following conclusions. According to A_1, using depth information as well improves speed

TABLE 4. **Evaluation of depth estimation methods.** The manual selection of the validation split of the derived depth prediction and completion KITTI 2017 dataset [21] was used.

Method	RMSE	RMSE(log)	Abs Rel	Sq Rel	log10	Scale-inv.
MonoDepth	4.532	0.150	0.090	0.749	0.040	0.142
MegaDepth	6.719	0.336	0.322	1.994	0.124	0.289

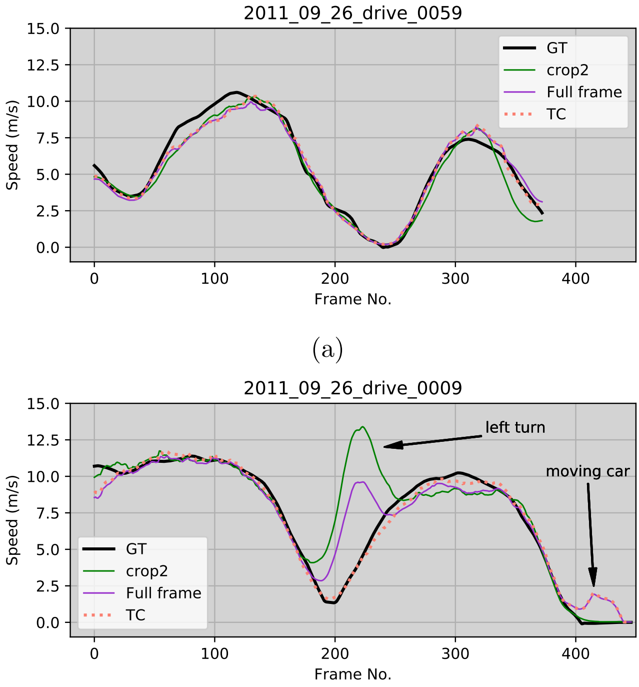
estimation performance, as opposed to considering optical flow alone. A_2 shows that replacing OF with horizontal optical flow results in slightly higher RMSE values. As demonstrated by A_3, applying temporal smoothing at the pixel level increases performance in some cases but only marginally. Finally, according to experiment A_4, running the neural networks on smaller image crops degrades performance considerably.

TABLE 5. **Results of speed estimation experiments.** RMSE values are shown using crop$_2$ defined in Table 2. A_i, $i \in \{1, 2, 3, 4\}$ refers to the algorithms from Section 3.

Method	Base algorithm A	Horizontal optical flow (A_2)	Pixel-level smoothing (A_3)	Methods only on crop (A_4)
FlowNet2 (A_1)	2.921	3.005	2.919	3.093
PWC-Net (A_1)	2.472	2.621	2.472	3.170
FlowNet2 & MonoDepth	2.305	2.448	2.399	2.618
PWC-Net & MegaDepth	1.915	2.059	1.908	3.180
FlowNet2 & MegaDepth	2.485	2.526	2.475	2.901
PWC-Net & MonoDepth	**1.467**	1.707	1.865	2.967

Furthermore, from Table 5 it can be seen that the best results are obtained when PWC-Net is combined with MonoDepth. Figure 2 shows speed estimation results for two KITTI recordings using the base pipeline. This simple method captures speed changes with low error in straight travel scenarios (Figure 2a), but having difficulty in cases when the car is taking a turn (around frame 200 on Figure 2b speed decreases as the car is turning left, yet optical flow increases on the right side – and in crop$_2$ too – of the wide KITTI images).

Modification A_5 of the base algorithm is evaluated on three image crops and the full frame. Table 6 illustrates that speed estimation accuracy improves in general as the size of the image increases. The best results are obtained again by the PWC-Net – MonoDepth combination. When the full frame is used errors moderately decrease at car turning events (around frame 200 on



(a)

(b)

Figure 2. **Speed estimation on sample KITTI videos.** Results are shown for the base pipeline with PWC-Net and MonoDepth; crop$_2$: defined in Table 2; *Full frame*: full wide image; GT: ground truth speed; TC: compensating for car turning events. For details see text.

Figure 2b), but overestimations are still present in dynamic scenes due to the motion of other cars for instance (after frame 400 on Figure 2b).

Table 6. **Results using different image crops.** RMSE values are shown for A_5. Image crops are defined in Table 2. TC: compensating for car turning events (for details see text).

Method	crop$_1$	crop$_2$	crop$_3$	Full frame	TC
FlowNet2 & MonoDepth	2.170	2.305	2.558	2.370	2.138
PWC-Net & MegaDepth	2.363	1.915	2.015	1.786	1.445
FlowNet2 & MegaDepth	2.671	2.485	2.583	2.544	2.125
PWC-Net & MonoDepth	1.735	1.467	1.505	**1.178**	**0.977**

In order to decrease speed overestimations at car turning events an additional modification of the base pipeline was experimented with. In such cases the average horizontal optical flow from the left part of the image has the same direction as the average from the right side. Whenever this condition is true, instead of the mean optical flow magnitude from the full wide frame, the absolute value of the difference between the means of horizontal optical flow of the left and right sides is computed, and divided by the mean disparity corresponding to the whole frame. Applying this heuristic compensation for turning events (TC) decreases the RMSE to under 1 m/s, as shown in Table 6. The performance improvement is illustrated by Figure 2b as well.

5. Discussion

Two optical flow estimation (FlowNet2 [9] and PWC-Net [19]) and two depth estimation (MonoDepth [7] and MegaDepth [13]) algorithms were investigated. Evaluating them on ground truth data from the KITTI dataset showed that PWC-Net and MonoDepth achieved better performance in several error metrics. The reason for this is presumably some combination of the following: the MonoDepth model used was fine-tuned on data from KITTI, MegaDepth does not predict metric depth but ordinal depth, it seems that during training of the PWC-Net model KITTI data was used as well. Nonetheless, the conclusion is that better performance optical flow and depth estimation methods result in better speed estimates. Besides, continuous efforts are made to improve these two fundamental computer vision algorithms, including their joint training in an unsupervised manner (see, e.g., [25]). Accordingly, fine-tuning to arbitrary images becomes accessible without the need for difficult to obtain ground truth labels.

After evaluating several modifications of the intuitive approach presented in this paper, meaningful conclusions were formulated: combining optical flow with depth information improves accuracy, using only the horizontal component of optical flow is not beneficial, applying temporal smoothing helps since it reduces the noise present in optical flow and depth estimation methods, using the full wide image frames as input to the deep neural networks and these results for speed estimation provides better approximations as opposed to using smaller image crops. It should be noted that other experiments were carried out as well, the results of which are not presented in the current paper.

There are two major limitations of the proposed method. Firstly, speed is erroneously estimated when the car is turning. However, in such cases estimation errors can be corrected by taking into account that horizontal optical flow on the left and right side of the wide images has the same direction (see the last column of Table 6 and Figure 2b). Secondly, the proposed method is most reliable when the background is static. For example in heavy traffic scenarios when the surrounding cars are moving as well, the correlation of optical flow with ego-speed might be small and speed can be over- (see Figure 2b) or underestimated. In such scenarios combining monocular depth estimation with semantic segmentation [10] represents one promising direction; and the estimation of relative speed can help, which is another problem where recent advances are being made (see, e.g., [18] or [11]). One might also argue that the deep neural network methods providing the best speed estimates were fine-tuned on ground truth data from KITTI. But, as explained above, efforts are being made to train such methods on unlabelled data [25].

To improve the presented method, future works can treat the task as a regression problem and adopt for example a lightweight multilayer perceptron using as input the aggregated optical flow and depth results from different smaller regions of the original image. Another possibility is the exploitation of the more sophisticated convolutional neural network, which, however would possibly require a larger amount of training data [4].

6. Conclusion

In this work a simple algorithm was investigated for ego-speed estimation from images of a camera mounted on the moving car, using state-of-the-art deep neural network based optical flow estimation and monocular depth prediction. The method relies on the intuition that optical flow magnitude is highly correlated with the moving speed of the observer and that the closer objects are to the observer the faster they appear to be moving. Extensive evaluations of the intuitive algorithm lead to a RMSE of less than 1 m/s on a representative subset of the widely exploited KITTI dataset. As a closing remark, it is noteworthy that due to the recent and ongoing advancements in deep learning, monocular vision-based approaches are a promising direction for ego-speed estimation, and autonomous driving in general.

Acknowledgements

The project has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013 and EFOP-3.6.3-VEKOP-16-2017-00002). The author acknowledges his PhD supervisors, Horia F. Pop, for his assistance, and András Lőrincz, for his helpful ideas that promoted the prosperous progression of this project. The author would also like to thank Kinga B. Faragó for her cooperation in supervising the progress of two students, and Szilvia Szeier and Kevin A. Hartyányi for their efforts related to their scientific student conference, which contributed to this work as well.

References

[1] Abuella, H., Miramirkhani, F., Ekin, S., Uysal, M., and Ahmed, S. ViLDAR - visible light sensing based speed estimation using vehicle's headlamps. *arXiv e-prints* (2018), arXiv:1807.05412.

[2] Banerjee, K., Van Dinh, T., and Levkova, L. Velocity estimation from monocular video for automotive applications using convolutional neural networks. In *IEEE IV Symposium* (2017), pp. 373–378.

[3] Doğan, S., Temiz, M. S., and Külür, S. Real time speed estimation of moving vehicles from side view images from an uncalibrated video camera. In *Sensors* (2010).

[4] Dong, H., Wen, M., and Yang, Z. Vehicle speed estimation based on 3d convnets and non-local blocks. *Future Internet 11*, 6 (2019).

[5] Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. Vision meets robotics: The KITTI dataset. *IJRR* (2013).

[6] Geiger, A., Lenz, P., and Urtasun, R. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *CVPR* (2012).

[7] Godard, C., Aodha, O. M., and Brostow, G. J. Unsupervised monocular depth estimation with left-right consistency. In *CVPR* (2017), pp. 6602–6611.

[8] Han, I. Car speed estimation based on cross-ratio using video data of car-mounted camera (black box). *Forensic Science International 269* (2016), 89–96.

[9] Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., and Brox, T. FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR* (2017), pp. 1647–1655.

[10] Jiang, H., Larsson, G., Maire, M., Shakhnarovich, G., and Learned-Miller, E. Self-supervised relative depth learning for urban scene understanding. In *ECCV* (2018), Springer, pp. 20–37.

[11] Kampelmühler, M., Müller, M., and Feichtenhofer, C. Camera-based vehicle velocity estimation from monocular video. In *CVWW* (2018).

[12] Kumar, A., Khorramshahi, P., Lin, W.-A., Dhar, P., Chen, J.-C., and Chellappa, R. A semi-automatic 2d solution for vehicle speed estimation from monocular videos. In *CVPR Workshops* (2018).

[13] Li, Z., and Snavely, N. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR* (2018), pp. 2041–2050.

[14] Luvizon, D. C., Nassu, B. T., and Minetto, R. A video-based system for vehicle speed measurement in urban roadways. *IEEE Transactions on Intelligent Transportation Systems 18*, 6 (2017), 1393–1404.

[15] Menze, M., and Geiger, A. Object scene flow for autonomous vehicles. In *CVPR* (2015).

[16] NVIDIA. Nvidia drive AGX, 2019. https://www.nvidia.com/en-us/self-driving-cars/drive-platform/hardware/.

[17] Qimin, X., Xu, L., Mingming, W., Bin, L., and Xianghui, S. A methodology of vehicle speed estimation based on optical flow. In *Proceedings of 2014 IEEE International Conference on Service Operations and Logistics, and Informatics* (2014), pp. 33–37.

[18] Salahat, S., Al-Janahi, A., Weruaga, L., and Bentiba, A. Speed estimation from smart phone in-motion camera for the next generation of self-driven intelligent vehicles. In *IEEE 85th VTC* (2017), pp. 1–5.

[19] Sun, D., Yang, X., Liu, M.-Y., and Kautz, J. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *CVPR* (2018), pp. 8934–8943.

[20] Temiz, M. S., Kulur, S., and Doğan, S. Real time speed estimation from monocular video. *ISPRS Archives XXXIX-B3* (2012), 427–432.

[21] Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., and Geiger, A. Sparsity invariant CNNs. In *3DV* (2017).

[22] Xu, Q., Li, X., and Chan, C.-Y. A cost-effective vehicle localization solution using an interacting multiple model-unscented kalman filters (IMM-UKF) algorithm and grey neural network. *Sensors 17*, 6 (2017).

[23] Y. G. Anil Rao, N. Sujith Kumar, H. S. Amaresh, and H. V. Chirag. Real-time speed estimation of vehicles from uncalibrated view-independent traffic cameras. In *TENCON 2015 - IEEE Region 10 Conference* (2015), pp. 1–6.

[24] Yao, B., and Feng, T. Machine learning in automotive industry. *Advances in Mechanical Engineering* (2018).

[25] Zou, Y., Luo, Z., and Huang, J.-B. DF-Net: Unsupervised joint learning of depth and flow using cross-task consistency. In *ECCV* (2018), Springer, pp. 38–55.

[1]Faculty of Informatics, Eötvös Loránd University. H-1117 Budapest, Pázmány P. stny 1/C, Hungary.

[2]Faculty of Mathematics and Computer Science, Babeş-Bolyai University. No. 1 Mihail Kogalniceanu St., RO-400084 Cluj-Napoca, Romania.

Email address: rillrobert@cs.ubbcluj.ro, rillroberto88@yahoo.com
(ORCID iD: http://orcid.org/0000-0002-3004-7294)