Boosting semantic segmentation with multi-task self-supervised learning for autonomous driving applications

Jelena Novosel
Valeo Vision Systems
jnovosel@protonmail.com

Prashanth Viswanath
Valeo Vision Systems
prashanth.viswanath@valeo.com

Bruno Arsenali
Valeo Vision Systems
bruno.arsenali@valeo.com

Abstract

Autonomous cars need to understand the complex and dynamic environment around them. The solutions for tasks such as classification, segmentation, and detection provide information required to understand this environment. The state-of-the-art solutions for these tasks are based on a supervised learning, which requires a large amount of annotated data. It is extremely expensive and labour-intensive to produce such data. Unlabelled images and videos are available in large quantities at a negligible cost, but are rarely used. Self-supervised learning is a new concept that exploits unlabeled data. In this study, we use it to improve the performance of a single supervised task (i.e., semantic segmentation). We explore two self-supervised tasks: colorization and depth prediction. The performance is assessed on two datasets: Cityscapes and KITTI. Overall, we show an improvement of up to 3 % when self-supervised tasks are trained with semantic segmentation. In conclusion, self-supervised learning improves the performance of semantic segmentation at no additional annotation nor inference-related computational costs.

1 Introduction

Autonomous cars and advanced driver-assistance systems (ADASs) have advanced substantially in the last decade. To operate successfully, they need to understand the dynamic environment around them. This implies that they need information about the surrounding objects. A large proportion of this information may be obtained through the visual systems integrated within the ego-vehicle.

In the recent years, computer vision and machine learning have advanced rapidly and deep learning is now regarded as a standard approach for tasks such as classification [23, 50], segmentation [7, 3], and detection [45, 34]. A large number of the state-of-the-art methods are based on supervised learning which requires manual data labeling, a time-consuming and an expensive process. For example, it takes 1.5 h on average to perform annotation and quality control for one image of Cityscapes [11].

Unlabelled images and videos are available in large quantities at a negligible cost. Unfortunately, their potential is rarely exploited. Unsupervised learning is used to find hidden structures within unlabelled data. It works well on problems such as clustering [4] and dimensionality reduction [60]. However, since unsupervised learning is not designed to solve a particular problem, it fails to capture relevant information needed to successfully perform vision tasks (e.g., segmentation).

Self-supervised learning, as an emerging concept, has potential to overcome limitations and exploit the benefits of supervised and unsupervised learning. It is a special type of supervised learning in which the labels are automatically generated from the unlabelled data. More specifically, problems are modelled through self-supervision and objectives are measured similarly to those of supervised learning. As such, in contrast to unsupervised learning, self-supervised learning focuses on an

Machine Learning for Autonomous Driving Workshop at the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

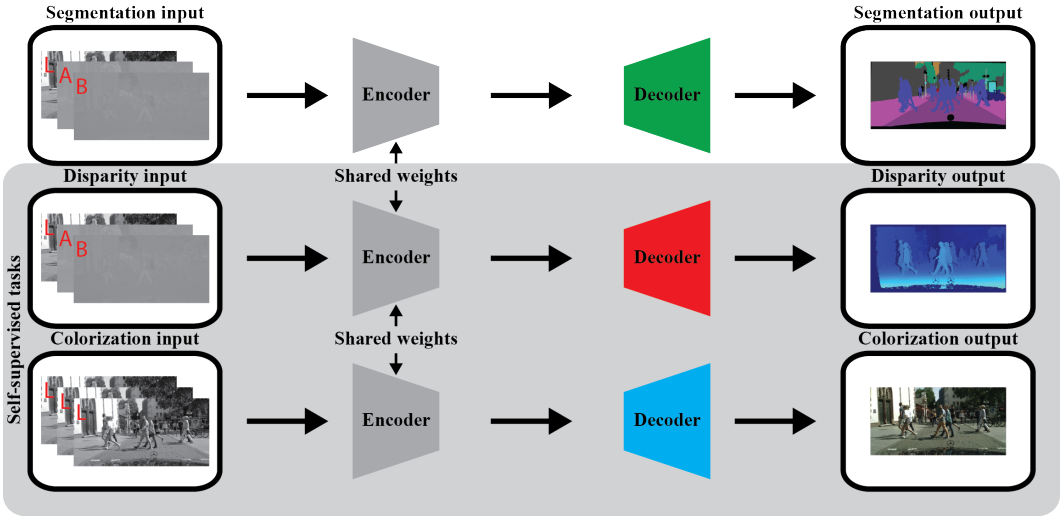

Figure 1: The overview of the proposed system, which consists of a single shared encoder and three task specific decoders (semantic segmentation, depth estimation, and colorization).

optimization of a particular task, which forces the network to better learn semantic information with no additional label-related issues. For additional details on the topic, we refer the reader to [26].

Most prior work on supervised and self-supervised learning solves a single task at a time (e.g., [6] and [21]). This gives good performance, but ignores a lot of useful information. In contrast, when multiple tasks are jointly trained, domain-specific information is used to a larger extent, which results in a better generalization [5]. Indeed, multi-task learning is deployed successfully in many applications [10, 12, 44, 27]. In contrast, to the best of our knowledge, self-supervised multi-task learning is yet to be explored in a greater extent.

In this paper, we exploit self-supervised multi-task learning to improve the performance of a single supervised task (i.e., semantic segmentation). Two self-supervised tasks are explored: colorization and depth prediction. Different multi-task weighting strategies are used and the corresponding results are reported. Performance is evaluated on two datasets: Cityscapes [11] and KITTI [18]. The goal of the paper is not to outperform the state-of-the-art semantic segmentation methods. The goal is to improve the performance with self-supervised learning at no additional annotation costs. The rest of the paper is organized as follows. Section 2 presents the related work. Section 3 describes the proposed model. Section 4 presents the experiments and the results. The conclusion is drawn in section 5.

2 Related works

Details on two self-supervised tasks (i.e., image colorization and depth prediction) are presented in subsection 2.1. An overview of semantic segmentation in the context of deep learning follows in subsection 2.2. Finally, related work on multi-task learning is given in subsection 2.3.

2.1 Self-supervised learning

Self-supervised learning is an emerging concept that exploits automatic generation of ground truth labels. Image colorization [61, 24, 30, 56] and depth estimation [21, 65, 17, 59, 57, 55, 37, 22] are used in the context of self-supervised learning. Other tasks, which are beyond the scope of this paper, are proposed in [13, 40, 41, 19, 43, 31, 39, 51]. A more detailed overview on colorization and depth prediction is given below.

Image colorization Image colorization is a task of adding color to a grayscale image. The most noted approaches [61, 24, 30] solve colorization with a convolutional neural network. Iizuka et al. [24] use colors as classes and the network is trained jointly with classification (cross-entropy) and colorization (L_2) loss. Zhang et al. [61] use only classification loss, but instead of color value prediction, they predict color histograms and use class weights to increase diversity of colors. Concurrently, Larsson et al. [30] propose a similar network that achieves comparable accuracy.

Depth prediction Supervised depth prediction shows promising results [54], but as it requires expensive depth sensors, the focus has shifted towards self/un-supervised methods [21, 65, 17, 59, 57, 55, 37, 22]. Garg et al. [17] use stereo image pairs to introduce self-supervised depth prediction, in which depth is an intermediate step, and self-supervision is defined through the reconstruction loss. Godard et al. [21] further improve the consistency of depth prediction. Zhou et al. [65] propose a fully differentiable approach where depth and ego-motion are predicted jointly from a monocular video. Vijayanarasimhan et al. [55] learn moving objects masks and rigid motion parameters for objects and ego-motion. Many of the subsequent work improve these initial results [59, 57, 37, 22].

2.2 Semantic segmentation

Semantic segmentation provides a label for each pixel in an image. Long et al. [36] propose the first fully convolutional network for semantic segmentation. SegNet [2] introduces shortcut connections. Yu and Kotlun [58] propose dilated convolutions. DeepLab [6] is based on these convolutions and its latest version [7] is among the best performing solutions for semantic segmentation. Other solutions include networks such as PSPNet [64], PANet [33], and DANet [16]. Due to computational requirements, many of these solutions are not suitable for the automotive industry. To enable semantic segmentation in real-time, smaller networks are proposed. ENet [42], MobileNet [48], ICNet [63], and ShuffleNet [62] are a few examples of these networks.

2.3 Multi-task learning

Multi-task learning aims to leverage shared domain information contained in multiple tasks to improve the generalization of all tasks. MultiNet [52] is the first architecture with a shared encoder for classification, detection, and segmentation. In [38], a network with cross-stitch units is proposed to learn an optimal combination of shared and task-specific representations. UberNet [29] is a network that learns a large number of tasks under one architecture.

Different strategies are available to achieve a good balance across multiple tasks. A standard, brute force, optimization schemes for task weighting are the grid search and random search. More advance methods are also proposed. For example, task-dependent uncertainty is proposed to find weights for multiple objectives [27], GradNorm framework exploits gradients to weight multiple tasks [8]. Dynamic weight average exploits the rate of change of task loss to weight the objectives [32].

The most similar to our work is multi-task self-supervised learning done by Doersch and Zisserman [14], where four self-supervised tasks are jointly trained. ResNet101 [23] is used as their base architecture. Such a large network is problematic for the automotive industry due to the computational complexity and memory requirement. Further, no advance task weighting scheme are explored.

Our work exploits colorization and depth prediction as self-supervised tasks. With colorization, the network learns contextual information, whereas with the depth it learns about the 3D world. Additionally, we use a small network and explore the use of advance weighting schemes.

3 Joint supervised and self-supervised learning

In the present study, we use self-supervised tasks to boost the performance of a supervised task. Colorization and depth prediction are used as self-supervised tasks and semantic segmentation as a supervised task. Inference may be performed jointly or separately. Thus, self-supervised tasks do not have any influence on the run-time of the semantic segmentation. We propose an architecture with a shared encoder and task-specific decoders, as illustrated in Figure 1.

Each task has a separate loss. All losses are weighted and summed. The tasks are trained at the same time in an end-to-end fashion. Subsections 3.1 and 3.2 provide details on self-supervised tasks and semantic segmentation, respectively. Subsection 3.3 provides details on multi-task learning.

3.1 Self-supervised learning

Image colorization Inspired by previous work [24, 61, 30], our colorization operates in the CIE Lab space. Given the lightness channel L, the proposed method predicts the corresponding a and b channels of the color image. In this study, colorization is formulated as a multinomial classification

problem. The a and b channels are quantized into a small number of bins. The quantized channels are used as colorization ground truth, where the value of each pixel corresponds to a colorization class. To reduce the range of values in each bin and the total number of classes, the values of the a and b channels are clipped before the quantization. To ensure the majority of values that appear in the a and b channels are taken into account, we apply "two-sigma rule". From the training data, we estimate means (μ_a, μ_b) and standard deviations (σ_a, σ_b) of the color values for the a and b channels. The lower and upper values for clipping of the a and b channels are defined as $\mu_a \pm 2\sigma_a$ and $\mu_b \pm 2\sigma_b$, respectively. The resulting range for each of the two channels is divided into six bins, which results in a total of 36 classes. To predict colors, a softmax classifier is used. As some colors appear more frequently than others, class weighting is utilized, where median frequency balancing [15] is used to calculate the weights. Finally, the colorization loss is modelled as the weighted cross-entropy objective:

$$L_{color}(C, \hat{C}) = -\frac{1}{N} \sum_{i=0}^{N-1} w_i \, C_i \log(\hat{C}_i) \tag{1}$$

where N denotes the total number of pixels, w_i denotes the class specific weight for the i-th pixel, C_i denotes the colorization ground truth, and $\hat{C}_i = e^{zc_i} / \sum_c e^{zc_{i,c}}$ denotes the class predictions given the output zc of the final convolutional layer of the colorization decoder and the total number of color classes c.

Depth prediction Similar to prior work [21, 65, 17, 59, 57, 55, 37, 22], we propose depth prediction in which image reconstruction is used as self-supervision during training. Starting from a rectified pair of stereo images, our network learns how to reconstruct one image from the other. More specifically, during training, both left (I^l) and right (I^r) images are used and the network is trained to predict a field that transforms the right image into the left. The predicted field corresponds to image disparity (d). From the predicted disparity and the right image, we reconstruct the left image:

$$\hat{I}^l = \text{bilinear}(I^r, d) \tag{2}$$

where $\hat{I}^l$ denotes the reconstructed left image, and bilinear$(\cdot, \cdot)$ denotes the bilinear interpolation function. The reconstruction loss is modelled as the $L2$ objective:

$$L_{depth}(I^r, I^l, \hat{I}^l) = \frac{1}{N} \sum_{i=0}^{N-1} (I_i^l - \hat{I}_i^l)^2 \tag{3}$$

The sigmoid function ($d = e^{zd}/(e^{zd} + 1)$) is used to convert the output zd of the final convolutional layer of the depth decoder to the disparity. Finally, from the output disparity, the depth can be obtained by using $\hat{d} = bf/d$, where b and f denote the focal length and the baseline, respectively.

3.2 Semantic segmentation

Unlike the self-supervised tasks, semantic segmentation requires ground truth labels. The loss of semantic segmentation is modelled as the standard cross-entropy objective:

$$L_{seg}(S, \hat{S}) = -\frac{1}{N} \sum_{i=0}^{N-1} S_i \log(\hat{S}_i) \tag{4}$$

where S_i denotes the semantic ground truth for the i-th pixel, and $\hat{S}_i = e^{zs_i} / \sum_s e^{zs_{i,s}}$ denotes the class prediction given the output zs of the final convolutional layer of the segmentation decoder and the total number of semantic classes s.

3.3 Multi-task learning

We use self-supervised tasks (i.e., colorization and depth prediction) to boost the performance of semantic segmentation. Hence, the total loss in this study is a weighted sum of task-specific losses:

$$L_{total} = \lambda_1 L_{color} + \lambda_2 L_{depth} + \lambda_3 L_{seg} \tag{5}$$

where λ_1, λ_2, and λ_3 denote the task weights. As mentioned, it is important to achieve a good balance between the tasks. Consequently, we explore multiple task-weighing strategies (i.e, grid search, uncertainty based weighting [27], and dynamic weight average [32]).

4 Experiments and results

Implementation and evaluation details on both single-task and multi-task training with several task-weighting strategies are provided in sections 4.1 and 4.2, respectively. The results are presented on Cityscapes and KITTI datasets. Finally, we discuss our results in the context of existing approaches.

4.1 Implementation details

Network architecture In this study, we use a network architecture that is based on a U-net [46], in which the outputs of the encoder layers are combined with the inputs of the decoder layers through concatenation. The backbone of our network, is ResNet18 [23]. Each convolutional layer is followed by the batch normalization [25] and the rectified linear unit (ReLU), except for the last decoder layer, which is followed by a task-specific activation function as mentioned in sections 3.1 and 3.2.

Training procedure The proposed method is implemented in Keras [9]. As shown in Figure 1, the method operates in CIE Lab space. CIE Lab images are inputs to semantic segmentation and depth prediction, while colorization has the L channel repeated three times as an input. For all experiments, Adam gradient descent [28] is used, with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and initial learning rate of 0.001. A mini-batch of size 2 (batch size was limited by the hardware) is used in all experiments. Training is done for 100 and 20 epochs for Cityscapes and KITTI datasets, respectively. Additionally, as different inputs (Lab vs. L channel repeated three times) to the shared encoder may be problematic, experiments involving colorization were performed with an embedding convolutional 1x1 layer (i.e., L channel is feed into embedding layer which is used as an input to a shared encoder). However, the use of such embedding layer resulted in worse results that the use of L channel repeated three times.

We explore uncertainty based weighting, dynamic weight average, and a grid search as task-weighting strategies. For grid search, the weight of semantic segmentation is set to one (i.e., $\lambda_3 = 1$), while the weights of colorization (λ_1) and depth prediction (λ_2) are chosen from the following values: $0.1, 1, 10, 100$.

4.2 Evaluation

Extensive set of experiments is done to evaluate the performance of the proposed method. Different task combinations and weighting strategies are considered and the corresponding results are reported. We use only the standard metrics utilized in prior work (e.g., [21], [7], and [2]).

Cityscapes Cityscapes is a large-scale dataset with driving scenes from 50 cities. It contains 5000 pixel-level annotations (i.e., fine annotations) along with 20000 coarse annotations. Each annotation comes with the corresponding pair of rectified stereo images. The image resolution is 1024×2048 and the dataset includes 30 classes. We use a subset of 19 classes for the training and evaluation.

KITTI KITTI is a benchmark dataset containing data from different sensors. Pixel-level semantic segmentation annotations were recently added (200 training and 200 testing images). The classes are the same as in Cityscapes and the image resolution is 1242×375. In this study, the original training set of 200 images is randomly divided to 170 images for training and 30 images for validation.

Semantic segmentation (Single task baseline) The aforementioned datasets are used to demonstrate that self-supervised tasks improve the performance of semantic segmentation. In this study, each image from the Cityscapes dataset is resized to 1024×512.

Three semantic segmentation baselines that differ only in the encoder initialization are considered. The first (i.e., SSrandom) uses the Xavier initialization [20], the second (i.e., SSImageNet) has weights pre-trained for classification on ImageNet [47], and the third (i.e., SSCoarseCTS) has weights pre-trained by using the two self-supervised tasks on Cityscapes images with coarse annotations (i.e., CoarseCTS). In all three cases, the encoder is followed by a decoder and fine annotations of Cityscapes are used for training. The experiments are repeated for KITTI, where the corresponding Cityscapes baselines are used to initialize the weights due to small number of samples. The baseline results, for Cityscapes and KITTI, are reported in Table 1 and Table 3, respectively.

Multi-task learning Additional experiments are conducted in which we train semantic segmentation with either colorization and depth prediction or together with both of them. The encoders are pre-

Table 1: Semantic segmentation (SS), colorization (C), and disparity (D) prediction results on the Cityscapes (CTS) validation dataset. For SS and C, standard class mean intersection over union (mIOU) is used. For D standard absolute (AbsRel) and square (SqRel) errors, along with the root mean square error (RMSE) and its logarithmic (RMSE_log) counterpart are used. Different weighting strategies are evaluated: grid search (GS), uncertainty based weighting (UW) and dynamic weight average (DWA). For multi-task learning, metrics for a network initialized on ImageNet and coarse Cityscapes (within parenthesis) are reported.

		Sem. Segmentation	Colorization	Disparity*			
		Sem. mIOU	Color mIOU	AbsRel	SqRel	RMSE	RMSE_log
Single	SSrandom	0.58	n/a	n/a	n/a	n/a	n/a
	SSImageNet	0.61	n/a	n/a	n/a	n/a	n/a
	SSCoarseCTS	0.59	n/a	n/a	n/a	n/a	n/a
Two tasks	SS + C (GS: $\lambda_1 = 1$, $\lambda_2 = 1$)	0.61 (0.58)	0.50 (0.40)	n/a	n/a	n/a	n/a
	SS + C (UW)	0.59 (0.57)	0.49 (0.47)	n/a	n/a	n/a	n/a
	SS + C (DWA)	0.61 (0.59)	0.40 (0.43)	n/a	n/a	n/a	n/a
	SS + D (GS: $\lambda_1 = 1$, $\lambda_3 = 100$) (**Best**)	**0.64** (0.60)	n/a	0.24 (0.28)	2.07 (2.28)	3.09 (3.16)	0.30 (0.32)
	SS + D (UW)	0.62 (0.59)	n/a	0.25 (0.26)	2.11 (2.22)	3.08 (3.13)	0.31 (0.32)
	SS + D (DWA)	0.62 (0.58)	n/a	0.35 (0.33)	2.81 (2.78)	3.55 (3.53)	0.36 (0.36)
Multitask	SS + C + D (GS: $\lambda_1 = 1$, $\lambda_2 = 1$, $\lambda_3 = 100$)	0.62 (0.59)	0.48 (0.38)	0.28 (0.31)	2.26 (2.72)	3.13 (3.34)	0.32 (0.33)
	SS + C + D (UW)	0.60 (0.58)	0.48 (0.41)	0.25 (0.26)	2.27 (2.16)	3.06 (3.07)	0.32 (0.32)
	SS + C + D (DWA)	0.62 (0.59)	0.35 (0.43)	0.31 (0.31)	2.60 (2.64)	3.53 (3.52)	0.35 (0.36)

*The values are capped at 50.

Table 2: Semantic segmentation results (mean (m) and weighted (i) intersection over union (IOU)) on Cityscapes test dataset on both class and category level.

	Class mIOU	Class iIOU	Category mIOU	Category iIOU
Baseline (SSImageNet)	0.57	0.30	0.83	0.64
Best	0.60	0.31	0.84	0.65

trained on ImageNet and Cityscapes with coarse annotations. Further, different weighting strategies (i.e., grid search, uncertainty based weighting [27], and dynamic weight average [32]) are explored.

The results of multi-task learning for Cityscapes dataset are reported in Table 1. The best performance of semantic segmentation is achieved when trained together with depth prediction (ImageNet initialized, grid search weighting), and the worst performance is achieved when trained together with colorization (CoarseCTS initialized, uncertainty based weighting). Additionally, our best model and our baseline were evaluated on Cityscapes test data and the results are reported in Table 2. Examples of the results of our baseline and best model on Cityscapes data are shown in Figure 2.

Our results also indicate that with grid search and dynamic weight average, performance of semantic segmentation either remains the same or is improved when trained with one or more self-supervised tasks. In contrast, with uncertainty based weighting, in some cases, the performance degrades.

When trained as a single task, colorization mIOU is 0.45 (ImageNet) / 0.44 (CoarseCTS), whereas for depth prediction AbsRel is 0.27 (ImageNet) / 0.28 (CoarseCTS) and SqRel is 2.32 (ImageNet) / 2.44 (CoarseCTS). These results indicate that segmentation also improves performance of self-supervised tasks and that image color and depth are tightly linked to the semantics of a scene. We hypothesis that introducing explicit knowledge about scene semantics enables the network to learn richer features. Examples of results of self-supervised tasks are depicted in Figure 3.

For KITTI, Cityscapes pre-trained weights are used to initialize the network, after which the network is finetuned with KITTI data. The results are reported in Table 3. The best performance is obtained when semantic segmentation is trained with depth prediction (ImageNet initialized and uncertainty based weighting). Examples of the segmentation results on KITTI data are shown in Figure 4.

Model comparison Finally, we list mIOU performance of some recently proposed methods on Cityscapes dataset in Table 4. Our best solution yields mIOU of 0.60 (test set) / 0.64 (validation

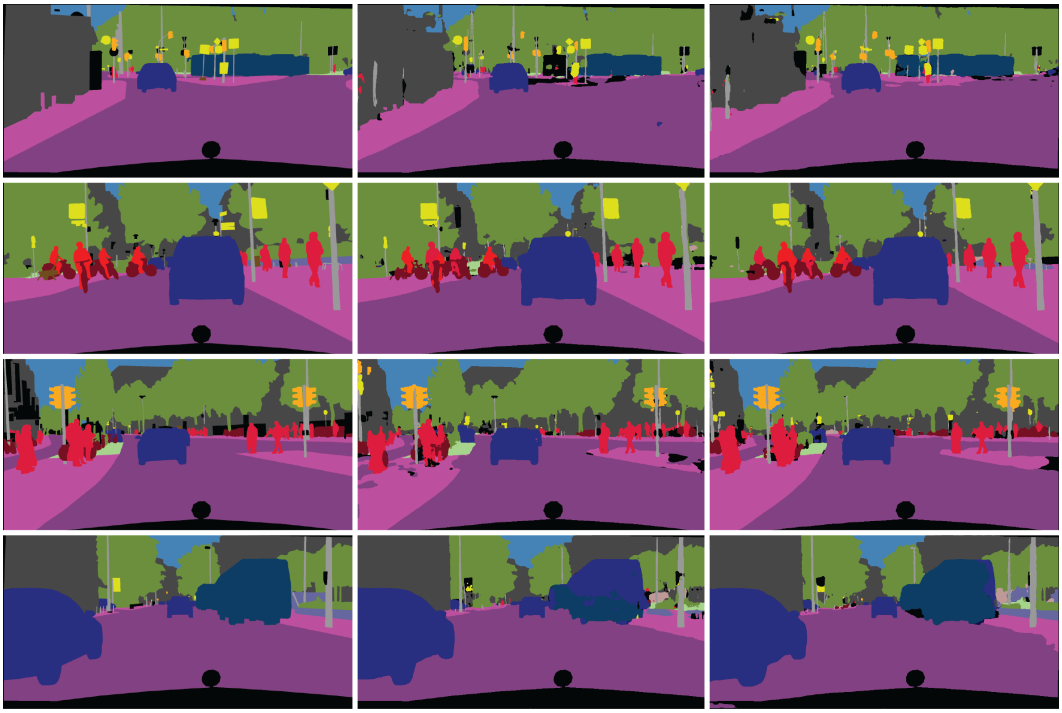

Figure 2: Examples of semantic segmentation results on the Cityscapes validation dataset. From left to right, images are as follows: the ground truth semantic segmentation, results of our baseline model (SSImageNet), and results of our best model. For example, an improvement of our best model over the baseline is visible for the bus class in the first and the bottom row.

Figure 3: Examples of self-supervised tasks results on the Cityscapes validation dataset. From left to right, images are as follows: the original images, results from the colorization task, the disparity ground truth, and results from the depth prediction task.

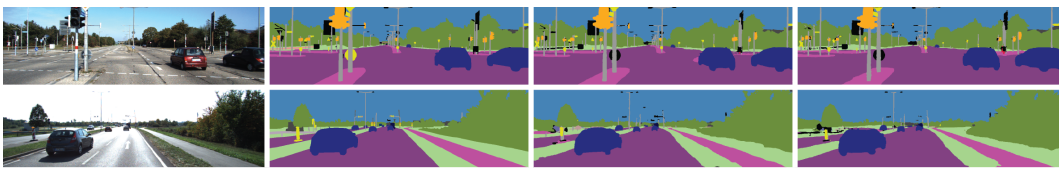

Figure 4: Examples of semantic segmentation results on KITTI dataset. From left to right, images are as follows: the original, the ground truth semantic segmentation, results from our baseline (SSImageNet), and results from our best model.

Table 3: Standard class mean intersection over union (mIOU) of semantic segmentation (SS) results on KITTI (internal validation) dataset. Different weighting strategies are evaluated: grid search (GS), uncertainty based weighting (UW) and dynamic weight average (DWA). For multi-task learning, metrics for a network initialized on ImageNet and coarse Cityscape (within parenthesis) are reported.

Single	SSrandom	0.46
	SSImageNet	0.47
	SSCoarseCTS	0.46
Two tasks	SS + C (GS)	0.45 (0.47)
	SS + C (UW)	0.43 (0.45)
	SS + C (DWA)	0.45 (0.48)
	SS + D (GS)	0.41 (0.47)
	SS + D (UW)	**0.49** (0.47)
	SS + D (DWA)	0.46 (0.48)
Multi	SS + C + D (GS)	0.48 (0.46)
	SS + C + D (UW)	0.46 (0.41)
	SS + C + D (DWA)	0.47 (0.46)

Table 4: Semantic segmentation results (class mean intersection over union) on Cityscapes dataset (‡ test dataset and † validation) for different methods.

SegNet [2]	0.56 ‡	
PSPNet [64]	0.78 ‡	0.74 †
DeepLab v3 [7]	0.81 ‡	
ENet [42] [1]	0.58 ‡	0.53 †
ICNet [63] [1]	0.69 ‡	0.56 †
UNet - ResNet18 [49]		0.58 †
UNet - MobileNet [49]		0.61 †
ResNet18 with distillation [35]		0.73 †
ShuffleNet V2 + DPC [53]		0.71 †
Ours - Baseline (SSImageNet)	0.57 ‡	0.61 †
Ours - Best	0.60 ‡	0.64 †

set) which is quantitatively better than some of the recently proposed solutions. It is outperformed by DeepLab v3 and PSPNet, which are networks with a much larger number of parameters. With respect to smaller networks, our solution is better than ENet (on both test and validation dataset) and ICNet (on validation dataset) and worse than ShuffleNet V2 with atrous convolutions [53], which reports results only on ten Cityscapes classes, and ResNet18 with distillation [35].

5 Conclusion and future work

In this paper, we propose self-supervised learning to improve the performance of the semantic segmentation. Two self-supervised tasks are explored: colorization and depth prediction. The approach is evaluated on two datasets: Cityscapes and KITTI. Several initialization and task-weighting strategies are investigated. We show an improvement of up to 3 % with respect to the baseline and we hypothesize that additional self-supervised tasks should improve the performance even further.

We show that colorization does not improve the performance of semantic segmentation, in contrast to depth prediction, which improves the performance on both Cityscapes and KITTI, indicating that information about the 3D environment is highly correlated to the semantics of an image.

Our results also show the relevance of task weighting in the context of multi-task learning. We show that grid search outperforms some of the recently proposed task-weighting methods when self-supervised tasks are used to improve the performance of a single supervised task. Grid search is a time consuming method, especially when the number of tasks is large. Hence, it is of the utmost importance to design a task-weighting method that performs well in this specific setting.

In conclusion, self-supervised learning improves the performance of semantic segmentation at no additional annotation and inference-related computational cost. Future work includes exploration of additional self-supervised tasks and a design of new task-weighting strategies.

Acknowledgments

We thank Amritpal Singh Gill (HERE Technologies) for his insights and initial discussions and Mihai Ilie (Valeo) for providing GPU support.

References

[1] A. Arnab, O. Miksik, and P. H. Torr. On the robustness of semantic segmentation models to adversarial attacks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[2] 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.

[3] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. One-shot video object segmentation. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2017.

[4] M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *The European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[5] R. Caruana. Multitask learning: A knowledge-based source of inductive bias. In *International Conference on Machine Learning (ICML)*, pages 41–48, 1993.

[6] L.-C. 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*, 40(4):834–848, 2017.

[7] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[8] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning (ICML)*, pages 794–803, 2018.

[9] F. Chollet et al. Keras. `https://keras.io`, 2015.

[10] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *International Conference on Machine Learning (ICML)*, pages 160–167, 2008.

[11] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[12] L. Deng, G. Hinton, and B. Kingsbury. New types of deep neural network learning for speech recognition and related applications: An overview. In *International Conference on Acoustics, Speech and Signal Processing*, pages 8599–8603, 2013.

[13] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. *IEEE International Conference on Computer Vision (ICCV)*, pages 1422–1430, 2015.

[14] C. Doersch and A. Zisserman. Multi-task self-supervised visual learning. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2051–2060, 2017.

[15] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2650–2658, 2015.

[16] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu. Dual attention network for scene segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3146–3154, 2019.

[17] R. Garg, V. K. B.G., G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *The European Conference on Computer Vision (ECCV)*, pages 740–756, 2016.

[18] A. Geiger. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[19] S. Gidaris, P. Singh, and N. Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations (ICLR)*, 2018.

[20] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[21] C. Godard, O. M. Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6602–6611, 2017.

[22] A. Gordon, H. Li, R. Jonschkowski, and A. Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. *arXiv preprint arXiv:1904.04998*, 2019.

[23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[24] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Transactions on Graphics*, 35(4):110:1–110:11, 2016.

[25] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[26] L. Jing and Y. Tian. Self-supervised visual feature learning with deep neural networks: A survey. *arXiv preprint arXiv:1902.06162*, 2019.

[27] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[28] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[29] I. Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[30] G. Larsson, M. Maire, and G. Shakhnarovich. Learning representations for automatic colorization. In *The European Conference on Computer Vision (ECCV)*, pages 577–593, 2016.

[31] H.-Y. Lee, J.-B. Huang, M. K. Singh, and M.-H. Yang. Unsupervised representation learning by sorting sequences. *IEEE International Conference on Computer Vision (ICCV)*, pages 667–676, 2017.

[32] S. Liu, E. Johns, and A. J. Davison. End-to-end multi-task learning with attention. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[33] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8759–8768, 2018.

[34] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *The European Conference on Computer Vision (ECCV)*, pages 21–37, 2016.

[35] Y. Liu, K. Chen, C. Liu, Z. Qin, Z. Luo, and J. Wang. Structured knowledge distillation for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[36] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.

[37] R. Mahjourian, M. Wicke, and A. Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5667–5675, 2018.

[38] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. Cross-stitch networks for multi-task learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[39] I. Misra, C. L. Zitnick, and M. Hebert. Shuffle and learn: Unsupervised learning using temporal order verification. In *The European Conference on Computer Vision (ECCV)*, pages 527–544, 2016.

[40] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. *The European Conference on Computer Vision (ECCV)*, pages 69–84, 2016.

[41] M. Noroozi, H. Pirsiavash, and P. Favaro. Representation learning by learning to count. *IEEE International Conference on Computer Vision (ICCV)*, pages 5898–5906, 2017.

[42] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016.

[43] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2536–2544, 2016.

[44] B. Ramsundar, S. M. Kearnes, P. Riley, D. Webster, D. E. Konerding, and V. S. Pande. Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072*, 2015.

[45] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.

[46] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.

[47] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of Computer Vision*, 115(3):211–252, 2015.

[48] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, 2018.

[49] M. Siam, M. Gamal, M. Abdel-Razek, S. Yogamani, and M. Jagersand. Rtseg: Real-time semantic segmentation comparative study. In *IEEE International Conference on Image Processing (ICIP)*, pages 1603–1607, 2018.

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

[51] N. Srivastava. Unsupervised learning of visual representations using videos. *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[52] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun. Multinet: Real-time joint semantic reasoning for autonomous driving. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1013–1020, 2018.

[53] S. Türkmen and J. Heikkilä. An efficient solution for semantic segmentation: Shufflenet v2 with atrous separable convolutions. In *Image Analysis*, pages 41–53. Springer, 2019.

[54] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5038–5047, 2017.

[55] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki. SfM-Net: Learning of structure and motion from video. *arXiv preprint arxiv:1704.07804*, 2017.

[56] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. P. Murphy. Tracking emerges by colorizing videos. In *The European Conference on Computer Vision (ECCV)*, 2018.

[57] Z. Yin and J. Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1983–1992, 2018.

[58] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[59] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. D. Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 340–349, 2018.

[60] J. Zhang, J. Yu, and D. Tao. Local deep-feature alignment for unsupervised dimension reduction. *IEEE Transactions on Image Processing*, 27(5):2420–2432, 2018.

[61] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In *The European Conference on Computer Vision (ECCV)*, pages 649–666, 2016.

[62] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6848–6856, 2018.

[63] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. Icnet for real-time semantic segmentation on high-resolution images. In *The European Conference on Computer Vision (ECCV)*, pages 405–420, 2018.

[64] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017.

[65] T. Zhou, M. R. G. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6612–6619, 2017.