




Self-distilled Self-supervised Depth Estimation in Monocular Videos

Julio Mendoza  and Helio Pedrini ^(✉) 

University of Campinas, Institute of Computing, Campinas, SP 13083-852, Brazil
helio@ic.unicamp.br

Abstract. In this work, we investigate approaches to leverage self-distillation via predictions consistency on self-supervised monocular depth estimation models. Since per-pixel depth predictions are not equally accurate, we propose a mechanism to filter out unreliable predictions. Moreover, we study representative strategies to enforce consistency between predictions. Our results show that choosing proper filtering and consistency enforcement approaches are key to obtain larger improvements on monocular depth estimation. Our method achieves competitive performance on the KITTI benchmark.

Keywords: Depth estimation · Self-distillation · Monocular videos

1 Introduction

Depth estimation is an essential task used in a wide range of applications in computer vision, for instance, 3D modeling, virtual and augmented reality, and robot navigation. Depth information can be obtained using sensors such as LIDAR or RGB-D cameras. However, in some scenarios, we cannot rely solely on them due to their limited range and operating conditions. Thus, alternative approaches such as estimating depth from images become more appealing. Supervised deep learning methods for depth estimation have shown impressive results. However, these approaches depend on the expensive acquisition of high-quality ground-truth depth data for training.

In contrast, self-supervised depth estimation approaches do not require ground-truth. Since the only inputs required are stereo images or monocular sequences, they can be trained on diverse data sets without depth labels. Self-supervised methods leverage geometric priors to learn image reconstruction as an auxiliary task. Depth maps are obtained as an intermediary result of the image reconstruction process.

Several works have shown that self-supervised depth estimation can be benefited from learning additional auxiliary tasks, for example, self-distillation. Self-distillation methods aim to improve a model performance by distilling knowledge from the model itself. An interesting strategy for performing self-distillation consists in extracting information from distorted versions of the input data by enforcing consistency between their predictions [26]. In this work,

we propose a self-distillation approach via prediction consistency to improve self-supervised depth estimation from monocular videos. Since enforcing consistency between predictions that are unreliable cannot provide useful knowledge, we propose an strategy to filter out unreliable predictions. Moreover, the idea of enforcing consistency between predictions has been widely explored in self-distillation [8, 18, 27] and semi-supervised learning [1, 13, 20–22, 24]. In order to explore the space of consistency enforcement strategies, we adapt and evaluate representative approaches on the self-supervised depth estimation task.

In summary, the main contributions of our work are the following: (i) the proposition of a multi-scale self-distillation method based on prediction consistency, (ii) the design of an approach to filter unreliable per-pixel predictions on the pseudo-labels used in self-distillation, and (iii) the exploration and adaptation of several consistency enforcement strategies for self-distillation. To validate our method, we show a detailed evaluation and a comparison against state-of-the-art methods on the KITTI benchmark. Our code is available at <https://github.com/jmendozais/SDSSDepth>.

2 Related Work

Self-supervised Depth Estimation. The main intuition of self-supervised depth estimation methods is to leverage multi-view geometry relations computed from depth and camera motion predictions to reconstruct one view with the pixel values from another view. Depth and camera motion can be obtained from deep networks that are trained by minimizing the reconstruction error. Garg et al. [9] used this intuition to train a depth network using stereo pairs as views. Similarly, Zhou et al. [29] proposed a method to obtain the views from monocular sequences and used deep networks to estimate relative pose and depth maps. Several works addressed limitations of these methods such as inaccurate prediction in occluded regions [16] and regions with moving objects [2, 12]. Other works aim to improve the learning signal by enforcing consistency between several representations of the scene [3, 16, 17] or by using auxiliary tasks such as semantic segmentation or self-distillation [14, 15]. Similarly to [14], our method leverages self-distillation to improve depth estimation. Unlike prior work, we focus on exploring representative strategies to distill knowledge from the predictions of our model.

Pseudo-labeling Approaches for Self-supervised Depth Estimation.

Many self-supervised methods trained from stereo images [23, 25] or monocular sequences [4, 5, 14, 15, 23] rely on pseudo-labels to provide additional supervision for training their depth networks. These methods can use state-of-the-art classical stereo matching algorithms [23], external deep learning methods [4, 5], or their own predictions [25] to obtain pseudo-labels. Since the quality of the pseudo-labels is not always guaranteed, methods filter out unreliable per-pixel predictions based on external confidence estimates [4, 5, 23] or uncertainty estimates that are a result of the method itself [25]. Additionally, some methods

leverage multi-scale predictions for creating pseudo-labels by using the predictions at the highest-resolution as pseudo-labels to supervise predictions at lower resolutions [28] or by selecting, per pixel, the prediction with the lowest reconstruction error among the multi-scale predictions [19].

We focus on methods that use their own prediction as pseudo-labels. For example, Kaushik et al. [14] augmented a self-supervised method by performing a second forward pass with strongly perturbed inputs. The predictions from the second pass are supervised with predictions of the first pass. Liu et al. [15] proposed to leverage the observation depth maps predicted from day-time images are more accurate than predictions from night-time images. They used predictions from day-time images as pseudo-labels and train a specialized network with night-time images synthesized using a conditional generative model.

Self-distillation. These methods let the target model leverage information from itself to improve its performance. An approach is to transfer knowledge from an instance of the model, previously trained, via predictions [8, 18, 27] and/or features to a new instance of the model. This procedure could be repeated iteratively. Self-distillation has a regularization effect on neural networks. It was shown that, at earlier iterations, self-distillation reduces overfitting and increases test accuracy, however, after too many iterations, the test accuracy declines and the model underfits [18].

Self-distillation has been extensively explored, mainly in image classification problems. An approach performs distillation by training instances of a model sequentially such that a model trained on a previous iteration is used as a teacher for the model trained in the current iteration [8]. Similarly, Yang et al. [27] proposed to train a model in a single training generation imitating multiple training generations using a cyclic learning rate scheduler and using the snapshots obtained at the end of the previous learning rate cycle as a teacher. Our work explores the idea of leveraging multiple snapshots in a single training generation on the self-supervised depth estimation problem.

Consistency Regularization. Enforcing consistency between predictions obtained from perturbed views of input examples is one of the main principles behind consistency regularization approaches on deep semi-supervised works. An early method [20] used this principle doing several forward passes on perturbed versions of the input data. Furthermore, other methods showed that the usage of advanced data augmentation perturbations [24] or a combination of a weak and strong data augmentation perturbation in a teacher-student training scheme [21] can be helpful to improve the resulting models.

Existing works showed that average models, i.e., models whose weights are the average of the model being trained at different training steps, can be more accurate [1, 13, 22]. Average models can be used, as teachers, to obtain more accurate pseudo-labels [1, 22]. Moreover, the use of cyclic learning rate schedulers can improve the quality of the models that are averaged and the resulting model at accuracy and generalization [13], as well as it can be adapted to the consistency

regularization framework [1]. Similarly to [1], our method uses a cyclic cosine annealing learning rate schedule to obtain a better teacher model.

3 Depth Estimation Method in Monocular Videos

3.1 Preliminaries

Our method is built on self-supervised depth estimation approaches that use view reconstruction as main supervisory signal [29]. These approaches require to find correspondences between pixel coordinates on frames that represent views of the same scene. We represent this correspondences in Eq. 1.

$$x_s \sim \mathbf{K}\mathbf{T}_{\mathbf{t} \rightarrow \mathbf{s}}\mathbf{D}_{\mathbf{t}}(x_t)\mathbf{K}^{-1}x_t \quad (1)$$

We reconstruct the target frame using the correspondences and the pixel intensities in the source frame $\hat{\mathbf{I}}_{\mathbf{s} \rightarrow \mathbf{t}}(x_t) = \mathbf{I}_{\mathbf{s}}(x_s)$. This process is known as image warping. This approach requires the dense depth map $\mathbf{D}_{\mathbf{t}}$ of the target image, which we aim to reconstruct, the Euclidean transformation $\mathbf{T}_{\mathbf{t} \rightarrow \mathbf{s}}$, and camera intrinsics \mathbf{K} . Our model used to convolutional neural network to predict the depth maps and the Euclidean transformation, and assumes that the camera intrinsics are given. The networks are trained using the adaptive consistency loss \mathcal{L}_{ac} . We refer the reader to [17] for a detailed explanation.

3.2 Self-distillation via Prediction Consistency

The core idea of self-distillation based on prediction consistency is to provide additional supervision to the model by enforcing consistency between the depth map predictions obtained from different perturbed views of an input image. Our self-distillation approach applies two different data augmentation perturbations to an input snippet. To use less computational resources, we use snippets of two frames $\mathcal{I} = \{\mathbf{I}_{\mathbf{t}}, \mathbf{I}_{\mathbf{t}+1}\}$. The model predicts the depth maps for all images in the input snippet. Since we need to apply two data augmentation perturbations, we have two depth maps for each frame in the snippet. Then, we enforce consistency by minimizing the difference between the predicted depth maps for each frame.

There are several approaches to enforce consistency between prediction. The simplest variation of our method use the pseudo-label approach. It considers one of depth maps as pseudo-label $\mathbf{D}_{\mathbf{k}}^{(\text{pl})}$, which implies that gradients are not back-propagated through it, and the other depth map as prediction $\mathbf{D}_{\mathbf{k}}^{(\text{pred})}$. In Sect. 3.4, we improve our method considering other consistency enforcement strategies. Moreover, we enforce prediction consistency using the mean squared error (MSE) as difference measure. In addition, we filter the unreliable depth values on the pseudo-label using a composite mask. Equation 2 shows the self-distillation loss term for a snippet \mathcal{I} .

$$\mathcal{L}_{sd} = \frac{1}{|\mathcal{I}|} \sum_{\mathbf{I}_{\mathbf{k}} \in \mathcal{I}} \frac{1}{|\mathbf{M}_{\mathbf{k}}^{(\text{c})}|} \sum_{x \in \Omega(\mathbf{I}_{\mathbf{k}})} \mathbf{M}_{\mathbf{k}}^{(\text{c})}(x) \left(\mathbf{D}_{\mathbf{k}}^{(\text{pl})}(x) - \mathbf{D}_{\mathbf{k}}^{(\text{pred})}(x) \right)^2 \quad (2)$$

where \mathbf{I}_k is a frame in the snippet and $\mathbf{M}_k^{(c)}$ is its composite mask, $\Omega(\mathbf{I}_k)$ is the set of pixel coordinates, and $\mathbf{D}_k^{(pl)}$ and $\mathbf{D}_k^{(pred)}$ represent the pseudo-label and predicted depth maps, respectively.

Since our model predicts the depth maps at multiple scales, we compute the self-distillation loss for each scale. We assume that the pseudo-label at the finest scale is more accurate than the pseudo-labels at coarser scales. Thus, we only use the finest pseudo-label. We upscale the predictions to the finest scale to match the pseudo-label scale. Finally, we compute the self-distillation loss for each scale. Figure 1 depicts our self-distillation approach.

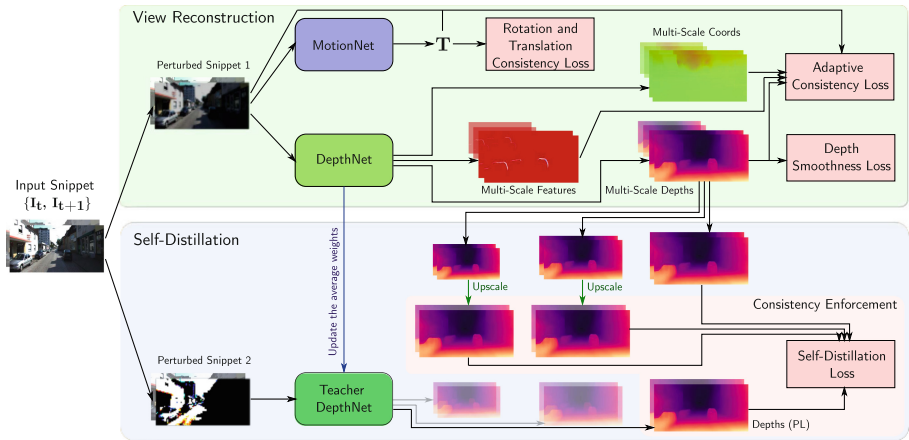


Fig. 1. Overview of our method. The self-distillation component leverages the multi-scale predictions obtained from the view reconstruction component. The predictions are upscaled to the finest resolution. More accurate predictions are obtained from the teacher model. The teacher predictions at the finest resolution are used as pseudo-labels to improve the predictions obtained from view reconstruction.

3.3 Filtering Pseudo-labels

We noticed empirically that unreliable depth prediction produces very large differences between pseudo-labels and predictions. These very large differences make training unstable and do not allow the model to converge randomly. We address this problem by excluding pixels with very large differences using a threshold value. In this section, we present two schemes to determine the threshold.

In the first scheme, we compute the threshold as a percentile P on the pseudo-label and prediction differences for all pixels in a batch of snippets. Then, we create valid mask considering as valid all the pixels with differences smaller than the threshold, as shown in Eq. 3.

$$\mathbf{M}^{(p)}(x) = [(\mathbf{D}^{(pl)}(x) - \mathbf{D}^{(pred)}(x))^2 < P] \quad (3)$$

where $[\cdot]$ denotes the Iverson bracket operator. The final mask is obtained combining the latter mask with the compound mask. The final mask could be expressed as $\mathbf{M} = \mathbf{M}^{(p)} \odot \mathbf{M}^{(c)}$, where \odot represents the element-wise product. Finally, we replace $\mathbf{M}^{(c)}$ with \mathbf{M} in Eq. 3.

We believe that the idea of using a threshold obtained from the distribution of differences by batch might be detrimental because we do not take into consideration that batches with reliable predictions should have thresholds that exclude less pixels than the threshold used on batches with more unreliable predictions.

In the second scheme, we address this limitation by approximating a global threshold $P^{(\text{EMA})}$ using the exponential moving average (EMA) of the percentile values for each batch during training. Another advantage of using a moving average is that we take into consideration that the distribution of depth differences change during training. This means that, when the depth differences become smaller during training, the threshold changes by increasing the weight of the percentiles from latter batches on the average. Equation 4 shows our global threshold approximation.

$$P_t^{(\text{EMA})} = P_{t-1}^{(\text{EMA})} \cdot \beta + P_t \cdot (1 - \beta) \quad (4)$$

where P_t is the threshold computed from the batch at the t training iteration, $P_t^{(\text{EMA})}$ is threshold obtained using the EMA at the t training iteration, and β controls the influence of the previous moving average percentile and the current percentile into the computation of the current threshold. Similarly to the first scheme, we compute a valid mask $\mathbf{M}^{(\text{EMA})}$ using $P^{(\text{EMA})}$, we combine this mask with the compound mask $\mathbf{M} = \mathbf{M}^{(\text{EMA})} \odot \mathbf{M}^{(c)}$ and, finally, we use \mathbf{M} instead of $\mathbf{M}^{(c)}$ in Eq. 2.

3.4 Consistency Enforcement Strategies

In previous sections, we used a pseudo-label strategy to enforce consistency between depth predictions. Here, we describe representative consistency enforcement strategies adapted to our self-distillation approach. Figure 2 depicts these consistency enforcement strategies. We named each strategy similarly to the methods that introduced the key idea into external domains [1, 13, 20, 22]. Similarly to the pseudo-label strategy, variants of our method that use the strategies described in this section also adopt the second scheme described in Sect. 3.3 to filter out unreliable per-pixel predictions before computing the predictions difference.

Π -Model. Similarly to the pseudo-label approach, this strategy consists in enforcing consistency between prediction from two perturbed views of the same input. In contrast with the pseudo-label approach, the gradients are back-propagated through both predictions.

Mean Teacher. Instead of using the same depth network to generate the pseudo-labels and the predictions, we can introduce a teacher network that can

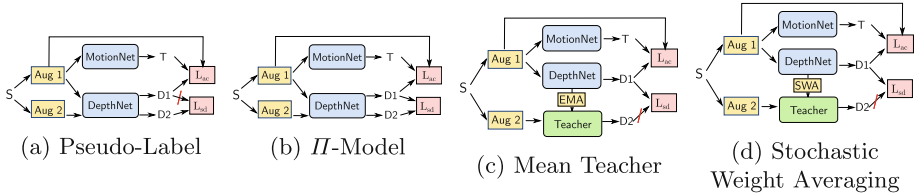


Fig. 2. Simplified views of the consistency enforcement strategies. S denotes the input snippet, $Aug 1$ and $Aug 2$ denote two perturbed views of the input snippet, T denotes the camera motion transformation, $D1$ and $D2$ denote depth maps predictions, L_{ac} denotes the adaptive consistency loss, L_{sd} denotes the self-distillation loss, and red lines $\color{red}{\text{---}}$ mark connections where the gradients are not back-propagated. (Color figure online).

potentially predict more accurate pseudo-labels, and provide better supervisory signal to the model currently being trained, the student network. In this approach, the teacher depth network weights are the EMA of the depth network weights in equally spaced training iterations.

Stochastic Weight Averaging. Similarly to the mean teacher strategy, we set the teacher depth network weights as the EMA of the depth network weights. In contrast, the training process is split into several cycles. At each cycle, the learning rate decreases and the teacher depth network is updated with the weights of depth network at the end of the last epoch of each training cycle, where the learning rate reaches its lowest value.

In the first generation of the training process, we use the student network to predict pseudo-label. Once the model has converged to a proper local optimum, we use its weights to initialize the teacher network. Then, in the following cycles, the training process mimics multiple training generations using a cyclic cosine annealing learning rate. At the end of each cycle, when the learning rate reaches its lowest value, and model likely converged to a good local optimum, we update the weights the teacher network using EMA with the student network weights.

3.5 Additional Considerations

Final Loss. The overall loss is a weighted sum of our self-distillation loss \mathcal{L}_{sd} , adaptive consistency loss \mathcal{L}_{ac} [17], depth smoothness loss \mathcal{L}_{ds} , translation consistency loss \mathcal{L}_{tc} , and rotation consistency loss \mathcal{L}_{rc} . The rotation and translation consistency losses are similar to the cyclic consistency loss defined in [12]. In contrast, our translation consistency loss only considers camera motion. Equation 5 shows our final loss.

$$\mathcal{L} = \sum_{i \in S} \frac{1}{2^i} \left(\mathcal{L}_{ac}^{(i)} + \lambda_{ds} \mathcal{L}_{ds}^{(i)} + \lambda_{sd} \mathcal{L}_{sd}^{(i)} \right) + \lambda_{rc} \mathcal{L}_{rc} + \lambda_{tc} \mathcal{L}_{tc} \quad (5)$$

where \mathcal{S} is the set of scales and λ_{sd} , λ_{ds} , λ_{rc} , λ_{tc} is the weight of the self-distillation, depth smoothness, rotation consistency, and translation consistency loss terms, respectively.

4 Evaluation

4.1 Experimental Setup

Dataset. We use the KITTI benchmark [10]. It is composed of video sequences with 93 thousand images acquired through high-quality RGB cameras captured by driving on rural areas and highways of a city.

We used the Eigen split [6] with 45023 images for training and 687 for testing. Moreover, we partitioned the training set on 40441 for training, 4582 for validation. For result evaluation, we used the standard metrics.

Training. Our networks are trained using ADAM optimization algorithm using $\beta_1 = 0.9$, $\beta_2 = 0.999$. We used the batch size of 4 snippets. We use 2-frame snippets unless otherwise specified. We resize the frames to resolutions of 416×128 pixels unless otherwise specified.

The training process is divided into three stages. In the first stage, we train the model with the self-distillation loss disabled. The model is trained using a learning rate of $1e-4$ during 15 epochs, and then its reduced to $1e-5$ during 10 additional epochs. In the second stage, we train the model with self-distillation term enabled. The model is trained with a learning rate of $1e-5$ during 10 additional epochs. Finally, in the third stage, we train the model enabling SWA teacher for the depth network with a cyclical cosine learning rate schedule with an upper bound of $1e-4$, a lower bound of $1e-5$, and using 4 cycles of 6 epochs each, unless otherwise specified.

4.2 Self-distillation via Prediction Consistency

Table 1 shows the results with the simplest variant of our self-distillation approach. The results show a consistent improvement when self-distillation loss is used. The model trained with self-distillation loss outperforms the baseline at all error metrics and almost all accuracy metrics.

When searching for the optimal weight λ_{sd} for the self-distillation term, we noticed that large λ_{sd} values allow to obtain good results. However, due to large depth differences, the model diverges on some executions. Due to this instability, we use a smaller $\lambda_{sd} = 1e2$. This observation motivated us to explore approaches to filter unreliable predictions.

4.3 Filtering Pseudo-labels

Table 2 shows that our two filtering strategies outperform that variation of our method does not use any additional filtering approach other than the composite

Table 1. Comparison of the baseline model and the variation of our method that uses the *pseudo-label* strategy.

Method	↓ Lower is better				↑ Higher is better		
	Abs Rel	Sq Rel	RMSE	LRMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Baseline	0.128	1.005	5.152	0.204	0.848	0.951	0.979
PL	0.126	0.907	5.068	0.202	0.847	0.951	0.980

mask in the majority of error and accuracy metrics. Moreover, the results show that the approach that uses the EMA of the percentiles to estimate the threshold is better than using only the percentile of each batch.

Table 2. Comparison of variants of our method with and without filtering strategies. P denotes that we filtered pseudo-labels using a percentile by batch as thresholds, and $P^{(EMA)}$ denotes that we filtered pseudo-labels using a threshold that is the EMA computed from percentiles of the batches during training iterations.

Method	↓ Lower is better				↑ Higher is better		
	Abs Rel	Sq Rel	RMSE	LRMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
PL (w/o filtering)	0.126	0.907	5.068	0.202	0.847	0.951	0.980
PL + P	0.126	0.911	5.033	0.203	0.847	0.952	0.980
PL + $P^{(EMA)}$	0.126	0.904	5.024	0.202	0.847	0.952	0.980

4.4 Consistency Enforcement Strategies

Table 3 shows that, regardless the consistency enforcement strategy, self-distillation via prediction consistency can improve the performance of our baseline model. Moreover, the results show that the variant that uses SWA strategy outperforms the other consistency enforcement strategies in most of the error and accuracy metrics. This variant is used as our final model. Some qualitative results are illustrated in Fig. 3.

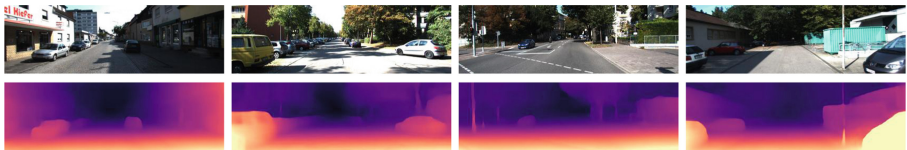
**Fig. 3.** Qualitative results. Depths maps obtained using our final model.

Table 3. Comparison of the representative consistency enforcement strategies. PL denotes the pseudo-label, Π M denotes the Π -Model, MT denotes the mean teacher, and SWA denotes the stochastic weight averaging strategy.

Method	↓ Lower is better				↑ Higher is better		
	Abs Rel	Sq Rel	RMSE	LRMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Baseline	0.128	1.005	5.152	0.204	0.848	0.951	0.979
PL + $P^{(EMA)}$	0.126	0.904	5.024	0.202	0.847	0.952	0.980
Π -M + $P^{(EMA)}$	0.126	0.902	5.041	0.202	0.847	0.952	0.980
MT + $P^{(EMA)}$	0.126	0.898	5.061	0.201	0.846	0.952	0.981
SWA + $P^{(EMA)}$	0.125	0.881	5.056	0.202	0.848	0.952	0.980

4.5 State-of-the-Art Comparison

Table 4 shows a quantitative comparison with state-of-the-art methods. Our method outperforms methods that explicitly address moving objects such as [2, 3, 12]. Fang et al. [7] obtained better results due to the usage of VGG16 as encoder, which has 138 million parameters. In contrast, our method uses a smaller encoder ResNet-18, which has 11 million parameters. The results show that our method achieves competitive performance when compared to state-of-the-art methods.

Table 4. Results of depth estimation on the Eigen split of the KITTI dataset. We compared our results against several methods of the literature. To allow a fair comparison, we report the results of competitive methods trained with a resolution of 416×128 pixels. N.F. denotes the number of frames in the input snippet (*) indicates newly results obtained from an official repository. (-ref.) indicates that the online refinement component is disabled.

Method	N.F.	↓ Lower is better				↑ Higher is better		
		Abs Rel	Sq Rel	RMSE	Log RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Gordon et al. [12]	2	0.129	0.959	5.230	0.213	0.840	0.945	0.976
Our method	2	0.125	0.881	5.056	0.202	0.848	0.952	0.980
Zhou et al. [29]*	3	0.183	1.595	6.709	0.270	0.734	0.902	0.959
Mahjourian et al. [16]	3	0.163	1.240	6.220	0.250	0.762	0.916	0.967
Casser et al. [2] (-ref)	3	0.141	1.026	5.290	0.215	0.816	0.945	0.979
Chen et al. [3] (-ref)	3	0.135	1.070	5.230	0.210	0.841	0.948	0.980
Godard et al. [11]	3	0.128	1.087	5.171	0.204	0.855	0.953	0.978
Our method	3	0.123	0.906	5.083	0.200	0.856	0.953	0.980
Fang [7]	3	0.116	–	4.850	0.192	0.871	0.959	0.982

5 Conclusions

We showed that to take full advantage of self-distillation, we need to consider additional strategies, such as the filtering approaches we proposed in this work, to deal with unreliable predictions. We demonstrated that choosing a proper consistency enforcement strategy in self-distillation is important. Our results suggest that the

features of consistency enforcement strategies, such as (i) enforcing teacher quality and (ii) enforcing difference between teacher and student network weights, which are embedded in the SWA strategy, are important to obtain larger improvements. We explored various strategies to benefit from self-distillation for self-supervised depth estimation when the input are monocular sequences. Moreover, we believe that our findings can provide useful insights to leverage self-distillation in methods that use stereo sequences as input, as well as semi-supervised and supervised methods.

References

1. Athiwaratkun, B., Finzi, M., Izmailov, P., Wilson, A.G.: There are many consistent explanations of unlabeled data: why you should average. In: International Conference on Learning Representations (2019)
2. Casser, V., Pirk, S., Mahjourian, R., Angelova, A.: Depth prediction without the sensors: leveraging structure for unsupervised learning from monocular videos. In: AAAI Conference on Artificial Intelligence, vol. 33, pp. 8001–8008 (2019)
3. Chen, Y., Schmid, C., Sminchisescu, C.: Self-supervised learning with geometric constraints in monocular video: connecting flow, depth, and camera. In: IEEE International Conference on Computer Vision, pp. 7063–7072 (2019)
4. Cho, J., Min, D., Kim, Y., Sohn, K.: Deep monocular depth estimation leveraging a large-scale outdoor stereo dataset. *Expert Syst. Appl.* **178**, 114877 (2021)
5. Choi, H., et al.: Adaptive confidence thresholding for monocular depth estimation. In: IEEE/CVF International Conference on Computer Vision, pp. 12808–12818 (2021)
6. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Advances in Neural Information Processing Systems, pp. 2366–2374 (2014)
7. Fang, Z., Chen, X., Chen, Y., Gool, L.V.: Towards good practice for CNN-based monocular depth estimation. In: IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1091–1100 (2020)
8. Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., Anandkumar, A.: Born again neural networks. In: International Conference on Machine Learning, pp. 1607–1616. PMLR (2018)
9. Garg, R., Bg, V.K., Carneiro, G., Reid, I.: Unsupervised CNN for single view depth estimation: geometry to the rescue. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 740–756. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_45
10. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: the KITTI dataset. *Int. J. Robot. Res.* **32**(11), 1231–1237 (2013)
11. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth prediction. In: International Conference on Computer Vision, October 2019
12. Gordon, A., Li, H., Jonschkowski, R., Angelova, A.: Depth from videos in the wild: unsupervised monocular depth learning from unknown cameras. arXiv preprint [arXiv:1904.04998](https://arxiv.org/abs/1904.04998) (2019)
13. Izmailov, P., Podoprikin, D., Garipov, T., Vetrov, D., Wilson, A.G.: Averaging weights leads to wider optima and better generalization. arXiv preprint [arXiv:1803.05407](https://arxiv.org/abs/1803.05407) (2018)

14. Kaushik, V., Jindgar, K., Lall, B.: ADAADepth: adapting data augmentation and attention for self-supervised monocular depth estimation. arXiv preprint [arXiv:2103.00853](https://arxiv.org/abs/2103.00853) (2021)
15. Liu, L., Song, X., Wang, M., Liu, Y., Zhang, L.: Self-supervised monocular depth estimation for all day images using domain separation. In: IEEE/CVF International Conference on Computer Vision, pp. 12737–12746 (2021)
16. Mahjourian, R., Wicke, M., Angelova, A.: Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5667–5675 (2018)
17. Mendoza, J., Pedrini, H.: Adaptive self-supervised depth estimation in monocular videos. In: Peng, Y., Hu, S.-M., Gabbouj, M., Zhou, K., Elad, M., Xu, K. (eds.) ICIG 2021. LNCS, vol. 12890, pp. 687–699. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87361-5_56
18. Mobahi, H., Farajtabar, M., Bartlett, P.: Self-distillation amplifies regularization in Hilbert space. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 3351–3361. Curran Associates, Inc. (2020)
19. Peng, R., Wang, R., Lai, Y., Tang, L., Cai, Y.: Excavating the potential capacity of self-supervised monocular depth estimation. In: IEEE International Conference on Computer Vision (2021)
20. Sajjadi, M., Javanmardi, M., Tasdizen, T.: Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In: Advances in Neural Information Processing Systems, vol. 29, pp. 1163–1171 (2016)
21. Sohn, K., et al.: FixMatch: simplifying semi-supervised learning with consistency and confidence. In: Advances in Neural Information Processing Systems, vol. 33 (2020)
22. Tarvainen, A., Valpola, H.: Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems, vol. 30. Curran Associates, Inc. (2017)
23. Tonioni, A., Poggi, M., Mattocchia, S., Di Stefano, L.: Unsupervised domain adaptation for depth prediction from images. IEEE Trans. Pattern Anal. Mach. Intell. **42**(10), 2396–2409 (2019)
24. Xie, Q., Dai, Z., Hovy, E., Luong, T., Le, Q.: Unsupervised data augmentation for consistency training. In: Advances in Neural Information Processing Systems, vol. 33 (2020)
25. Xu, H., et al.: Digging into uncertainty in self-supervised multi-view stereo. In: IEEE/CVF International Conference on Computer Vision, pp. 6078–6087 (2021)
26. Xu, T.B., Liu, C.L.: Data-distortion guided self-distillation for deep neural networks. In: AAAI Conference on Artificial Intelligence, vol. 33, pp. 5565–5572 (2019)
27. Yang, C., Xie, L., Su, C., Yuille, A.L.: Snapshot distillation: teacher-student optimization in one generation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2859–2868 (2019)
28. Yang, J., Alvarez, J.M., Liu, M.: Self-supervised learning of depth inference for multi-view stereo. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7526–7534 (2021)
29. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1851–1858 (2017)