SegStereo: Exploiting Semantic Information for Disparity Estimation

Guorun Yang[1], Hengshuang Zhao[2], Jianping Shi[3], Zhidong Deng[1(✉)], and Jiaya Jia[2,4]

[1] Department of Computer Science, State Key Laboratory of Intelligent Technology and Systems, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing, China
ygr13@mails.tsinghua.edu.cn, michael@mail.tsinghua.edu.cn
[2] The Chinese University of Hong Kong, Shatin, Hong Kong
hszhao@cse.cuhk.edu.hk, leojia@cse.cuhk.edu.hk
[3] SenseTime Research, Beijing, China
shijianping@sensetime.com
[4] Tencent YouTu Lab, Shenzhen, China

Abstract. Disparity estimation for binocular stereo images finds a wide range of applications. Traditional algorithms may fail on featureless regions, which could be handled by high-level clues such as semantic segments. In this paper, we suggest that appropriate incorporation of semantic cues can greatly rectify prediction in commonly-used disparity estimation frameworks. Our method conducts semantic feature embedding and regularizes semantic cues as the loss term to improve learning disparity. Our unified model SegStereo employs semantic features from segmentation and introduces semantic softmax loss, which helps improve the prediction accuracy of disparity maps. The semantic cues work well in both unsupervised and supervised manners. SegStereo achieves state-of-the-art results on KITTI Stereo benchmark and produces decent prediction on both CityScapes and FlyingThings3D datasets.

Keywords: Disparity estimation · Semantic cues
Semantic feature embedding · Softmax loss regularization

1 Introduction

Disparity estimation is a fundamental problem in computer vision. It is important in depth prediction, scene understanding, autonomous driving, to name a few. The main goal of disparity estimation is to find corresponding pixels from

G. Yang and H. Zhao—Equal contribution.

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-01234-2_39) contains supplementary material, which is available to authorized users.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11211, pp. 660–676, 2018.
https://doi.org/10.1007/978-3-030-01234-2_39

stereo images for inferring object distance according to the displacement between matching pixels.

Most previous methods [5,7,15,36] used hand-crafted reliable features to represent image patches and then selected matching pairs. They either formulate the task as supervised learning [22,26] based on current labeled dataset [14,41], or resort to unsupervised learning to form photometric loss for disparity prediction [13,17]. Recently, with the development of deep neural networks, the performance of disparity estimation is significantly improved [43]. The deep feature extracted from networks can exploit inherent global information in paired input compared to traditional methods, therefore benefits from a large number of training data either in supervised or unsupervised manner.

Although deep learning based methods produce impressive feature representation given its large receptive field, it is still difficult to overcome local ambiguity, which is a common problem in disparity estimation. For example, in Fig. 1, the disparity prediction in the center of road and vehicle area is not correct. It is because the matching clues for disparity estimation on those ambiguous areas are not enough to guide the model to seek correct direction for convergence, which is however the central objective for both supervised and unsupervised stereo learners.

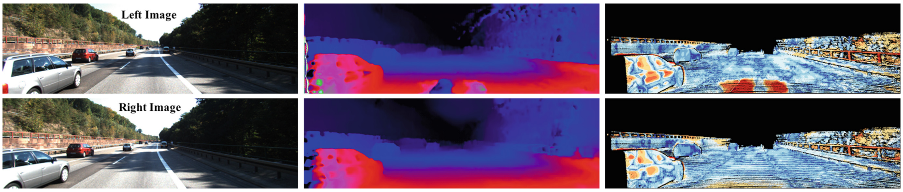

Fig. 1. Examples of prediction of unsupervised models on KITTI Stereo dataset. Left: input stereo images. Top-middle and top-right: colorized disparity and error maps predicted without semantic clues. Bottom-middle and bottom-right: colorized disparity and error maps predicted by *SegStereo*. With the guidance of semantic cues, disparity estimation of *SegStereo* is more accurate especially on the local ambiguous areas.

Human can perform binocular alignment well at ambiguous areas by exploiting more cues such as global perception of foreground and background, scaling relative to the known size of familiar objects, and semantic consistency for individuals. Such ambiguous areas in disparity estimation always locate in the central region given a big target. They are easy to deal with by semantic classification.

Based on the above-mentioned observation, we design an unified model called *SegStereo* that incorporates semantic cues into backbone disparity estimation network. Basically, we use the ResNet [19] with correlation operation [11] as the encoder and several deconvolutional blocks as decoder to regress a full-size disparity map. The correlation operation is designed in [11] to compute matching cost volumes based on pairs of feature maps. A segmentation sub-network is employed in our model to extract semantic features that are connected to the disparity branch as the *semantic feature embedding*. Moreover, we propose the warped semantic consistency via *semantic loss regularization*, which further enhances robustness of disparity estimation. Both semantic and disparity evaluation is fully-convolutional so that the proposed *SegStereo* enables end-to-end training.

Our *SegStereo* model with semantic clues embedded benefits both unsupervised and supervised training. In the unsupervised training, both photometric consistency loss and semantic softmax loss are computed and propagated backward. Both the semantic feature embedding and semantic softmax loss can introduce beneficial constraints of semantic consistency. The results evaluated on KITTI Stereo dataset [33] demonstrate the effectiveness of our strategies. We also apply the unsupervised model to CityScapes dataset [10]. It yields better performance than classical SGM method [21]. For the supervised training scheme, we adopt the supervised regression loss instead of unsupervised photometric consistency loss to train the model, which achieves state-of-the-art results on KITTI Stereo benchmark. We further apply the *SegStereo* model to FlyingThings3D dataset [31]. It reaches high accuracy with normal fine-tuning.

Our main contribution and achievement are summarized below.

- We propose a unified framework called *SegStereo* that incorporates semantic segmentation information into disparity estimation pipeline, where semantic consistency becomes an active guidance for disparity estimation.
- The semantic feature embedding strategy and semantic guidance softmax loss help train the system in both unsupervised and supervised manner.
- Our method achieves state-of-the-art results on KITTI Stereo datasets. The results on CityScapes and FlyingThings3D dataset also manifest the effectiveness of our method.

2 Related Work

Supervised Stereo Matching. Traditional methods design local descriptors to compute local matching cost [15,20], followed by some global optimization steps [21]. Zbontar and LeCun [43] are the first to use CNN for matching cost computation. Luo *et al.* [30] designed a siamese network that extracts marginal distributions over all possible disparities for each pixel. Chen *et al.* [8] presented a multi-scale deep embedding model that fuses feature vectors learned within different scale-spaces. Shaked and Wolf [38] proposed a highway network architecture with a hybrid loss that conducts multi-level comparison of image patches.

Inspired by other pixel-wise labeling tasks, the fully-convolution network (FCN) [29] was used to enable end-to-end learning of disparity maps. Mayer *et al.* [31] raised DispNet with a correlation module to encode matching cues instead of picking corresponding pairs from stereo images. Kendall *et al.* [25] proposed the GC-Net framework that combines contextual information by means of 3D convolutions over a cost volume. A three-stage network of Gidaris and Komodakis [16] implements a pipeline to detect, replace, and refine disparity errors respectively. Pang *et al.* [34] presented a cascade network where the second stage learned the residual between initial result and ground-truth values.

Yu *et al.* [42] designed a two-stream network for generation and selection of cost aggregation proposals respectively. Liang *et al.* [28] integrated disparity estimation and refinement into one network. It reaches state-of-the-art performance on KITTI benchmark [33]. Chang and Chen [6] exploited context information for finding correspondence by a pyramid stereo matching network. In contrast, our method concentrates on combining semantic information to improve disparity estimation by semantic feature embedding.

Unsupervised Stereo Matching. In recent years, a number of unsupervised learning methods based on spatial transformation were proposed for view synthesis, depth prediction, optical flow and disparity estimation. Unsupervised methods get rid of the dependence of ground-truth labels, which are always expensive to access. Flynn *et al.* [12] presented an image synthesis network called DeepStereo that learns a cost volume combined with a separate conditional color model. Xie *et al.* [40] designed a Deep3D network that minimizes pixel-wise reconstruction loss to generate right-view images.

Garg *et al.* [13] proposed an end-to-end framework to learn single-view depth by optimizing the projection errors in a calibrated stereo environment. The improved method [17] introduces a fully-differentiable structure and an extra left-right consistency check that leads to better results. A semi-supervised approach was proposed by Kuznietsov *et al.* [27] where supervised and unsupervised alignment loss are used to train the network for depth estimation. Yu *et al.* [23] focused on unsupervised learning of optical flow via photometric constancy and motion smoothness. Meister *et al.* [32] defined a bidirectional census loss to train optical flow. An iterative unsupervised learning network presented by Zhou *et al.* [45] adopts left-right checking to pick suitable matching pairs. Compared with these unsupervised methods, our model applies warping reconstruction to both photometric image and semantic maps, along with additional semantic feature embedding, to reliably estimate disparity.

Semantic-Guided Algorithms. Compared to disparity estimation, semantic segmentation is a high-level classification task where each pixel in the image is assigned to a class [7,29,39,44]. Several methods apply scene parsing information to other tasks. Guney and Geiger [18] leveraged object knowledge in MRF formulation to resolve stereo ambiguity. Bai *et al.* [2] tackled instance-level segmentation and epipolar constraints to reduce the uncertainty of optical flow estimation. A cascaded classification framework of Ren *et al.* [35] iteratively refines semantic masks, stereo correspondence and optical flow fields. Behl *et al.* [4] integrated the instance recognition cues into a CRF-based model for scene flow estimation.

With similar motivation to ours, Cheng *et al.* [9] designed an end-to-end trainable network called SegFlow, which enables joint learning for video object segmentation and optical flow. This model contains a segmentation branch and a flow branch whose feature maps concatenate. We differently focus on disparity estimation, where objects in the scene are captured at the same time so that stable structural information can be exploited. In addition, our *SegStereo* model also propagates softmax loss back to disparity branch by warping, which makes semantic information effective in the whole training process. In addition, our model enables unsupervised learning of disparity with photometric loss and semantic-aware constraints.

3 Our Method

In this section, we describe our *SegStereo* disparity estimation architecture, suitable for both unsupervised and supervised learning. We first present the basic network for disparity regression. Then we detail our incorporation strategies of semantic cues, including semantic feature embedding and semantic loss regularization. Both of them are effective to rectify disparity prediction. Finally, we show how disparity estimation is achieved under unsupervised and supervised conditions.

3.1 Basic Network Architecture

Our overall *SegStereo* network is shown in Fig. 2. The backbone network is ResNet-50 [19]. Instead of directly computing disparity on raw pixels, we adopt

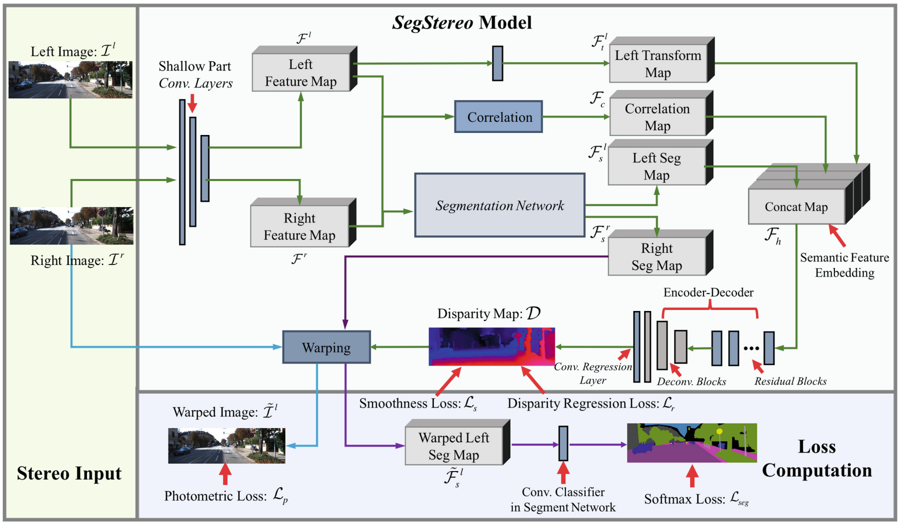

Fig. 2. Our *SegStereo* framework. We extract intermediate features $\mathcal{F}_l$ and $\mathcal{F}_r$ from stereo input. We calculate the cost volume $\mathcal{F}_c$ via the correlation operator. The left segmentation feature map ${\mathcal{F}_s}^l$ is aggregated into disparity branch as *semantic feature embedding*. The right segmentation feature map ${\mathcal{F}_s}^r$ is warped to left view for per-pixel semantic prediction with *softmax loss regularization*. Both steps incorporate semantic information to improve disparity estimation. The *SegStereo* framework enables both unsupervised and supervised learning, using photometric loss $\mathcal{L}_p$ or disparity regression loss $\mathcal{L}_r$.

the shallow part of ResNet-50 model to extract image features $\mathcal{F}^l$ and $\mathcal{F}^r$ on the paired input $\mathcal{I}^l$ and $\mathcal{I}^r$, which is known as robust to local context information encoding.

The cost volume features for stereo matching $\mathcal{F}_c$ are computed by correlation layer between $\mathcal{F}^l$ and $\mathcal{F}^r$, similar to that of DispNetC [31]. To preserve detail information on left stereo feature, we apply a convolution block on $\mathcal{F}^l$ and obtain transformed feature ${\mathcal{F}_t}^l$. Meanwhile, a segmentation network is utilized to compute semantic features ${\mathcal{F}_s}^l$ and ${\mathcal{F}_s}^r$ for left and right images respectively, sharing shallow layer representation with disparity network. The left transformed disparity features ${\mathcal{F}_t}^l$, the correlated features $\mathcal{F}_c$ and the left semantic features ${\mathcal{F}_s}^l$ are concatenated as hybrid feature representation $\mathcal{F}_h$. Here, semantic cues are preliminarily introduced to the disparity network as *Semantic Feature Embedding.*

After feature embedding, we feed $\mathcal{F}_h$ into the disparity encoder-decoder to get full-size disparity map $\mathcal{D}$. The disparity map is further used to warp right semantic feature ${\mathcal{F}_s}^r$ to left under *Semantic Loss Regularization*, detailed in Sect. 3.3. They constitute the key components of our framework. We describe more setting details in Sect. 4.1 and in supplementary material.

3.2 Semantic Feature Embedding

The basic disparity estimation frameworks work well on image patches with edges and corners where clear matching cues are located. It can be optimized with photometric loss in an unsupervised system or guided by supervised ℓ_1 norm regularization otherwise. The remaining major issue is on flat regions, as shown in the first row of Fig. 1. We use semantic cues to help prediction and rectify the final disparity map. As a result, we first incorporate the cues by embedding of semantic feature.

Our semantic feature embedding combines information from left disparity features ${\mathcal{F}_t}^l$, the correlated features $\mathcal{F}_c$ and the left semantic features ${\mathcal{F}_s}^l$. It contains the following advantages. (1) The employed segmentation branch shares the shallow computation with backbone disparity network for efficient computation and effective representation. (2) The semantic feature ${\mathcal{F}_s}^l$ gives more consistent representations on those flat regions compared to the disparity feature ${\mathcal{F}_t}^l$, which introduce object-level prior knowledge. (3) The low-level features and high-level recognition information are fused explicitly via the aggregation of ${\mathcal{F}_t}^l$, $\mathcal{F}_c$ and ${\mathcal{F}_s}^l$. The experiments in Sects. 4.5 and 4.6 further manifest that our semantic embedding helps disparity branch predict more convincing results in both unsupervised and supervised learning. In addition, the right semantic features ${\mathcal{F}_s}^r$ are reserved for the following semantic feature warping and loss regularization.

3.3 Semantic Loss Regularization

The semantic information cues can also guide learning of disparity as a loss term. As shown in Fig. 2, based on the predictive disparity map $\mathcal{D}$, we employ feature warping on the right segmentation map ${\mathcal{F}_s}^r$ to get the reconstructed left segmentation map $\tilde{\mathcal{F}_s}^l$, and use left segmentation ground truth labels as guidance to learn a per-pixel classifier. Finally, the semantic cues guidance loss $\mathcal{L}_{seg}$ is measured between classified warped maps and ground-truth labels.

When training the disparity network, the semantic loss $\mathcal{L}_{seg}$ is propagated back to disparity branch through semantic convolutional classifier and feature warping layer. Along with basic photometric loss $\mathcal{L}_p$ or regression loss $\mathcal{L}_r$, semantic loss $\mathcal{L}_{seg}$ imposes extra object-aware constraints to guide disparity training. The experiments prove that semantic loss regularization can effectively resolve the local disparity ambiguities, especially in the unsupervised learning period.

3.4 Objective Function

The semantic information detailed above can be used in both unsupervised and supervised systems. Here we detail the loss functions in these two conditions.

Unsupervised Manner. One image in a stereo pair can be reconstructed from the other with estimated disparity, which should be close to the original raw input. We utilize this property as photometric consistency to help learn the disparity in an unsupervised manner. Given estimated disparity $\mathcal{D}$, we apply image warping $\varPhi$ on the right image $\mathcal{I}^r$ and get the warped left image reconstruction as $\tilde{\mathcal{I}}^l$. Then we adopt ℓ_1 norm to regularize the photometric consistency with photometric loss $\mathcal{L}_p$ expressed as

$$\mathcal{L}_p = \frac{1}{N}\sum_{i,j} \delta^p_{i,j} \|\tilde{\mathcal{I}}^l_{i,j} - \mathcal{I}^l_{i,j}\|_1, \tag{1}$$

where N is the number of pixels. $\delta^p_{i,j}$ is a mask indicator to avoid outlier as image boarder or occluded regions, where no pixel correspondence exists. If the resulting photometric difference on position (i, j) is greater than a threshold ϵ, $\delta^p_{i,j}$ is 0, otherwise, it is 1.

The photometric consistency enables disparity learning in an unsupervised manner. If there is no regularization term in $\mathcal{L}_p$ to enforce local smoothness of the estimated disparity, local disparity may be incoherent. To remedy this issue, we apply ℓ_1 penalty to disparity gradients $\partial\mathcal{D}$ with the smoothness loss $\mathcal{L}_s$ defined as

$$\mathcal{L}_s = \frac{1}{N}\sum_{i,j}[\rho_s(\mathcal{D}_{i,j} - \mathcal{D}_{i+1,j}) + \rho_s(\mathcal{D}_{i,j} - \mathcal{D}_{i,j+1})], \tag{2}$$

where $\rho_s(\cdot)$ is the spatial smoothness penalty implemented as generalized Charbonnier function [3].

With the semantic feature embedding and semantic loss, the overall loss in our unsupervised system is $\mathcal{L}_{unsup}$, containing the photometric loss $\mathcal{L}_p$, smoothness loss $\mathcal{L}_s$, and the semantic cues loss $\mathcal{L}_{seg}$. We note that disparity labels are not involved in loss computation so that disparity estimation is considered as an unsupervised learning process here. To balance learning of different loss branches, we introduce loss weights λ_p for $\mathcal{L}_p$, λ_s for $\mathcal{L}_s$, and λ_{seg} for $\mathcal{L}_{seg}$. Thus the total loss $\mathcal{L}_{unsup}$ is expressed as

$$\mathcal{L}_{unsup} = \lambda_p \mathcal{L}_p + \lambda_s \mathcal{L}_s + \lambda_{seg} \mathcal{L}_{seg}. \tag{3}$$

Supervised Manner. The proposed semantic cues for disparity prediction also works in supervised training, where the ground truth disparity map $\hat{D}$ is provided. We directly adopt the ℓ_1 norm to regularize prediction where the disparity regression loss $\mathcal{L}_r$ is

$$\mathcal{L}_r = \frac{1}{N_{\mathcal{V}}} \sum_{i,j \in \mathcal{V}} \|\mathcal{D}_{i,j} - \hat{\mathcal{D}}_{i,j}\|_1, \tag{4}$$

where $\mathcal{V}$ is the set of valid disparity pixels in $\hat{\mathcal{D}}$ and $N_{\mathcal{V}}$ is the number of valid pixels. For utilizing the semantic cues, both feature embedding and semantic softmax loss are adopted as described in Sects. 3.2 and 3.3. Loss weight λ_r is used for regression term $\mathcal{L}_r$. The overall loss function $\mathcal{L}_{sup}$ becomes

$$\mathcal{L}_{sup} = \lambda_r \mathcal{L}_r + \lambda_s \mathcal{L}_s + \lambda_{seg} \mathcal{L}_{seg}. \tag{5}$$

4 Experimental Results

In this section, we evaluate key components in the *SegStereo* model. We mainly pretrain the model on CityScapes dataset [10] and evaluate it on KITTI Stereo 2015 dataset [33]. We also compare the performance of our method with other disparity estimation methods on KITTI benchmark [33]. Further, we apply our *SegStereo* model to FlyingThings3D dataset [31] to assess performance on different scenes.

4.1 Model Specification

PSPNet-50 [44] is employed as a segmentation network due to its high performance. The layers (from "conv1_1" to "conv3_1") of PSPNet-50 are used as the shallow part. The extracted features $\mathcal{F}_l$ and $\mathcal{F}_r$ have a 1/8 spatial size to raw images. We select the output of "conv5_4" layer of PSPNet-50 as semantic features. The weights in the shallow part and segmentation network are fixed when training *SegStereo*.

For cost volume computation, we perform 1D-correlation [31] between $\mathcal{F}^l$ and $\mathcal{F}^r$ according to epipolar constraints. Both max displacement and padding size are set to 24 so that the channel number of correlated features $\mathcal{F}_c$ is 25. For left feature transformation, the kernel size of transformed convolutional layer is $1 \times 1 \times 256$. All of $\mathcal{F}_c$, ${\mathcal{F}_t}^l$ and ${\mathcal{F}_s}^l$ have the same spatial size. We directly concatenate them to form the hybrid feature map $\mathcal{F}_h$.

Disparity encoder behind hybrid features $\mathcal{F}_h$ contains 12 residual blocks. Several common convolutional operations in residual blocks are replaced with dilation patterns [44] to integrate wider context information. Disparity decoder consists of 3 deconvolutional blocks and 1 convolutional regression layer to output full-size disparity map. We provide more details in supplementary material.

The right segmentation map ${\mathcal{F}_s}^r$ is of 1/8 size to the raw image, while the estimated disparity map D is of full size. To perform feature warping, we first upsample ${\mathcal{F}_s}^r$ to the full size. We afterwards downsample warped feature map to 1/8 size and get the final reconstructed left segmentation feature map as $\tilde{\mathcal{F}_s}^l$.

4.2 Baseline Model Excluding Semantic Information

To validate the effect of incorporating semantic cues, we design a baseline model called *ResNetCorr* without any semantic information. The hybrid features $\mathcal{F}_h$ in *ResNetCorr* is concatenated with the correlated features $\mathcal{F}_c$ and left transformed features ${\mathcal{F}_t}^l$. The rest encoder-decoders are attached behind $\mathcal{F}_h$, as that of *SegStereo*. The softmax L_{seg} term is excluded in loss computation. We provide the structural definition of *ResNetCorr* model in supplementary material.

4.3 Datasets and Evaluation Metrics

The CityScapes dataset [10] is released for urban scene understanding. It provides rectified stereo image pairs and corresponding disparity maps precomputed by SGM algorithm [21]. It contains 5,000 high quality pixel-level finely annotated maps for left-view. These images are split into sets with numbers $2,975$, 500 and $1,525$ for training, validation and testing. In addition, this dataset provides $19,997$ stereo images and their SGM labels in extra training set. We will use these extra data for model pretraining.

The KITTI Stereo 2015 dataset [33] contains 200 training and 200 testing image pairs. The 200 training images also has semantic labels [1]. We mainly use the dataset for fine tuning and evaluation. The KITTI Stereo 2012 dataset [14] also provides disparity maps, which contain 194 training and 195 testing image pairs.

The FlyingThings3D dataset [31] is a virtual dataset for scene matching including optical flow estimation and disparity prediction. This dataset is rendered by computer graphics techniques with background objects and 3D models. It provides 22,390 images for training and 4,370 images for testing.

To evaluate the results, we apply the end-point-error (EPE), which measures the average pixel deviation and the bad pixel error (D1). The latter calculates the percentage of disparity errors below a threshold. Both the errors in non-occluded region (Noc) and all pixels (All) are evaluated.

4.4 Implementation Details

Our implementation of the *SegStereo* model is based on a customized Caffe [24]. We use the "poly" learning rate policy where current learning rate equals to the base one multiplying $(1 - \frac{iter}{max\_iter})^{power}$. Such learning policy is also adopted in [5,44] for better performance. When training on CityScapes dataset, we set base learning rate to 0.01, power to 0.9. Momentum and weight decay are set to 0.9 and 0.0001, respectively. These parameters of learning policy are kept on supervised fine-tuning process.

For data augmentation, we adopt random resizing, color shift and contrast brightness adjustment. The random factor is between 0.5 to 2.0. The maximum color shift along RGB axes is set to 10 and the maximum brightness shift is set to 5. The contrast multiplier is between 0.8 and 1.2. The "cropsize" is set to 513 × 513 and batch size is set to 16.

In unsupervised training, the loss weights λ_p, λ_s and λ_{seg} for photometric, softmax and smoothness terms are set to 1.0, 10.0, 0.1, respectively. The threshold ϵ in photometric loss is set to 10. When switching to supervised training, if providing semantic labels, the loss weights for λ_r, λ_s and λ_{seg} for regression, softmax and smoothness term are set to 1.0, 1.0, 0.1. If no semantic labels are provided, the loss weight of softmax term is set to 0. The Charbonnier parameters α, β and ϵ in smoothness loss term are 0.21, 5.0 and 0.001 as described in [23].

Table 1. Results of unsupervised training models on KITTI Stereo 2015 [33].

Model	Noc pixels		All pixels	
	EPE	D1 error	EPE	D1 error
1. Evaluation of semantic feature embedding				
ResCorr (photometric loss)	2.46	12.78	3.36	14.08
SegStereo (photometric loss)	**1.98**	**10.76**	**2.72**	**12.08**
ResCorr (photometric loss + smooth loss)	2.13	11.05	2.43	12.16
SegStereo (photometric loss + smooth loss)	**1.87**	**9.39**	**2.17**	**10.53**
2. Evaluation of softmax loss regularization				
SegStereo (photometric)	1.98	10.76	2.72	12.08
SegStereo (photometric + smooth)	1.87	9.39	2.17	10.53
SegStereo (photometric + smooth + softmax)	**1.61**	**8.95**	**1.89**	**10.03**
3. Comparison to other unsupervised methods				
Zhou [45]	–	8.61	–	9.91
Godard [17]	–	–	–	9.19
SegStereo (pretrain on Cityscapes dataset)	1.61	8.95	1.89	10.03
SegStereo (ft on KITTI Stereo dataset)	**1.46**	**7.70**	**1.84**	**8.79**

4.5 Unsupervised Learning

Semantic Feature Embedding. The first experiment in Table 1 compares the errors between *ResNetCorr* and *SegStereo* models. We observe that with semantic features from PSPNet-50, *EPE* is improved by 20% and the *D1 error* is reduced by 15% when only adopting photometric loss. When combining the photometric and smoothness losses to train the models, *EPE* is improved by 12% and the *D1 error* is reduced by 13%. It shows that semantic feature embedding significantly reduces the disparity errors.

Softmax Loss Regularization. The second experiment in Table 1 is to validate the effect of softmax loss regularization. Based on photometric loss, we use smoothness loss to penalize discontinuity on disparity maps, which reduces *EPE* from 2.72 to 2.17 and the D1 error from 12.08 to 10.53 on all pixels. With additional softmax loss to constrain semantic consistency, *EPE* decreases from 2.17 to 1.89 and the *D1 error* decreases from 10.53 to 10.03. Thus, the regularization of softmax loss reduces *EPE* by 13% and the *D1 error* by 5%, respectively.

Figure 3 shows results of different loss combinations (with or without softmax loss). We observe that the gain of softmax loss mainly arises on big objects, such as road and car, which directly help enhance disparity prediction on local ambiguous regions.

Finetune on KITTI Stereo Dataset. We compare our approach with other unsupervised methods in the third experiment as listed in Table 1. To adapt our model to KITTI dataset, we finely tune *SegStereo* on the 200 images of KITTI 2015 training set. We set the maximum iteration number to 500 and batch size to 16, so that 40 epochs are conducted. All photometric loss, smoothness loss

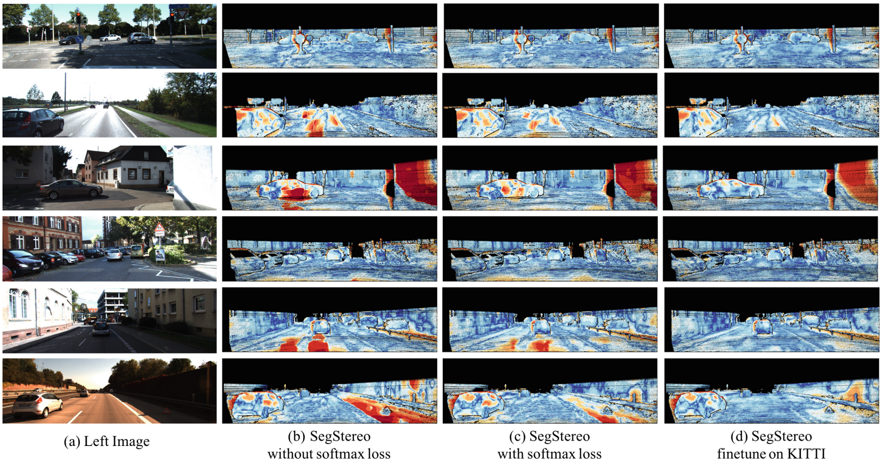
(a) Left Image
(b) SegStereo without softmax loss
(c) SegStereo with softmax loss
(d) SegStereo finetune on KITTI

Fig. 3. Qualitative examples of unsupervised *SegStereo* models on KITTI Stereo 2015 dataset [33]. With the guidance of softmax regularization and additional fine-tune process, the accuracy of disparity is improved.

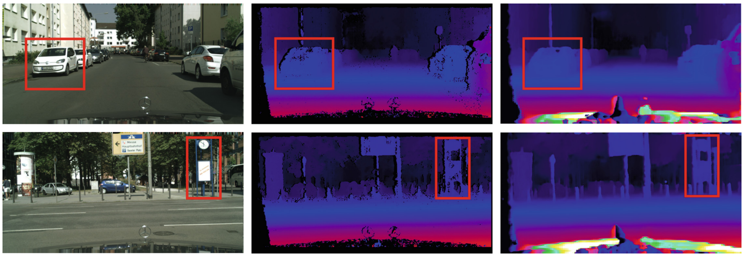

Fig. 4. Qualitative examples of unsupervised-learning version of the *SegStereo* model on CityScapes validation set [10]. From left to right: left input images, disparity maps predicted by SGM algorithm [21], and our disparity maps.

and softmax loss are used in this process. Qualitative results in Fig. 3 show that prediction errors are further reduced by fine-tuning. Our model outperforms the other two unsupervised methods [17,45] on KITTI 2015 benchmark.

CityScapes Results. We adapt the unsupervised *SegStereo* model to CityScapes dataset [10]. In Fig. 4, we give several examples to visualize quality on the validation set. Compared to the results of SGM algorithm [21], our method yields better structures in term of global scene information and details of objects.

4.6 Supervised Learning

KITTI Results. In supervised learning, the ground-truth disparity maps are directly applied to train our *SegStereo* model. As KITTI stereo dataset is too

Table 2. Results of supervised-training models evaluated on KITTI Stereo 2015 [33]

Model	Train				Test			
	EPE		D1		EPE		D1	
	Noc	All	Noc	All	Noc	All	Noc	All
1. Pretrained on Cityscapes dataset								
ResNetCorr	–	–	–	–	1.43	1.46	7.33	7.64
SegStereo	–	–	–	–	**1.39**	**1.41**	**7.01**	**7.34**
2. Pretrained on Cityscapes extra set and FlyingThings3D dataset								
ResNetCorr	–	–	–	–	1.19	1.21	5.46	5.64
SegStereo	–	–	–	–	**1.15**	**1.17**	**5.20**	**5.38**
3. Finetune on KITTI stereo 2012 and 2015 dataset								
ResNetCorr	0.40	0.41	0.68	0.76	0.73	0.76	2.13	2.40
SegStereo	0.40	0.41	0.65	0.70	0.73	0.75	2.11	2.30
SegStereo (corr13)	0.39	0.40	0.65	0.70	**0.66**	**0.70**	**1.96**	**2.25**

small, we pre-train our model on CityScapes dataset. Although the disparity maps computed by SGM algorithm contain errors and holes, they are useful for our model to get reasonable accuracy. The maximum iteration is set to $90K$. Different from unsupervised training, here the disparity regression loss $\mathcal{L}_r$ plays the major role. We also compare the performance between *ResNetCorr* and *SegStereo*. The first experiment in Table 2 shows that disparity error rate is slightly reduced by semantic feature embedding when we pretrain the models on CityScapes dataset [10].

In the second experiment, we fuse the extra training set in CityScapes and training set in FlyingThings3D dataset to pretrain *ResNetCorr* and *SegStereo*. Since there is no semantic labels in such two datasets, we do not compute softmax loss. The weights in segmentation branch of *SegStereo* is pretrained on CityScapes training set and fixed. We extend the maximum iterations to $500K$. Compared to the first experiment, with more training data, both *ResNetCorr* and *SegStereo* achieve higher accuracy. And the performance of *SegStereo* is still better than *ResNetCorr*.

Table 3. Comparison with other disparity estimation methods on the test set of KITTI 2015 [33]. Our method achieves state-of-the-art results on this benchmark.

Methods	Noc			All			Runtime (s)
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all	
SPS-st [41]	3.50	11.61	4.84	3.84	12.67	5.31	2
Content-CNN [30]	3.32	7.44	4.00	3.73	8.58	4.54	1
DispNetC [31]	4.11	3.72	4.05	4.32	4.41	4.34	**0.06**
MC-CNN [43]	2.48	7.64	3.33	2.89	8.88	3.89	67
PBCP [37]	2.27	7.71	3.17	2.58	8.74	3.61	68
Displets v2 [18]	2.73	4.95	3.09	3.00	5.56	3.43	265
L-ResMatch [38]	2.35	5.74	2.91	2.72	6.95	3.42	48
DRR [16]	2.34	4.87	2.76	2.58	6.04	3.16	0.4
GC-NET [25]	2.02	5.58	2.61	2.21	6.16	2.87	0.9
CRL [34]	2.32	3.12	2.45	2.48	3.59	2.67	0.47
DeepStereo [42]	2.06	5.32	2.32	2.17	5.46	2.79	1.13
iResNet [28]	2.07	**2.76**	2.19	2.25	**3.40**	2.44	0.12
PSMNet [6]	**1.71**	4.31	2.14	**1.86**	4.62	2.32	0.41
SegStereo (Ours)	1.76	3.70	**2.08**	1.88	4.07	**2.25**	0.6

In the third experiment, we use KITTI Stereo 2012 and 2015 datasets to finely tune our pretrained models from the second experiment. We set the maximum iteration to 90K and base learning rate to 0.01. To facilitate performance comparison, we split Stereo 2015 training set [30] so that 40 images are randomly selected for validation and the remaining 160 images are used for training. Table 2 lists errors on both training and validation sets. Compared to the *ResNetCorr* model, semantic feature embedding prevents overfitting and brings a certain improvement on disparity estimates.

To exploit more detailed matching cues on fine scales, we redesign *SegStereo*, where the shallow part is end with the "conv1_3" layer of PSPNet-50. To adapt to the increased feature map size, the maximum displacement and padding size of the correlation layer are both set to 96. We also up-sample the semantic feature maps from "conv5_4" layer for semantic feature embedding. This redesigned model is also pretrained on the fusion set of CityScapes and FlyingThings3D, followed by fine-tuning on KITTI Stereo dataset. The new *SegStereo* model (with remark "corr13" in Table 2) outperforms general *SegStereo* by leveraging more detail information.

Table 3 compares our model to other approaches on KITTI 2015 benchmark [33]. Our method achieves state-of-the-art results. Figure 5 gives several visual examples on KITTI 2015 test set. By incorporating semantic information, our *SegStereo* model is able to handle challenging scenarios. In supplementary material, we also provide results on KITTI 2012 benchmark [14] and segmentation results.

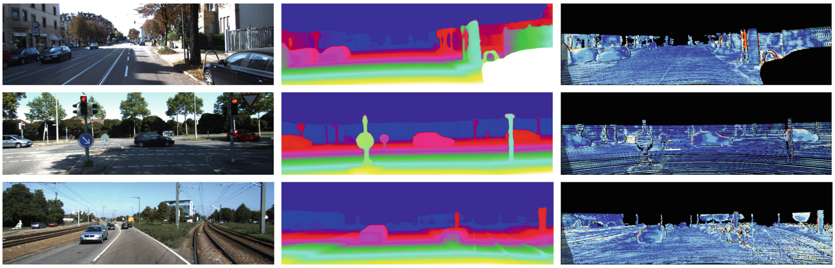

Fig. 5. Supervised-learning results on KITTI Stereo 2015 test sets [33]. By incorporating semantic information, our method is able to estimate accurate disparity. From left to right, we show left input images, disparity predictions of *SegStereo*, and error maps

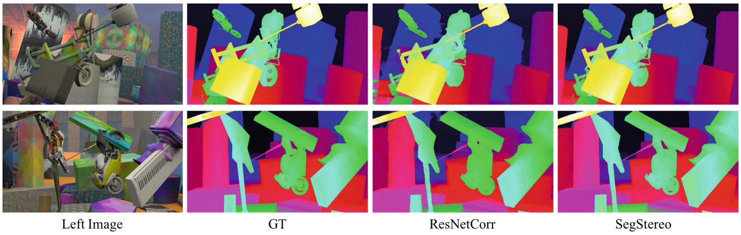

Fig. 6. Qualitative examples of *ResNetCorr* and *SegStereo* model on FlyingThings3D validation set [31]. From left to right, left images, ground-truth, *ResNetCorr* results and *SegStereo* results

Table 4. Comparison with other disparity estimation methods on the test set of FlyingThings3D [31].

Model	SGM [21]	DispNetC [31]	GC-Net [25]	CRL [34]	iResNet [28]	**ResNetCorr**	**SegStereo**
EPE	7.29	2.33	1.84	1.67	**1.27**	3.50	1.45
D1	16.18	10.04	9.67	6.70	4.90	8.45	**3.50**

FlyingThings3D Results. To illustrate that our *SegStereo* model can adapt to other datasets, we test the supervised-training *ResNetCorr* and *SegStereo* on FlyingThings3D dataset [31]. Here, we directly select the pretrained models from the second experiments of Table 2. The two models are compared with other methods on the validation set of FlyingThings3D in Table 4. With the guidance of semantic information, the *SegStereo* model outperforms *ResNetCorr* and becomes state-of-the-art, which indicates that segmentation modules is effective and general for disparity estimation across various datasets. Figure 6 shows several visual examples on validation set.

5 Conclusion

In this paper, we have proposed a unified model *SegStereo*, which integrates semantic feature maps into disparity prediction pipeline. A softmax loss is combined with common photometric loss or disparity regression loss to enable training in both unsupervised and supervised manners. Our *SegStereo* leads to more reliable results, especially on ambiguous areas. Experiments on KITTI Stereo datasets demonstrate the effectiveness of the semantic-guided strategy. Our method achieves state-of-the-art performance on this benchmark. Results on CityScapes and FlyingThings3D datasets further manifest its adaptability.

Acknowledgment. This work was supported in part by the National Key R&D Program of China under Grant No. 2017YFB1302200 and by Joint Fund of NORINCO Group of China for Advanced Research under Grant No. 6141B010318.

References

1. Alhaija, H.A., Mustikovela, S.K., Mescheder, L., Geiger, A., Rother, C.: Augmented reality meets deep learning for car instance segmentation in urban scenes. In: BMVC (2017)
2. Bai, M., Luo, W., Kundu, K., Urtasun, R.: Exploiting semantic information and deep matching for optical flow. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 154–170. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_10
3. Barron, J.T.: A more general robust loss function. arXiv preprint arXiv:1701.03077 (2017)
4. Behl, A., Jafari, O.H., Mustikovela, S.K., Alhaija, H.A., Rother, C., Geiger, A.: Bounding boxes, segmentations and object coordinates: how important is recognition for 3D scene flow estimation in autonomous driving scenarios? In: ICCV (2017)

5. Brown, M., Hua, G., Winder, S.: Discriminative learning of local image descriptors. TPAMI **33**(1), 43–57 (2011)
6. Chang, J., Chen, Y.: Pyramid stereo matching network. In: CVPR (2018)
7. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs (2015)
8. Chen, Z., Sun, X., Wang, L., Yu, Y., Huang, C.: A deep visual correspondence embedding model for stereo matching costs. In: ICCV (2015)
9. Cheng, J., Tsai, Y.H., Wang, S., Yang, M.H.: SegFlow: joint learning for video object segmentation and optical flow. In: ICCV (2017)
10. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: CVPR (2016)
11. Dosovitskiy, A., et al.: FlowNet: learning optical flow with convolutional networks. In: ICCV (2015)
12. Flynn, J., Neulander, I., Philbin, J., Snavely, N.: DeepStereo: learning to predict new views from the world's imagery. In: CVPR (2016)
13. Garg, R., Vijay Kumar, B.G., 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
14. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The kitti vision benchmark suite. In: CVPR (2012)
15. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010. LNCS, vol. 6492, pp. 25–38. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19315-6_3
16. Gidaris, S., Komodakis, N.: Detect, replace, refine: deep structured prediction for pixel wise labeling. In: CVPR (2017)
17. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: CVPR (2017)
18. Guney, F., Geiger, A.: Displets: resolving stereo ambiguities using object knowledge. In: CVPR (2015)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
20. Heise, P., Jensen, B., Klose, S., Knoll, A.: Fast dense stereo correspondences by binary locality sensitive hashing. In: ICRA (2015)
21. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. TPAMI **30**(2), 328–341 (2008)
22. Hirschmuller, H., Scharstein, D.: Evaluation of stereo matching costs on images with radiometric differences. TPAMI **31**(9), 1582–1599 (2009)
23. Yu, J.J., Harley, A.W., Derpanis, K.G.: Back to basics: unsupervised learning of optical flow via brightness constancy and motion smoothness. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9915, pp. 3–10. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49409-8_1
24. Jia, Y., et al.: Caffe: convolutional architecture for fast feature embedding. In: ACM MM (2014)
25. Kendall, A., et al.: End-to-end learning of geometry and context for deep stereo regression. In: ICCV (2017)
26. Kong, D., Tao, H.: A method for learning matching errors for stereo computation. In: BMVC (2004)
27. Kuznietsov, Y., Stückler, J., Leibe, B.: Semi-supervised deep learning for monocular depth map prediction. In: CVPR (2017)

28. Liang, Z., et al.: Learning for disparity estimation through feature constancy. In: CVPR (2018)
29. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
30. Luo, W., Schwing, A.G., Urtasun, R.: Efficient deep learning for stereo matching. In: CVPR (2016)
31. Mayer, N., et al.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: CVPR (2016)
32. Meister, S., Hur, J., Roth, S.: UnFlow: unsupervised learning of optical flow with a bidirectional census loss. In: AAAI (2018)
33. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: CVPR (2015)
34. Pang, J., Sun, W., Ren, J., Yang, C., Yan, Q.: Cascade residual learning: a two-stage convolutional neural network for stereo matching. In: ICCV Workshop (2017)
35. Ren, Z., Sun, D., Kautz, J., Sudderth, E.B.: Cascaded scene flow prediction using semantic segmentation. In: ICCV Workshop (2017)
36. Revaud, J., Weinzaepfel, P., Harchaoui, Z., Schmid, C.: DeepMatching: hierarchical deformable dense matching. IJCV **120**(3), 300–323 (2016)
37. Seki, A., Pollefeys, M.: Patch based confidence prediction for dense disparity map. In: BMVC (2016)
38. Shaked, A., Wolf, L.: Improved stereo matching with constant highway networks and reflective confidence learning. In: CVPR (2017)
39. Vijay, B., Alex, K., Cipolla, R.: SegNet: a deep convolutional encoder-decoder architecture for image segmentation. TPAMI **39**(12), 2481–2495 (2017)
40. Xie, J., Girshick, R., Farhadi, A.: Deep3D: fully automatic 2D-to-3D video conversion with deep convolutional neural networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 842–857. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_51
41. Yamaguchi, K., McAllester, D., Urtasun, R.: Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 756–771. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_49
42. Yu, L., Wang, Y., Wu, Y., Jia, Y.: Deep stereo matching with explicit cost aggregation sub-architecture. In: AAAI (2018)
43. Zbontar, J., LeCun, Y.: Stereo matching by training a convolutional neural network to compare image patches. JMLR **17**(1), 2287–2318 (2016)
44. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017)
45. Zhou, C., Zhang, H., Shen, X., Jia, J.: Unsupervised learning of stereo matching. In: ICCV (2017)