UNSUPERVISED STEREO MATCHING USING CORRESPONDENCE CONSISTENCY

Sunghun Joung *Seungryong Kim* *Bumsub Ham* *Kwanghoon Sohn*

School of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea
E-mail: khsohn@yonsei.ac.kr

ABSTRACT

Deep convolutional neural networks (CNNs) have shown revolutionary performance improvements for matching cost computation in stereo matching. However, conventional CNN-based approaches to learn the network in a supervised manner require a large number of ground-truth disparity maps, which limits their applicability. To overcome this limitation, we present a novel framework to learn a CNNs architecture for matching cost computation in an unsupervised manner. Our method leverages an image domain learning combined with stereo epipolar constraints. Exploiting the correspondence consistency between stereo images as supervision, our method selects the training samples in each iteration during network training and uses them to learn the network. To boost the performance, we also propose a multi-scale cost computation scheme. Experimental results show that our method outperforms the state-of-the-art methods including even supervised learning based methods on various benchmarks.

Index Terms— stereo matching, matching cost, similarity learning, unsupervised learning, convolutional neural networks

1. INTRODUCTION

Establishing dense correspondence fields across stereo images is essential for numerous tasks such as stereo reconstruction, autonomous driving, robotics, intermediate view generation, and 3D scene reconstruction [1, 2, 3]. To estimate reliable correspondences, the matching cost computation to measure the dissimilarity between stereo images is one of the most important steps in stereo matching [1].

Conventionally, hand-crafted features were mainly used to define the matching cost [4, 5, 6, 7]. Since they have provided a limited performance due to the lack of robustness, most methods have attempted to refine the estimated disparity using a powerful optimizer or post processing scheme [8]. Recently, by leveraging convolutional neural networks (CNNs) [3, 9, 10], many approaches have reformulated the matching cost function in a learning framework, where the network is learned to estimate a reliable disparity map. As a pioneering work, Zbontar *et al.* proposed a MC-CNN architecture [3] to discriminate the positive sample pairs from a large number of training samples, thus enabling robust matching cost computation in stereo matching. It showed a highly improved performance compared to conventional hand-crafted methods [4].

To provide satisfactory performances, these supervised learning approaches require a large number of ground-truth disparity maps. However, constructing the large collection of ground truths is a very challenging work even with equipment such as structured light [11] and LiDAR [12]. Thus, currently, there exist few stereo datasets that have a limited number of disparity maps [11, 12]. To overcome this limitation, unsupervised learning approaches to monocular depth estimation have been proposed where they do not require ground truth disparity maps to train the network [13, 14]. However, they still show a limited performance compared to supervised learning approaches [3, 9, 10], and only show better performance than existing monocular depth estimation methods [15, 16, 17].

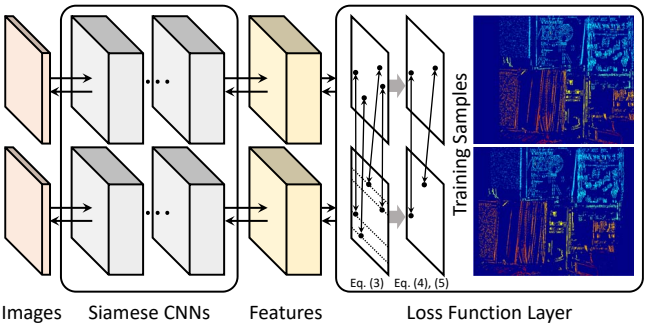


Fig. 1. Our unsupervised learning framework with siamese CNNs. To train siamese CNNs, we leverage an image domain learning combined with stereo epipolar constraints. At the loss layer, it selects the training samples using a correspondence consistency and discriminant sample mining, and the gradients of loss for these samples are back-propagated to learn the network.

In this paper, we propose a novel framework to train the CNN architecture that computes the matching cost for stereo matching in an unsupervised manner, as shown in Fig. 1. Unlike unsupervised monocular depth estimation methods [13, 14], we formulate the problem as a feature matching task, where the network is used to build a feature to compute the matching cost explicitly. Our key-ingredient is to use an image domain learning combined with stereo epipolar constraint. We leverage the inherent correspondence consistency between images as supervision for the training. Our approach selects training samples in each iteration during network training, and exploits those samples to learn the network. Furthermore, we propose a training sample mining scheme to boost the training performance and convergence rate. In the testing procedure, we propose a multi-scale cost computation to boost the matching performance. Experimental results show that our unsupervised learning based framework shows the state-of-the-art performance even against the supervised learning based method [3] on various benchmarks such as the Middlebury benchmark [11] and KITTI benchmark [12].

2. PROPOSED METHOD

2.1. Problem Formulation and Overview

Let us define a rectified pair of stereo images I, I' such that $I(i), I'(i) : \mathcal{I} \rightarrow \mathbb{R}^3$ for the pixel $i = [i_{\mathbf{x}}, i_{\mathbf{y}}]^T$, where $\mathcal{I} \subset \mathbb{N}^2$ is a discrete image domain. Given stereo images I, I', stereo matching aims at estimating a disparity map $D(i)$ for each pixel i by establish-

ing correspondences, satisfying $I(i_{\mathbf{x}}, i_{\mathbf{y}}) = I'(i_{\mathbf{x}} - D(i), i_{\mathbf{y}})$. To estimate the disparity among disparity candidates $d = \{1, ..., d_{\text{max}}\}$, where d_{max} is the maximum disparity range, the matching cost is first measured between $I(i_{\mathbf{x}}, i_{\mathbf{y}})$ and $I'(i_{\mathbf{x}} - d, i_{\mathbf{y}})$, and then determined by a winner-takes-all (WTA) strategy.

To measure the similarity between pixels, CNNs based feature vectors have shown a satisfactory performance [3, 9, 10]. Generally, it formulates the matching cost function in a learning framework in a way that feature vector through feed-forward process on an image $\mathcal{F}(I; \mathbf{W})$, where $\mathbf{W}$ is a network parameter, is used to measure the matching cost. These methods train the CNNs that make the positive samples to be more similar and the negative samples to be more dissimilar in the feature space of $\mathcal{F}(I; \mathbf{W})$. However, it still contains the limitation with the lack of ground-truth disparity map, and it is very challenging work to construct the large collection.

To overcome the above problem, we propose a method to learn the matching cost network without a supervision of ground truth disparity map. Unlike conventional methods [3, 9, 10] using patch-level training samples, our approach formulates the image domain learning framework as in [18]. It estimates the tentative positive samples by exploiting the inherent correspondence consistency between stereo images using epipolar geometry constraints. Specifically, to estimate the tentative positive samples in each iteration of network training, convolutional activation $\mathbf{A}(i) = \mathcal{F}(I(i); \mathbf{W})$ is first used as a feature to find the initial correspondences, and positive samples are then selected through correspondence consistency. With these positive samples in each iteration, the network parameter is gradually learned with an evolving iteration. Since even initial random parameter can provide enough number of positive samples, our learning framework guarantees convergence.

2.2. Positive Set Sampling via Correspondence Consistency

In this section, we introduce the framework for positive set sampling between stereo images in an unsupervised manner, followed by discriminant training sample mining.

2.2.1. Initial disparity map computation

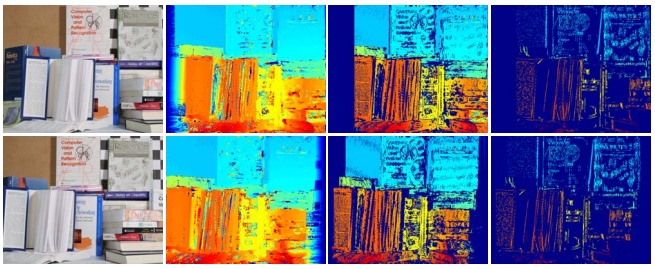
To leverage the correspondence consistency between stereo images, initial disparity maps of stereo images should be built. To this end, the matching cost function $C(i, d)$ is first measured as the dissimilarity between convolutional activations of stereo images such that $\mathbf{A}(i) = \mathcal{F}(I(i); \mathbf{W})$ and $\mathbf{A}'(i) = \mathcal{F}(I'(i); \mathbf{W})$, followed by a simple L_2 distance:

$$C(i, d) = \left\| \mathbf{A}(i_{\mathbf{x}}, i_{\mathbf{y}}) - \mathbf{A}'(i_{\mathbf{x}} - d, i_{\mathbf{y}}) \right\|^2, \tag{1}$$

where it is defined for all pixel $i \in \mathcal{I}$ and all disparity candidates $d = \{1, ..., d_{\text{max}}\}$. With these cost volume C, the disparity map is then estimated by finding the minimum matching cost across disparity search ranges in a WTA manner such that

$$D(i) = \text{argmin}_d \, C(i, d). \tag{2}$$

By computing the initial disparity maps $D(i)$ and $D'(i)$ from left image $I(i)$ and right image $I'(i)$, respectively, the positive samples can be selected though correspondence consistency, which will be described in the following section. It should be noted that conventionally, hand-crafted kernel-based methods [4, 5, 6] have shown an acceptable performance to provide the initial disparity map. Since convolutional activation $\mathbf{A} = \mathcal{F}(I; \mathbf{W})$ also can be considered as the feature through multiple kernel convolutions, convolutional activations $\mathbf{A}$ enable us to estimate enough number of positive samples, even with Gaussian random values of $\mathbf{W}$ in an initialization. Fig. 2(b) shows the initial disparity map estimated with random parameters, which shows an acceptable performance. We will show qualitative evaluation of our network with randomly initialized parameter in Sec. 3. By evolving the iteration during training, the network parameter is gradually learned to estimate the reliable disparity map.



(a) (b) (c) (d)

Fig. 2. Visualizations of positive set sampling using correspondence consistency during network training: (a) input left and right image, (b) estimated initial disparity maps, positive sample sets through (c) correspondence consistency check only, and (d) correspondence consistency check with discriminant training sample mining.

2.2.2. Correspondence consistency check

In stereo matching, most existing methods inherently assume that the pixels in left image have at most one matched pixel in right image across epipolar line [1]. Thus, the correspondence match from the left image to the right image should be consistent with that from the right image to the left image. Conventionally, such a correspondence consistency is popularly used to eliminate the erroneous disparities in post processing step [3, 19]. Unlike this, in our approach, we incorporate this correspondence consistency into loss function, where the tentative positive samples are determined in each iteration of training. Since the reliability of positive samples after the correspondence consistency is proved [20, 21, 18], these samples can be helpful cues to train the network in an unsupervised manner.

Specifically, with estimated disparity maps D and D' for left and right images, we can determine the positive samples that satisfy the following condition:

$$\left\| D(i_{\mathbf{x}}, i_{\mathbf{y}}) - D'(i_{\mathbf{x}} - D(i), i_{\mathbf{y}}) \right\|^2 \leq t, \tag{3}$$

where t is a threshold parameter. We then discard inconsistent pixels. Fig. 2(c) shows determined positive samples.

2.2.3. Discriminant training sample mining

The positive samples through the correspondence consistency might be distributed in all image domain. However, when training the network for measuring the dissimilarity, most positive samples on homogeneous regions cannot contribute the performance boosting. Furthermore, erroneous positives rather degrade the performance. To alleviate these limitations, we employ the training sample mining scheme, where the color and gradient constraints are used to eliminate the positive samples on homogeneous or erroneous regions and further *hard* positive samples are determined according to their matching cost.

First of all, to ensure the positive samples as reliably matched pixels, a color similarity constraint is used with an assumption that the matched pixels have similar color values such that

$$\left\| I(i_{\mathbf{x}}, i_{\mathbf{y}}) - I'(i_{\mathbf{x}} - D(i), i_{\mathbf{y}}) \right\|^2 \leq c, \tag{4}$$

where c is a threshold parameter.

Secondly, to prevent the positive samples to be selected on homogeneous regions, we simply eliminate the samples with small gradient such that

$$\|\nabla_{\mathbf{x}} I(i_{\mathbf{x}}, i_{\mathbf{y}})\| \leq g, \tag{5}$$

where $\nabla\mathbf{x}$ is a differential operator defined in the $\mathbf{x}$-direction, and g is a threshold parameter. Since a disparity map is estimated across search ranges in the $\mathbf{x}$-direction, a large gradient in the $\mathbf{x}$-direction can be a reliable cue to select structural distinctive positive samples.

Finally, among training samples, *hard* positive samples are determined, which can boost the performance and convergence rate. Specifically, hard positive samples are determined from the samples that have high matching costs, which are hard to be estimated as positive samples. After these training sample mining steps, the final training sample set such that Ω are determined through feed-forward process, as shown in Fig. 2(d).

2.3. Network Architecture

In this paper, we exploit a fast version of the network architecture as in [3] which has shown the state-of-the-art performance. It consists of 4 or 5 convolutional layers depending on datasets, followed by rectified linear units (ReLUs) except for the last convolutional layer. Furthermore, a channel-wise L_2 normalization layer is used as the last layer. For all convolutional layers, depth of convolutional kernel is 64 and convolution kernel size is 3.

The loss layer contains all procedure of correspondence consistency check, color and gradient constraint based mining, and hard positive mining. With determined training samples, the two kinds of loss function can be employed, correspondence contrastive loss and correspondence cross-entropy loss. Correspondence contrastive loss is to train the network by minimizing the regression loss [3, 22, 18], which makes the positive samples to be more similar and the negative samples to be more dissimilar. Correspondence cross-entropy loss [10] is to train the network in a way that the positive samples are classified properly among all disparity candidates.

2.3.1. *Correspondence contrastive loss*

Since our method is defined in the image domain, the loss function is also defined in the image domain. For training the network with stereo image pairs, the correspondence contrastive loss is defined as

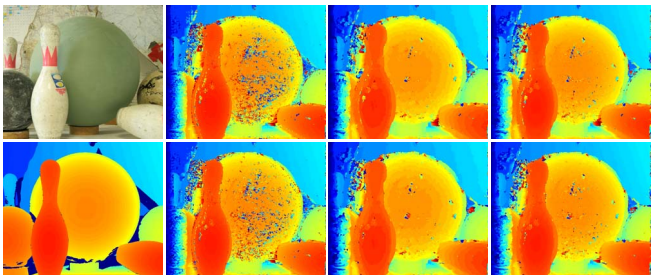
$$\mathcal{L}(\mathbf{W}) = \frac{1}{2N}\sum\nolimits_{i\in\Omega} l(i)\|\mathcal{F}(I(i);\mathbf{W}) - \mathcal{F}(I'(i');\mathbf{W})\|^2 + (1-l(i))\max(0, M - \|\mathcal{F}(I(i);\mathbf{W}) - \mathcal{F}(I'(i');\mathbf{W})\|^2), \tag{6}$$

where $l(i)$ denotes a class label that is 1 for a positive pair and 0 otherwise. N is the number of training samples. M is the maximal cost. Following [3], the negative samples are obtained by shifting the positive samples with some margin in a $\mathbf{x}$-direction such that $i'_{\mathbf{x}} = i_{\mathbf{x}} - D(i) + o_i$ where o_i is chosen from the interval of allowed negative offset.

2.3.2. *Correspondence cross-entropy loss*

We further propose correspondence cross-entropy loss that allows to computing a softmax loss for each pixel across all possible disparities. Specifically, for each pixel i and its possible disparity candidates, correspondence cross-entropy loss is defined such that

$$\mathcal{L}(\mathbf{W}) = -\frac{1}{2N}\sum\nolimits_{i\in\Omega}\sum\nolimits_{k} P_{\mathrm{T}}(k;i)\log(P(k;i)), \tag{7}$$



(a) (b) (c) (d)

Fig. 3. Qualitative evaluations of multi-scale cost computation using two loss functions: (a) left input image and ground truth disparity map, estimated disparity maps using (top) correspondence contrastive loss and (bottom) correspondence cross-entropy loss with (b) $s = 1$ (Error: 17.26% and 15.63%), (c) $s = 1/2$ (Error: 12.92% and 12.49%), and (d) $s \in \{1, 1/2\}$ (Error: 11.89% and 11.05%).

where k is defined for all possible disparities such that $k_{\mathbf{x}} = \{i_{\mathbf{x}} - 1, ..., i_{\mathbf{x}} - d_{\max}\}$. $P_T(k;i)$ is a label, which is defined as 1 if $k_{\mathbf{x}} = i_{\mathbf{x}} - D(i)$, and 0 otherwise. Furthermore, $P(k;i)$ is a softmax probability defined such that

$$P(k;i) = \frac{\exp(U - \|\mathcal{F}(I(i);\mathbf{W}) - \mathcal{F}(I'(k);\mathbf{W})\|^2)}{\sum_l \exp(U - \|\mathcal{F}(I(i);\mathbf{W}) - \mathcal{F}(I'(l);\mathbf{W})\|^2)}, \tag{8}$$

where l is defined for all possible disparities such that $l_{\mathbf{x}} = \{i_{\mathbf{x}} - 1, ..., i_{\mathbf{x}} - d_{\max}\}$ similar to k. We convert the matching cost to the similarity score with the constant U, which is set as 1 empirically.

2.4. Multi-scale Cost Computation

Given stereo images, we compute the matching cost by measuring the similarity between $\mathbf{A}_i = \mathcal{F}(I(i);\mathbf{W})$ and $\mathbf{A}'_i = \mathcal{F}(I'(i);\mathbf{W})$ with the learned network parameter $\mathbf{W}$. To further boost the matching performance, we propose multi-scale cost computation scheme, where the matching cost volumes are first built in a multi-scale manner, and then fused to estimate a final disparity map. It should be noted that in objection segmentation [23] or detection [24], multi-scale cost computation schemes have shown improved performance.

Based on share-net [9], we first build an image pyramid with multiple scale $s \in \{1, 1/2, ..., (1/2)^{S-1}\}$, where S is the number of scales. An image at each scale, I^s, is passed through the network, and the matching cost volume is built at each scale such that $C^s(i,d)$ where the maximum disparity range is also reduced as $d = \{1, ..., s \cdot d_{\max}\}$. To combine the matching cost, each matching cost volume $C^s(i,d)$ is resized to have the same resolution of original image. We use the bilinear interpolation to upsample the spatial resolution and duplicate the matching cost across disparities. Finally, we average multiple cost volumes to estimate final cost volume and determine the final disparity map.

3. EXPERIMENTAL RESULTS AND DISCUSSION

3.1. Experimental Settings

In our experiments, we implemented our network using the VLFeat MatConvNet toolbox[25]. We set all the initial network parameters as Gaussian random variable. We trained the network by minimizing two kinds of loss functions with stochastic gradient descent back propagation. For the hyperparameters we use, we set the threshold parameter for correspondence t to 3 and for the gradient g to 0.0625. The color constraint c was set to 0.02 and 0.13 and the interval of offset o_i to $(2, 6)$ and $(4, 10)$ for the Middlebury [11] and

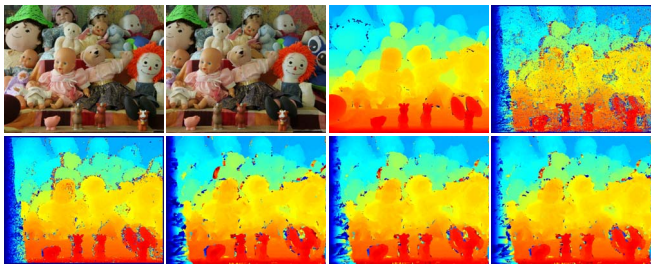


Fig. 4. Comparison of qualitative evaluation on the Middlebury [11]: (from left to right, from top to bottom) Left and right input image, ground-truth disparity map, estimated disparity maps using census [4], MC-CNN [3], our network with initial parameter, correspondence contrastive loss, and correspondence cross-entropy loss.

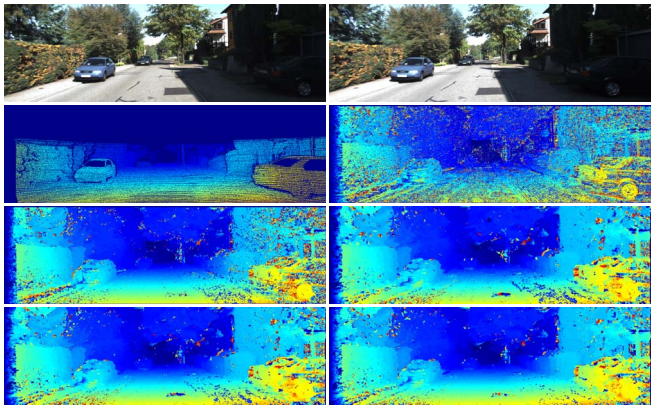


Fig. 5. Comparison of qualitative evaluation on the KITTI [12]: (from left to right, from top to bottom) Left and right input image, ground-truth disparity map, estimated disparity maps using census [4], MC-CNN [3], our network with initial parameter, correspondence contrastive loss, and correspondence cross-entropy loss.

KITTI [12] dataset as [3]. For both training and testing, we put the gray-scale image as an input. We compared our method with conventional hand-crafted features, census [4], and supervised CNNs based method, fast version of MC-CNN [3], on various benchmarks: Middlebury [11] as the concatenation of 2005 and 2006 version for a total of 27 images and KITTI [12] as 2012 version for a total of 194 images. We randomly split the dataset as training set and test set and trained our network only with training set and calculated qualitative evaluation on test set. For quantitative evaluation, we utilized the bad pixel error rates in non-occluded regions with 2- and 3-pixel margins. Furthermore, we evaluated our method with randomly initialized parameters.

3.2. Middlebury Benchmark

We first evaluated the effects of multi-scale cost computation in our learning framework on Middlebury stereo dataset [11] as shown in Fig. 3. The single-scale with $s = 1$ tends to preserves the fine detail, but has noises in low-textured region. On the single-scale with $s = 1/2$ shows more smooth results, but comes at risk of losing details. Our proposed multi-scale method with $s \in \{1, 1/2\}$, $S = 2$, shows better performance compare to results of the single-scale, which shows more smooth result while preserving fine details. Moreover, Fig. 4 shows qualitative results of our method compared to state-of-the-art methods. Interestingly, our unsupervised learning outperforms even supervised learning based method.

Table 1. Comparison of quantitative evaluation on Middlebury [11] and KITTI [12]. We refer to our network with an initial parameter as 'Ours_init.', with correspondence contrastive loss as 'Ours_reg.' and with correspondence cross-entropy loss as 'Ours_cls.'.

Error rates(%)		Middlebury [11]		KITTI [12]	
		> 2 px	> 3 px	> 2 px	> 3 px
Census [4]		32.53	30.72	49.14	45.82
MC-CNN [3]		17.51	16.63	22.65	20.22
Ours_init.	Single	20.79	18.93	30.90	29.40
	Multi	19.39	17.27	16.55	14.79
Ours_reg.	Single	18.51	17.11	28.26	25.87
	Multi	16.83	15.22	14.99	12.56
Ours_cls.	Single	17.98	16.88	24.31	21.90
	Multi	**16.81**	**15.08**	**14.46**	**11.96**

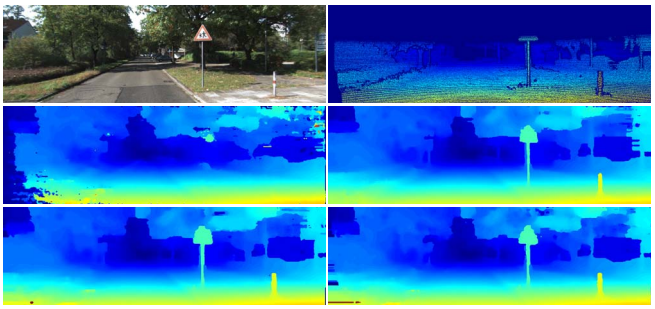


Fig. 6. Comparison of qualitative evaluation on KITTI [12] through post-processing steps: (from left to right, from top to bottom) Left input image, ground-truth disparity map, estimated disparity maps using census [4], MC-CNN [3], our network with correspondence contrastive loss and correspondence crossentropy loss.

3.3. KITTI Benchmark

We then evaluated our learning framework on KITTI stereo dataset [12] as shown in Fig. 5. Table 1 shows average error rates on Middlebury [11] and KITTI [12] dataset. Thanks to robustness of correspondence cross-entropy loss in an unsupervised manner and multi-scale cost computation, our method has shown the state-of-the-art performance compared to census [4] and MC-CNN [3] on various benchmarks.

Finally, we employed a common series of post-processing step as in [3], which consist of semiglobal matching, interpolation, subpixel enhancement and refinement as shown in Fig. 6, which also provides the reliable performance of our method.

4. CONCLUSION

We have presented the unsupervised learning framework for training the CNN architecture to compute the matching cost. To train the network as an unsupervised manner, we formulated the image domain learning combined with stereo epipolar constraint. We leveraged the correspondence consistency between stereo images as supervision for network training. With training sample mining scheme and multi-scale cost computation, the performance of the network could be boosted. Without using ground truth disparity map, proposed method has outperformed the state-of-the-art method with supervised approaches on various benchmarks.

5. ACKNOWLEDGMENTS

This work was supported by Institute for Information and communications Technology Promotion(IITP) grant funded by the Korea government(MSIP)(No.2016-0-00197).

6. REFERENCES

[1] D. Scharstein and R. Szeliski, "A taxanomy and evaluation of dense two-frame stereo correspondence algorithms," *IJCV*, vol. 47, no. 1-3, pp. 7–42, 2002.

[2] B. Ham, D. Min, C. Oh, M. N. Do, and K. Sohn, "Probability-based rendering for view synthesis," *IEEE Trans.IP*, vol. 23, no. 2, pp. 870–884, 2014.

[3] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *JMLR*, vol. 17, no. 1-32, pp. 2, 2016.

[4] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," *In: ECCV*, pp. 151–158, 1994.

[5] H. Hirschmuller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *IEEE Trans.PAMI*, vol. 31, no. 9, pp. 1582–1599, 2009.

[6] A. Honsi, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz, "Fast vost-volume filtering for visual correspondence and beyond," *IEEE Trans.PAMI*, vol. 35, no. 2, pp. 504–511, 2013.

[7] S. Kim, B. Ham, B. Kim, and K. Sohn, "Mahalanobis distance cross-correlation for illumination-invariant stereo matching," *IEEE Trans.CSVT*, vol. 24, no. 11, pp. 1844–1859, 2014.

[8] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Trans.PAMI*, vol. 30, no. 2, pp. 328–341, 2008.

[9] Z. Chen, X. Sun, L. Wang, Y. Yu, and C. Huang, "A deep visual correspondence embedding model for stereo matching costs," *In: ICCV*, pp. 972–980, 2015.

[10] W. Luo, A. G. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," *In: CVPR*, pp. 5695–5703, 2016.

[11] H. Hirschmuller and D. Scharstein, "Evaluation of cost functions for stereo matching," *In: CVPR*, pp. 1–8.

[12] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Are we ready for autonomous driing? the kitti vision benchmark suite," *IJRR*, vol. 32, no. 11, pp. 1231–1237, 2013.

[13] R. Garg, B. Kumar, G. Carneiro, and I. Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," *In: ECCV*, pp. 740–756, 2016.

[14] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," *arXiv:1609.03677*, 2016.

[15] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *In: NIPS*, pp. 2366–2374, 2014.

[16] S. Choi, D. Min, B. Ham, Y. Kim, C. Oh, and K. Sohn, "Depth analogy: Data-driven approach for single image depth estimation using gradient ssamples," *IEEE Trans.IP*, vol. 24, no. 12, pp. 5953–5966, 2015.

[17] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Trans.PAMI*, vol. 38, no. 10, pp. 2024–2039, 2016.

[18] C. B. Choy, J. Gwak, and Savarese S., "Universal correspondence network," *In: NIPS*, pp. 2406–2414, 2016.

[19] X. Sun, X. Mei, S. Jiao, M. Zhou, Z. Liu, and H. Wang, "Real-time local stereo via edge-aware disparity propagation," *PRL*, vol. 49, pp. 201–206, 2002.

[20] P. Agrawal, J. Carreira, and J. Malik, "Learning to see by moving," *In: ICCV*, pp. 37–45, 2015.

[21] Zagoruyko S. and N. Komadakis, "Learning to compare image patches via convolutional neural networks," *In: CVPR*, pp. 4353–4361, 2015.

[22] E. Simo-Serra, C. Torras, and F. Moreno-Noguer, "Dali: deformation and light invariant descriptor," *IJCV*, vol. 115, no. 2, pp. 136–154, 2015.

[23] L. C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," *In: CVPR*, pp. 3640–3649, 2016.

[24] T. Y. Lin, P. Dollar, R. Girshick, K. He, H. Bharath, and B. Serge, "Feature pyramid networks for object detection," *arXiv:1612.03144*, 2016.

[25] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," *ACM-MM*, pp. 689–662, 2015.