CONVOLUTIONAL FEATURE PYRAMID FUSION VIA ATTENTION NETWORK

Sangryul Jeon *Seungryong Kim* *Kwanghoon Sohn*

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

ABSTRACT

We present a novel fusion scheme between multiple intermediate convolutional features within convolutional neurual network (CNN) for dense correspondence estimation. In contrast to existing CNN-based descriptors that utilize a single convolutional activation, our approach jointly uses multiple intermediate features of CNN through the attention weight that balances the contribution of each features. We formulate the overall network as two sub-networks, correspondence network and attention network. The correspondence network is designed to provide multiple intermediate matching costs while the attention network is to learn the optimal weight between them. These two networks are learned in a joint manner to boost the correspondence estimation performance. Experiments demonstrate that our proposed method outperforms the state-of-the-art methods on various correspondence estimation tasks including depth estimation, optical flow, and semantic correspondence.

Index Terms— convolutional neural networks, attention mechanism, feature pyramid, dense correspondence

1. INTRODUCTION

Establishing dense correspondences across visually or semantically similar images is a fundamental step for numerous computver vision tasks such as structure-from-motion, visual SLAM, stereo matching, optical flow, non-rigid 3D reconstruction, and scene flow [1, 2, 3, 4].

To estimate reliable correspondences, designing robust feature descriptors or similarity functions to measure the matching evidence is one of the most important issues. Conventionally, they were designed to provide the invariance for photometric and geometric deformations between images in a hand-crafted manner [5]. Over the past few years, recent developments in the design of local image descriptors have been moved from carefully-engineered features to convolutional neural network (CNN) based features, due to higher level of robustness of CNNs [6] for visual deformantions. One of the key-component of the high performance is their hierarchical convolutional architecture to learn progressively complex visual features from low-level filters to high-level concepts . For dense correspondence estimation, a high level of invariance to visual deformations could potentially be achieved with deeper convolutional networks or large receptive fields, but would come at the cost of significantly reduced localization precision in matching details [7]. Such a trade-off between appearance invariance and localization precision induces inherent limitations on dense correspondence estimation.

On the other hand, in other computer vision applications, such as semantic segmentation and detection [8, 9], many approaches have exploited multiple intermediate features in CNN to overcome these limitations. They mainly have employed two types of network structures, *share*-net and *skip*-net. The first type, *share*-net, builds resized input images in multiple scales, passes each through a single shared network, and computes the final prediction based on the fusion of resulting multi-scale features [8]. Although they could incorporate more contextual information through a multi-scale feature representation, they need an intensive computational time and memory for network training due to the shared structures in the feature extraction network. Since a single shared network is used to generate multi-scale features, these methods inherently cannot handle the tradeoff between appearance invariance and localization precision. The second type, *skip*-net, exploits the multiple features from the intermediate layers of CNNs [10, 11, 12]. Based on the aggregation of features from hierarchical convolutional layers, they could potentially overcome the trade-off between appearance invariance and localization precision. However, exisiting these fusion approaches utilize a simple strategy to aggregate the matching costs estmated from multiple features in a hand-crafted manner, such as average- [13] or max-fusion [14], leading to a limited performance.

To address these issues, we propose a novel fusion scheme for multiple intermediate features within a CNN architecture to provide the appearance invariance and localization precision simultaneuosly. Our key-insight is that the intermediate features of CNN can be combined with a proper weight function that balances the contributions between them to provide an optimal matching performance. We formulate this compact intuition in a learning framework as an attention mechanism based on CNNs that learns the soft weight to fuse intermediate matching costs. We design a CNN architecture to estimate the multiple matching costs between intermediate feature candidates in the correspondence network and measure the optimal attention to aggregate these matching costs with the soft weight. These two networks are jointly learned in a synergistic manner through an adaptive correspondent classification loss layer, which possesses differentiability enabling an end-to-end training. Experimental results demonstrate that our network can be applied to various dense correspondence estimations including depth estimation, optical flow, and semantic correspondences, which proves our outstanding performance compared to the state-of-the-art methods.

2. PROBLEM FORMULATION AND CHALLENGES

Let us define a source and target image as I and I'. Formally, dense correspondence estimation, such as depth estimation, optical flow, and semantic correspondence, can be formulated as pixel-labeling problems. For pixel $i = [x_i, y_i]^T$ in a source image I, the goal of the task is to establish a distinctive correspondence i' among matching cadidates within a target image I'. To this end, it first defines a similarity function $\mathcal{S}(I_i, I'_l)$ for all possible matching candidates l, and then finds an index to produce a local minimum cost such that $i' = \mathrm{argmin}_l \mathcal{S}(I_i, I'_l)$. The matching candidate set can be varied according to problem formulations, e.g., 1-D search space for depth estimation and 2-D search space for optical flow and semantic correspondence.

To estimate reliable correspondences, the design of matching cost function $\mathcal{S}(I_i, I'_l)$ is one of the most important issues to discriminate positive and negative samples among the correspondence candidates efficiently and effectively. By leveraging CNNs, the matching cost function can be formulated as the similarity between resultant convolutional activations through feed-forward processes $A = \mathcal{F}(I; \mathbf{W}_c)$ and $A' = \mathcal{F}(I'; \mathbf{W}_c)$ with the siamese network parameters $\mathbf{W}_c$ such that

$$\mathcal{S}(I_i, I'_l) = \Phi(A_i, A'_l), \tag{1}$$

where $\Phi(\cdot, \cdot)$ is the similarity function.

To overcome a trade-off between an invariance to deformation and localization precision which conventional CNN-based methods have encountered, several efforts [11, 8] have been proposed to fuse the intermediate activations within CNNs, utilizing multiple intermediate features from a single network. One attempt is to measure the matching cost $\mathcal{S}_u(I_i, I'_l)$ by aggregating the multiple activations with an uniform weight such that

$$\mathcal{S}_u(I_i, I'_l) = 1/S \sum\nolimits_s \Phi(A^s_i, A'^s_l), \tag{2}$$

where $A^s = \mathcal{F}(I; \mathbf{W}^s_c)$ with intermediate convolutional parameters $\mathbf{W}^s_c$ and S is the number of intermediate activation levels. However, such a simple feature concatenation does not consider how much each intermediate feature contributes on estimation, causing the erroneous prediction from the less meaningful one to be propagated.

Another attempt is to measure the matching cost $\mathcal{S}_m(I_i, I'_l)$ by finding the maximal matching cost across scale s to select the most appropriate feature such that

$$\mathcal{S}_m(I_i, I'_l) = \max_s \left\{ \Phi(A^s_i, A'^s_l) \right\}. \tag{3}$$

This scheme has been adopted in the hand-crafted descriptor fusion techniques [15], but they are limited to select only one descriptor among the intermediate features. Furthermore, although these two types of fusion provide both robustness in variations and localization precision to some extent, they are inherently bounded to estimate an non-optimal similarity between the feature activations since there is no cue to balance the contributions of each activation.

3. PROPOSED METHOD

3.1. Network Configuration

Unlike existing fusion schemes [11] that aggregate the matching cost of intermediate convolutional activations in a hand-crafted manner, we formulate the activation fusion in a learning framework to predict an optimal weight in a fully convolutional and end-to-end manner. Our approach adaptively combines multiple intermediate convolutional features with an attention model, which balances the contributions from the intermediate features for each pixel. To realize this, the overall network consists of two sub-networks, namely correspondence and attention network. The correspondence network is to extract a convolutional feature to establish correspondences, and the attention network is to balance the effects of the matching costs computed from each intermediate convolutional activation.

3.1.1. Correspondence network

Similar to conventional features with CNNs [6], our correspondence network consists of successive convolution layers. Moreover, subsampling layers, i.e., max-pooling layers, are inserted between them to provide a substantial robustness through the larger receptive fields

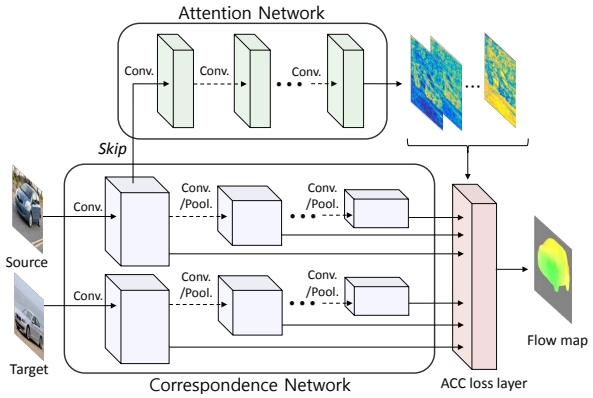

Fig. 1. The architecture of our network that consists of correspondence network and attention network. While the correspondence network learns the siamese convolutional layers to compute the matching costs across multiple scales, the attention network learns the attention to fuse multiple intermediate activations with a soft weight.

of the deeper convolutions and high localization precisions through the shallower convolutions simultanenously. These hierarchical activations enable us to deal with the trade-off between the appearance invariance and localization precision. From intermediate convolutional activations $A^s = \mathcal{F}(I; \mathbf{W}^s_c)$ from correspondence network, the similarity function to compute the matching cost is simply defined as an inner product such that

$$\Phi(A^s_i, A'^s_l) = < A^s_i, A'^s_l >, \tag{4}$$

where $< \cdot, \cdot >$ denote the inner product operator. Balancing the contribution of intermediate matching cost $\Phi(A^s_i, A'^s_l)$ across scales can be adjusted with an additional weight network, which will be described in the following section.

Specifically, the correspondence network consists of 9 or 12 convolutional layers according to the datasets, followed by rectified linear units (ReLUs) and batch normalization (BN) [16] except for the last convolutional layer. The total number of max-pooling layers inserted between convolutional layers is same to the intermediate feature levels S. For all convolution layers, the depth of kernel is 64 and the kernel size is 5 x 5.

3.1.2. Attention network

Formally, our matching cost can be formulated as a cost aggregation with its corresponding attention such that

$$\mathcal{S}(I_i, I'_l) = \sum\nolimits_s B^s_i \cdot \Phi(A^s_i, A'^s_l), \tag{5}$$

where B^s_i is the attention which balances the contribution of each intermediate feature across scale s at position i. While B^s_i defined in a hand-crafted fusion, such as average- or max-fusion, cannot consider the variations among intermediate features, our approach learns the attention through CNNs to predict the optimal weight such that $B^s_i = \mathcal{F}(I; \mathbf{W}_b)$ with attention network parameters $\mathbf{W}_b$.

To be specific, it consists of 3 or 5 convolutional layers depending on datasets, followed by ReLUs and BN. Furthermore, to keep the spatial resolution of the outputs to original one, the attention network does not contain the sub-sampling layers or stride scheme. Finally, the softmax layer is added to reduce the scale variation among attentions across scales for each pixel.

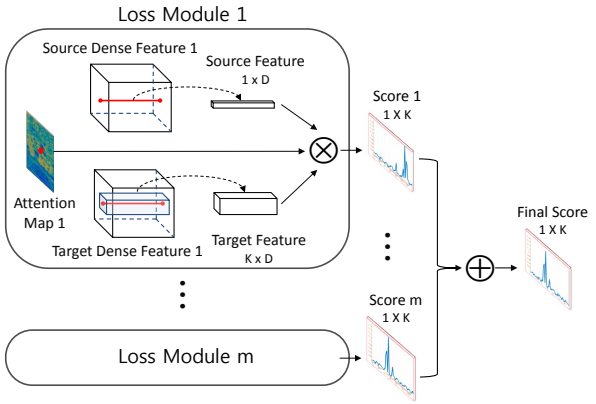

Fig. 2. Visualization of adaptive correspondent classification (ACC) loss layer. It is defined as a cross-entropy loss through intermediate convolutional activations with corresponding attentions. By aggregating these intermediate matching costs, the final loss is estimated. Note that the number of target feature candidates are set to K, which can be varied according to problem settings.

3.2. ACC: Adaptive Correspondent Classification Loss

To learn the overall network, we propose an adaptive correspondent classification (ACC) loss function. Since the correspondence estimation can be formulated as pixel-labeling problems, the loss function is also designed to train the network to predict the ground-truth probability at the corresponding position across matching candidates. Intuitively, we expect the ground-truth correspondent to have higher score while others have lower score as negative samples. We formulate the ACC loss layer that minimizes a cross-entropy loss with the softmax probability score function as an inner product.

For each pixel i and its possible correspondence candidates l, the ACC loss can be defined such that

$$\mathcal{L}(\mathbf{W}) = -\frac{1}{2N}\sum\nolimits_i \sum\nolimits_k P_{\mathrm{GT}}(k;i)\log(P(k;i)), \qquad (6)$$

where k is defined for all possible correspondence candidates. The ground-truth probability $P_{GT}(k;i)$ is 1 if k is a correspondent pixel and 0 otherwise. With the proposed matching cost $\mathcal{S}(I_i, I'_k)$ in (5), $P(k;i)$ is defined as a softmax probability such that

$$P(k;i) = \frac{\exp(\mathcal{S}(I_i, I'_k))}{\sum_l \exp(\mathcal{S}(I_i, I'_l))}, \qquad (7)$$

where l is defined for all possible disparities

3.2.1. *Differentiablity of ACC loss*

For end-to-end learning of the proposed network, the derivatives for the loss function must be computable, so that gradients of the final loss can be back-propagated to the shared correspondence network and the attention network simultaneously.

By the chain rule, the derivative of the final loss $\mathcal{L}$ with respect to A^s_i can be expressed as

$$\partial\mathcal{L}(\mathbf{W})/\partial A^s_i = \sum\nolimits_k (P_{GT}(k;i) - P(k;i))B^s_i A'^s_k, \qquad (8)$$

Similarly, $\partial\mathcal{L}(\mathbf{W})/\partial A'^s_k$ can be calculated.

Additionally, the derivative of the final loss $\mathcal{L}$ with respect to B^s_i can be formulated as follows:

$$\partial\mathcal{L}(\mathbf{W})/\partial B^s_i = \sum\nolimits_k (P_{GT}(k;i) - P(k;i)) < A^s_i, A'^s_k >. \qquad (9)$$

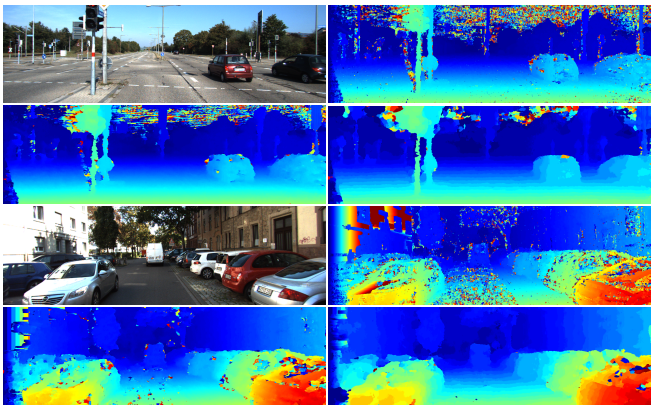

Fig. 3. Comparison of qualitative evaluations for stereo matching on the KITTI 2015 [17]: (from top to bottom, from left to right) left image and estimated disparity maps using MC-CNN [18], Luo *et al.* [19], and our method.

Table 1. Comparison of quantitative evaluations on the KITTI 2015 [17]. Average errors are measured with 2- and 3-pixel thresholds.

Methods	>2 pixel		>3 pixel	
	Non-Occ	All	Non-Occ	All
Census	49.13	50.35	32.53	45.82
MC-CNN [18]	18.47	20.04	14.96	16.59
Luo *et al.* [19]	9.96	11.67	7.23	8.97
Ours w.o/Att.	10.04	11.63	7.34	8.81
Ours	**9.81**	**11.08**	**7.16**	**8.76**

With $P_{GT}(k;i)) - P(k;i)$ term in the above derivatives, the large loss gradients are back-propagated to the both networks when the similarity score between source feature and ground-truth target feature is lower than other candidates. Furthermore, in (8), the loss gradients of A^s_i from matching candidate A'^s_k is weighted by B^s_i, thus the contribution to network training of candidate A'^s_k are adjusted according the B^s_i. In (9), the loss gradients of B^s_i are proposition to the similarity score $< A^s_i, A'^s_k >$, which decides the direction and amount of back-propagation. If A^s_i and A'^s_k is almost similar so that the score is near to the probability of 1, the network is learned to increase B^s_i or vice versa.

By formulating the ACC loss function for each intermediate feature and back-propagating the final loss gradients into their convolutional layers, the proposed network learns the *optimal weight* during network training without the tedious annotations of the *ground truth weight* for each intermediate feature.

4. EXPERIMENTAL RESULTS

4.1. Experimental Settings

In experiments, the proposed network was implemented using the Torch 7 toolbox [20]. Considering the trade-off between efficiency and robustness, the number of intermediate feature levels S was set to 3. To train our network, we used the ground-truth pixel-wise correspondences from the dataset as negative samples and pixels over all the candidates within the searching range as negative samples. We employed the ADAM algorithm [21] and used a learning rate of 0.01. The learning rate is decreased by a factor of 5 every 40 epochs.

In the following, we comprehensively evaluated our proposed network through comparisons to state-of-the-art methods with var-

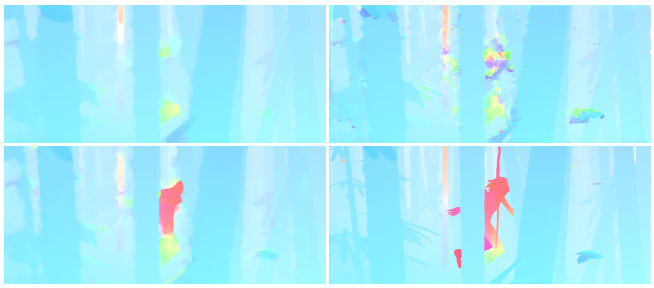

Fig. 4. Comparison of qualitative evaluations for optical flow on the MPI-Sintel benchmark [22]: (from top to bottom, from left to right) estimated flow fields using FlowNet [24], DeepFlow [25], Ours, and ground-truth flow fields.

Table 2. Comparison of quantitative evaluations on the MPI-Sintel benchmark [22]. Note that s0-10 is the EPE for pixels with motions between 0 and 10 pixels, and similarly s10-40 and s40+ are defined.

Methods	EPE-all	s0-10	s10-40	s40+
FlowNetS [24]	7.218	1.358	4.609	42.571
DeepFlow [25]	7.212	1.284	4.107	44.118
Ours w.o/Att.	7.208	1.291	4.045	42.021
Ours	**7.182**	**1.275**	**3.963**	**41.687**

ious dense correspondence estimation problems: disparity estimation on KITTI 2015 benchmark [17], optical flow on MPI-SINTEL benchmark [22], and semantic matching on Taniai benchmark [23]. For each task, we compared several state-of-the-art methods and showed how our proposed network significantly improves the correspondence performance. To additionally validate the attention mechanism, we evaluated our method without training on the attention network where the features from correspondence network are fused with an uniform weight (Ours w.o/Att.).

4.2. Depth Estimation on KITTI Benchmark [17]

To evaluate our method on stereo matching, our method was first compared to state-of-the-art stereo matching methods such as MC-CNN [18], Luo *et al.* [19] on the KITTI 2015 benchmark [17]. Note that for a fair comparison we do not employ smoothing or post processing schemes. To train the network, we randomly selected 160 image pairs as training set and use the remaining 40 image pairs for validation purposes. Our network consist of 5×5 convolutions with 2 max-pooling layers inserted after 3th and 6th convolutional layer, which results a largest receptive field size of 85×85 pixels. For quantitative evaluations, we used the bad-pixel error rates with ground-truth disparity maps. Fig. 3 and Tab. 1 show comparisons of qualitative and quantitative evaluations on KITTI 2015 benchmark [17]. As shown in results, our network outperforms previous CNNs based methods [18, 19] by a large margin on all criteria. Interestingly, we observed that our approach does not suffer from texture-less and repetitive regions while preventing from being blurred thanks to the optimally learned attentions as exemplified in Fig 4. The attention for $s = 3$ represents higher contribution than others on average, since they have larger receptive fields.

4.3. Optical Flow on MPI-Sintel Benchmark [22]

We also evaluated our method in optical flow settings on the challenging MPI-Sintel benchmark [22], which consists of more than

Fig. 5. Comparison of qualitative evaluations for semantic correspondence on the Taniai benchmark [23]. (from top to bottom, from left to right) source image, target image, Ours w.o/Att., and Ours.

1200 pairs of training images and 1500 pairs of testing images. For quantitative evaluations, we used end-point-error (EPE), which is the average euclidean distance between the flow fields. Our network consist of 12 layers of 3×3 and 5×5 convolutions with 4 max-pooling layers in a receptive field size of 137×137 pixels for flow estimation. For a fair comparison to existing optical flow methods, we removed outliers through forward-backward consistency check and refined the flow fields with a variational method similar to [26]. To be specific, we discarded inconsistent motion estimations with the 3 pixel constraints through the correspondence consistency. We interpolated or extrapolated the missing pixels with Epicflow [27]. Fig. 5 demonstrate comparison of our method with state-of-the-art methods such as FlowNet [24] and Deepflow [25]. Clearly, on challenging regions, e.g., small objects with large motion, our method estimated reliable flow fields with the help of balancing the trade-off between intermediate features. Tab. 2 shows quantitative evaluations. As in results, our proposed method outperforms other state-of-the-art algorithms especially for large displacements.

4.4. Semantic Matching on Taniai Benchmark [23]

Lastly, we evaluated our fusion framework on the Taniai benchmark [23] for semantic correspondence, which consists of 400 image pairs divided into tree groups: FG3DCar, JODS, and PASCAL with the ground-truth flow map on the foreground object. We formulated our method with the ImageNet pretrained VGG-Net [28] from the bottom conv1 to the conv3-4 layer for the initial parameters. Three max-pooling layers are located cafter conv2-2,conv3-2, and conv3-4. Fig. 5 demonstrates comparison of quantitative evaluations, which prove the robustness of our method.

5. CONCLUSION

We presented the fusion scheme for multiple intermediate features within CNN for dense correspondence estimation. To boost the matching performance, our key-insight is to combine multiple intermediate features of CNN with an attention that balances the contributions between them. We proposed the attention network that can be jointly learned with the feature extraction network. Thanks to its optimal combination between multiple intermediate features, our method has shown high correspondence estimation performance to provide the appearance invariance and localization precision simultaneously in dense correspondence estimation.

6. 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)

7. 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] C. Liu, J. Yuen, and A Torralba, "Sift flow: Dense correspondence across scenes and its applications," *IEEE Trans. PAMI*, vol. 33, no. 5, pp. 815–830, 2011.

[3] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *IEEE Trans. PAMI*, vol. 33, no. 3, pp. 500–513, 2011.

[4] P. Sturm and B. Triggs, "A factorization based algorithm for multi-image projective structure and motion," *In ECCV*, 1996.

[5] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[6] K. Alex, S. Ilya, and E. H. Geoffrey, "Imagenet classification with deep convolutional neural networks," *In Proc. of NIPS*, 2012.

[7] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Deepmatching: Hierarchical deformable dense matching," *IJCV*, vol. 120, no. 3, pp. 300–323, 2016.

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

[9] Dollar P. Lin, T.Y. and R. Girshick, "Feature pyramid networks for object detection," *arXiv:1612.03144*, 2016.

[10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *In CVPR*, pp. 3431–3440, 2015.

[11] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," *In CVPR*, pp. 447–456, 2015.

[12] S. Kim, D. MIn, B. Ham, S. Jeon, S. Lin, and K. Sohn, "Fcss: Fully convolutional self-similarity for dense semantic correspondence," *arXiv:1702.00926*, 2017.

[13] D. Ciresan, U. Meier, and J. schmidhuber, "Multi-column deep neural networks for image classification," *in CVPR*, pp. 3642–3649, 2012.

[14] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. PAMI*, vol. 32, no. 9, pp. 1627–1645, 2010.

[15] K.J. Hsu, Y.Y Lin, and Chuang.Y.Y., "Robust image alignment with multiple feature descriptors and matching-guided neighborhoods," *In CVPR*, pp. 1921–1930, 2015.

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

[17] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," *In CVPR*, pp. 3061–3070, 2015.

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

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

[20] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A matlab-like environment for machine learning," *In NIPS Workshop*, 2011.

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

[22] D.J. Butler, J. Wolf, G.B. Stanley, and M.J. Black, "A naturalistic open source movie for optical flow evaluation," *In ECCV*, pp. 611–625, 2012.

[23] T. Taniai, S.N. Sinha, and Y. Sato, "Joint recovery of dense correspondence and cosegmentation in two images," *in CVPR*, pp. 4246–4255, 2016.

[24] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," *In ICCV*, pp. 2758–2766, 2015.

[25] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," *In ICCV*, pp. 1385–1392, 2015.

[26] M. Bai, W. Luo, K. Kundu, and R. Urtasan, "Exploiting semantic information and deep matching for optical flow," *In ICCV*, 2015.

[27] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Epicflow: Edge-preserving interpolation of correspondences for optical flow," *In CVPR*, pp. 1164–1172, 2015.

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