Stereo Matching Confidence Learning based on Multi-modal Convolution Neural Networks

Zehua FU[1] and Mohsen ARDABILIAN FARD[1]

Universite de Lyon - Ecole Centrale de Lyon
Dp. Mathematiques Informatique
Laboratoire LIRIS - UMR 5205 CNRS
36, avenue Guy de Collongue
69134 Ecully Cedex - France

Abstract. In stereo matching, the correctness of stereo pairs matches, also called confidence, is used to improve the dense disparity estimation result. In this paper, we propose a multi-modal deep learning approach for stereo matching confidence estimation. To predict the confidence, we designed a Convolutional Neural Network (CNN), which is trained on image patches from multi-modal data, namely the source image pairs and initial disparity maps. To the best of our knowledge, this is the first approach reported in the literature combining multiple modality and patch based deep learning to predict the confidence. Furthermore, we explore and compare the confidence prediction ability of multiple modality data. Finally, we evaluate our network architecture on KITTI data sets. The experiments demonstrate that our multi-modal confidence network can achieve competitive results while compared with the state-of-the-art methods.

1 Introduction

Stereo matching is a fundamental problem in stereo vision. For two images of different views on the same scene, taken by cameras with horizontal displacement, the task of stereo matching is to find the corresponding pixels between the left and right images. The distance between the corresponding points is called disparity and the set of all disparities in the image is called disparity map. Despite decades of improvement, stereo matching still suffers from various issues, such as occlusion, ambiguity and extreme lighting conditions, which lead to incorrect stereo matches. In order to improve dense disparity estimation, several methods have been proposed to rate the correctness of matches. These methods are also called confidence measures.

According to the taxonomy proposed by Scharstein and Szeliski [1], stereo matching algorithms perform the following four steps (or subset of them): 1) Matching cost computation; 2) Cost aggregation; 3) Disparity computation/optimization and 4) Disparity refinement. The framework above produce several different types of data, including inputs and intermediate results, such as input RGB image pairs, matching cost volumes (MCVs) and initial disparity maps (IDMs), which are achieved by directly applying Winner Take All (WTA) strategy after matching cost computation. In early studies of confidence measures, approaches were designed and examined to estimate

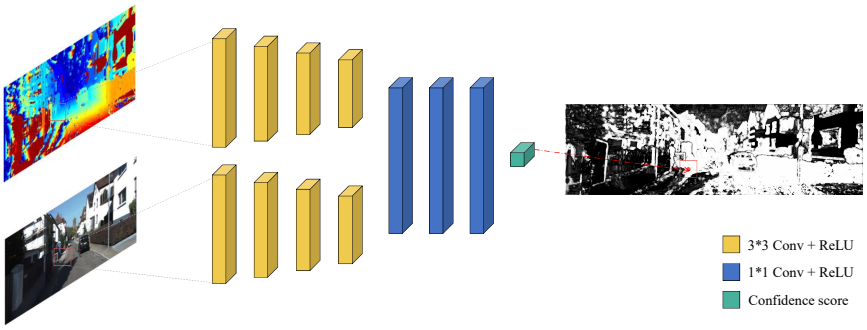

Fig. 1: The architecture of the proposed RGBD_LFN. By given two patches of different modalities including initial disparity map and RGB image, our goal is to estimate the confidence to correct matches on the current center pixel of patches.

the reliability of corresponding matches in stereo matching [2, 3, 4]. From the perspective of the used data type, manually designed measures can be categorized into three groups: 1) Confidence measures based on MCVs. In this category most approaches are related to the minimum, the second minimum matching cost, or a combination of them, e.g. Naive Peak Ratio (PKRN), Maximum Likehood Measure (MLM) and Left-Right Difference (LRD). 2) Confidence measures utilizing IDMs. Can be found in this category, approaches such as Left-Right Consistency (LRC), Variance of the Disparity Values (VAR) and the Median Deviation of Disparity Values (MDD). 3) Confidence measures employing source images pairs. As few approaches in this category, we give, as example, Magnitude of the Image Gradients Measure.

Hand-crafted confidence measures such as PKRN, MLM perform well on correct matches detection [4] but they have some weaknesses. They should be designed carefully with expertise and knowledge on stereo matching. Besides, most of them are only well suited for certain challenges. For example, Matching Score Measure (MSM) is the best choice for occlusion detection, while it has poor performance near discontinuities [4].

To alleviate the weakness of separate measures, some authors [5, 6, 7, 8] focus on feature combination approaches. Both Ensemble [5] and GCP [6] selected several confidence measures as feature vectors and applied random forests to train a regression classifier. After that, Park and Yoon [7] analyzed the specialty of various confidence measures and selected the effective ones by permutation importance through a regression forest framework. Then with the feature vectors of selected measures, they trained another random forest and used it to predict the confidence of correct correspondence. Their experimental results proved that the proposed regression forest could effectively select important confidence measures and their confidence estimation method outperformed method Ensemble [5] and GCP [6]. More recently, Poggi and Mattoccia [9] explored hand-crafted features for streaking detection in stereo matching. They pro-

posed an ensemble classifier trained by feature vectors similar to [5, 6, 7, 8] while achieved better results with time complexity of O(1).

Although joint features used for learning are thoughtfully formulated and selected, it is hard to make sure that all discriminating information has been taken into consideration. Recently, convolutional neural networks (CNNs) became popular in computer vision tasks because of their outstanding feature learning abilities [10, 11, 12, 13]. CNNs were first introduced to confidence prediction of correct matches in stereo matching by Zhong et al [14]. They proposed a siamese network [15] architecture, with two weight-shared sub-networks for both left and right image patches respectively for feature extraction. Following [14], Seki and Pollefeys [16] designed a 2-channel input patches for CNN based confidence learning. The design of input patches was inspired by left right consistency (LRC) measure with an assumption that the consistently matched pixels are correct. At the same time, Poggi and Mattoccia [17] proposed a patch-based CNN, learn confidence features of centre pixels by square patches from disparity maps.The experimental results indicate that both methods above are more efficient than the method proposed by Park and Yoon [7]. After that, Poggi et Mattoccia [18] proposed a deep learning based methodology to improve the effectiveness of the current top-performing confidence methods. Their experiments of 23 state-of-the-art confidence measures on three datasets discovered the local consistency in confidence map and demonstrated that this property can be learned by a deep network. Recently, Poggi et al. [19] summarized state-of-the-art stereo confidence measures and updated their review and quantitative evaluation based on Hu and Mordohai's work [4].

The contribution of this paper is mainly twofold. 1) we explore the confidence prediction ability of different types of data in stereo matching (e.g. source image pairs, MCVs and IDMs); 2) propose a novel CNN method which utilizes multi-modal data, including IDMs and referenced RGB images, as inputs. We explore and study two types of multi-modal CNNs on detecting disparity errors in stereo matching. Experimental results prove that our multi-modal approach can reach the state-of-the-art result on both KITTI2012 and KITTI2015 dataset.

The rest of this paper is organized as follows. Section 2 discusses how we select input data from stereo matching procedure, then describes two types of designed networks. Section 3 presents experimental results of confidence accuracy on challenging datasets and analyzes the results of different performances while comparing with other methods. Section 4 draws the conclusion to this paper.

2 Proposed method

In this section, we will begin with the background of proposed methods. Then we will discuss which types of stereo data can be used to train CNN networks for confidence estimation. For more than one modalities, there are several ways to construct networks and combine features. Therefore, we explore and test two kinds of models. Besides, training details will be mentioned at the end of this section.

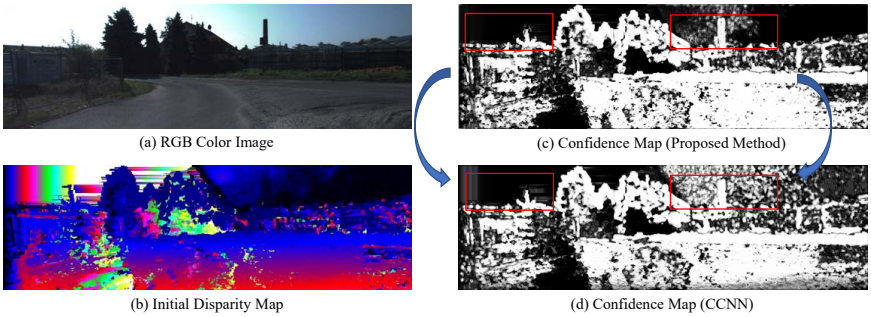

Fig. 2: Confidence quality result comparison of CCNN [17] and our proposed method on K12 Frame 99. Black pixels in (c) and (d) are considered as low confidence points.

2.1 Background

From the observation that the state-of-art confidence CNNs [17, 16] are all using disparity maps as inputs while there are several different kinds of data available in the stereo matching framework. In the early experiments, we trained the CNNs from patches of each data type that mentioned in Section 1. The structure of CNNs we used here are similar to Figure 3(a). From the results, we found that patches from both the entire matching cost volumes, patches combined minimum and second minimum matching cost as 2-channels almost do not have the ability to predict confidence, as these models did not converge in training stage. However, the latter one achieves good performance while producing manual features such as PKRN. We also found that CNN trained by initial IDM patches is equipped with high ability to differentiate incorrect matches while CNN trained by RGB image patches only have a very weak capacity.

Based on those observations, we trained a multi-modal Network to locate the error matches in IDMs. As shown in Figure 1, for every pixel in an IDM, we extract patches centered at current pixel both from IDM and related RGB image, then forward it to our MN, predicting the match correctness of current pixel.

2.2 Deep Network Architecture

According to Section 2.1, the initial disparity patches with one channel and its referenced RGB image patches with three channels are considered to be the inputs of the neural network. We explore two architectures with different fusion stages and fusion methods.

RGB-D Early Fusion Network(RGBD_EFN): This type of network simply considers IDM and referenced RGB image of an input pair as a 4-channel image. As shown in Figure 3(a), the network only has one branch during feature extraction, consisting of several convolutional and rectified linear unit(ReLU) layers. The following decision

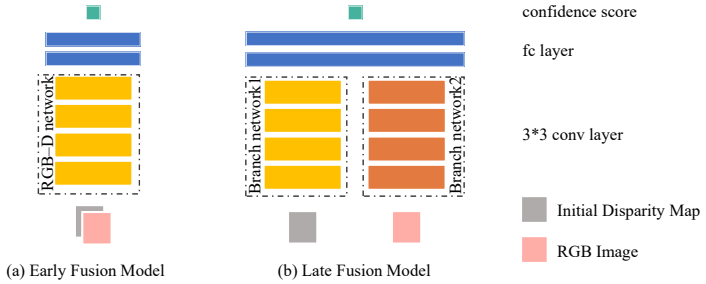

Fig. 3: Two architectures designed by different fusion stage. Architecture (a) is an early fusion structure which using RGB-Disparity 4-channel patches as input. For architecture (b), the features of two modalities are trained separately by two CNN branches without sharing weight. Then, the extracted features have a late fusion and forward to a decision network.

module consists simply of a number of fully connected layers with one output as the feature fusion network.

RGB-D Late Fusion Network(RGBD_LFN): As shown in Figure 3(b), it contains two sub-networks composed by convolutional layers and ReLU layers. The sub-networks extract feature vectors separately without sharing weights as siamese networks do, as we want to learn specific features of input data crossing domains. After being simply fused, the extracted feature vectors are forward propagated through several units of fully connected layers followed by ReLU layers. The last fully connected layer is followed by a sigmoid criterion to normalize the final result between 0 and 1, namely our confidence measure.

For both two networks above, each network has the fully connected (FC) layers. These FC layers will be replaced by fully convolutional layers with 1×1 kernels following Zbontar et LeCun [20]. We take the advantage that for those fully convolutional layers as the input size are not limited during test stage. So that we can predict the confidence of a sample through single forward pass rather than predicting patch by patch throughout the whole image. Besides, we also add paddings to all convolution layers to keep the size of images during prediction.

2.3 Details of learning

Optimization. We train all models with a binary cross-entropy (BCE) loss term,

$$Loss_{BCE} = -\sum_{i=1}^{N}(y_i \cdot log(\hat{y_i}) + (1-t) \cdot log(1-\hat{y_i})) \quad (1)$$

where y_i is the ground truth of i-th training sample, defining by the absolute difference from IDM to ground truth disparity map, $y_i \in \{0, 1\}$. $\hat{y}_i$ is the network output for the i-th training sample.

All networks are trained by mini-batch stochastic gradient descent (Mini-batch SGD) with batch-size, momentum term set to 128, 0.9 respectively. We trained for 15 epochs with the learning rate set to 0.003 at the beginning and decreased to 0.0003 at 11-th epoch. Weights are initialized by Xavier initialization method [21].

Preprecessing. During data preparation, patches size was set to 9×9. The number of convolution layers is set to 4. The number of fully connected layers and decision residual blocks are set to 3. In convolution layers, filter size is set to 3×3 with no padding. It is noteworthy that we just kept patches with valid non-occlusion disparity ground truth. For labeling the training data, the ground truth of confidence was set according to the central pixel disparity of the patch, as shown in Figure 1. Before sending to the network, we normalized each channel of RGB images and IDMs to $[0, 1]$.

All networks were implemented with torch7 [22] and cuDNN library [23].

3 Experiments

In this section, we introduce the challenging dataset and the evaluation methods we used at the very beginning. Then we design two experiments to evaluate proposed algorithms. In the first one, we compare the performance of the two proposed multi-modal confidence architectures. After that, we compare our best architecture with several state-of-the-art confidence methods. In the last experiment, we explored how training set size influences the confidence prediction accuracy.

3.1 KITTI datasei

We evaluate the performance of our method on the KITTI 2012 (K12) dataset [24, 25] and KITTI 2015 (K15) dataset [26]. K12 and K15 contain images from scenarios with varying weather conditions of a mid-size city, including rural areas and highways. The acquisition of K12 and K15 datasets were managed by two cameras (each of them has two units to capture the color images and grayscale images separately) settled on the top of a moving car, with a distance of 54 centimeters roughly. The stereo benchmark of K12 dataset consists of 194 training and 194 test rectified image pairs with a resolution of 1240×375 pixels, while the K15 dataset consists of 200 training and 200 test rectified image pairs with the same resolution. The training sets of both datasets contain semi-dense ground truth with sub-pixel accuracy but test sets not. Comparing with K12, K15 dataset contains more labels with dynamic objects like moving vehicles and denser labels with reflective regions like car glasses.

The disparity ground truths of KITTI datasets range from 1 to 255. According to the benchmark instructions, the correct estimation of a point is considered as the disparity error is less than 3. For the stereo confidence measures, we set the pixel-wise ground truth to 1 if the absolute differences between ground truth disparities and the initial disparities are no more than 3. Otherwise, the confidence values are set to 0.

3.2 Evaluation methodology

In order to evaluate the performance of our methods and compare the results with other state-of-the-art methods. We apply sparsification curves and its area under the curve (AUC) to benchmark quantitative accuracy refer to [4, 5, 7, 8]. For a confidence map of a given method, all effective pixels (pixels with ground truth) are sorted by descending confidence. Then the ordered pixels are divided into m equal parts (e.g, $m = 100$). Each time we pick the part with the lowest confidence of them and put down the bad pixel rate of the rest parts (bad pixel defined as differences larger than ± 3). In this way, we plot the sparsification curves. In the ideal case, all pixels with incorrect correspondence will be removed before correct ones, resulting in the optimal curves. The area under the optimal curve, namely optimal AUC, defined as:

$$A_{opt} = \int_{1-\varepsilon}^{1} \frac{d_m - (1-\varepsilon)}{d_m} = \varepsilon + (1-\varepsilon) ln(1-\varepsilon) \tag{2}$$

where ε presents the disparity error of current initial disparity map. Apparently, lower AUC values indicate better ability to predict confidence.

We using Δk to evaluate the improvement of method k [18] . As shown in Equation 3, AUC_{opt} presents the average optimal AUC value on each test dataset. Our baseline here is CCNN [17], which presented as AUC_{CCNN}.

$$\Delta k = \frac{AUC_{ccnn} - AUC_k}{AUC_{ccnn} - AUC_{opt}} \tag{3}$$

By using the defined AUC value, we evaluate our proposed method on K12 and K15 datasets with two stereo matching cost methods.

- SAD(Sum of Absolute Differences): A typical stereo algorithm. The MCVs are computed by using the absolute difference between two image intensities patches (9×9) from corresponding locations.
- MC-CNN [20]:A popular deep learning stereo method. A network is trained to calculate the matching cost by comparing the corresponding image patches. We used the pre-trained network (trained on K12, fast version) provided by authors.

After computing MCVs by two methods above, a *winner take all strategy* was simply applied to get the IDMs that we need.

3.3 Confidence Prediction Performance

We use AUC values to measure the confidence prediction abilities. First of all, for training learning-based classifiers, we split K12 dataset with ground truth (194 frames in total) into the training set (frames 0-93) and test set (frames 94-193). The whole K15 (frames 0-199) dataset is used as test set as suggested in [19]

Table 1: Confidence evaluation results with two matching cost algorithms on K12, K15 datasets. The first two rows are the comparison of alternative models. The following rows are comparisons with state-of-the-art methods(CCNN [17], CCNN$^+$ [18]). In order to prove the advantages of join RGB and disparity data cues, we also add CCNN* which used the similar structure except for the input modalities. ε is the average confidence error rate for each test dataset.

	K12 ($\varepsilon = 17.12\%$)		K15 ($\varepsilon = 17.46\%$)	
measure	AUC_k	Δ_k	AUC_k	Δ_k
RGBD_LFN	0.02362	28.98%	0.03208	15.13%
RGBD_EFN	0.02372	27.00%	0.03340	3.05%
CCNN*	0.02503	2.05%	0.0337	0.20%
CCNN$^+$	0.02489	4.68%	0.03306	6.12%
CCNN	0.02514	-	0.03373	-
Optimal	0.0199		0.0228	

(a) Confidence evaluation with MC-CNN matching cost

	K12 ($\varepsilon = 36.89\%$)		K15 ($\varepsilon = 32.79\%$)	
measure	AUC_k	Δ_k	AUC_k	Δ_k
RGBD_LFN	0.1061	45.77%	0.0921	8.89%
RGBD_EFN	0.1073	41.03%	0.0949	-2.39%
CCNN*	0.1157	8.16%	0.0933	3.90%
CCNN$^+$	0.1169	3.72%	0.0939	1.62%
CCNN	0.1178	-	0.0943	-
Optimal	0.0922		0.0699	

(b) Confidence evaluation with SAD matching cost

Alternative Models: For figuring out which designed multi-modal architecture works better on the confidence prediction task, we compared the proposed two multi-modal networks. The first part of Table 1 shows the average AUC values and Δk of each architecture by using MC-CNN and SAD matching cost. We can see that both RGBD_LFN and RGBD_EFN achieve good results on K12 dataset, while on K15 RGBD_LEF shows better generalization ability. Finally, we chose RGBD_LEF as it has the better performance.

Comparisons with State-of-the-art methods: To analyze the capability of our multi-modal method on predicting correct matches, we compared our method with two learn-

ing based methods proposed recently, CCNN [17], CCNN$^+$ [18]. Besides, we also add a model named CCNN*, which has the same structure with CCNN but use the same numbers of convolutional layers and fc layers as ours (CCNN: 64 conv layers, 100 fc layers; CCNN*: 112 conv layers, 384 fc layers). Again, all methods are trained on the same datasets mentioned above.

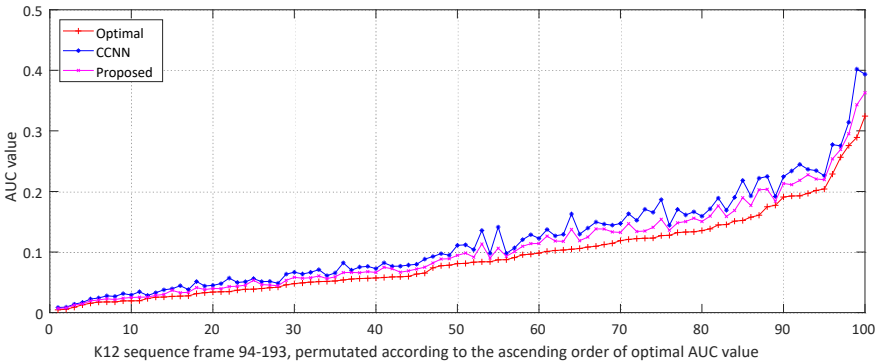

Fig. 4: Comparison of AUC values for 100 frames of K12 training image pairs with SAD MCVs. AUC values of the same method are sorting in the ascending order according to optimal AUC values. Here we selected CCNN [17] as a comparative item, which is trained by initial disparities patches only.

Fig. 6 shows a comparison of AUC values of the state-of-the-art learning based method CCNN and ours with SAD matching cost method on K12. AUC values of each method are sorted by the ascending orders of optimal AUC values. The curves of both two methods in Figure 6 wave with the similar trend to optimal AUC values. But the gap between optimal AUC and other two becomes larger when optimal AUC values increase. Refer to [7], it is more challenging to predict the confidence while the gap between optimal and predicted AUC values growing. From the figure, we can see clearly that the difference between our method and CCNN method becomes more obvious in the ascending order of optimal AUC values. So the comparison of AUC from different methods leads us to the conclusion that our method is more robust than CCNN.

While Comparing with other state-of-the-art methods as shown in Table 1, first of all, we can see that our RGBD_LFN method achieves the minimum average AUC above all. Observing the evaluation results between CCNN and CCNN*, we can find that CCNN model with increasing parameters can not provide significant improvement. We can also notice that our method is much better than CCNN* presented by Δ_k. It means that the improvement of our method is caused by the joint of RGB image features, rather than the increasing of convolutional and fully connected layers. The improvement of our method is much better than CCNN$^+$, an upgrade version of CCNN, learned the local consistency in confidence map produced by CCNN. This indicates that although the

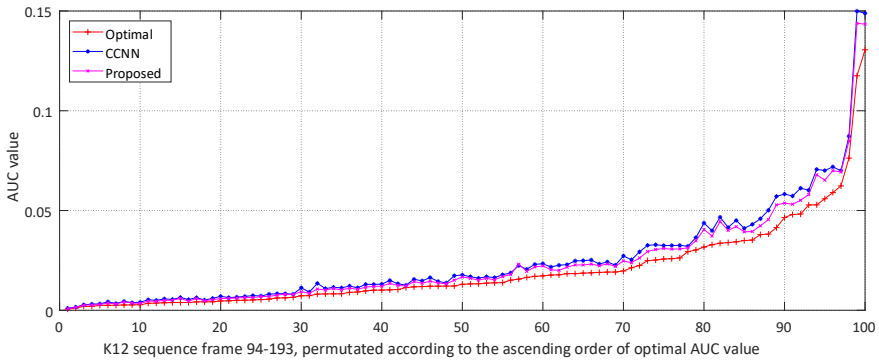

Fig. 5: Comparison of AUC values for 100 frames of K12 training image pairs with MC-CNN MCVs. AUC values of the same method are sorting in the ascending order according to optimal AUC values. Here we selected CCNN [17] as a comparative item, which is trained by initial disparities patches only.

leveraging information from neighborhood points can help to improve the effectiveness of confidence measures, many contents can not be learned due to the ambiguity of wrong disparities. For example, the texture less and repeat texture surface like walls, sky and greenbelts, resulting to peak regions or lots of noises in disparity maps as shown in 6, or dark places like shadows, leading to failure of matching cost calculation (wrong and small disparity values). The disparity maps based CCNN often failed in those regions. However, with the complement of RGB features, more information can be used to learn the edges in such kind of areas mentioned above. Finally, our method achieves the best performance with Both SAD and MC-CNN MCVs on K12 and K15 datasets. This indicates that our method has good generalization ability and independent to different MCVs.

3.4 Training set size

As we are using a deep learning method, we would like to explore whether the rising size of training data will improve the performance of confidence prediction. Hence we trained our network on the K12 dataset with incremental training samples and calculated average AUC values on the rest fourteen frames in K12 dataset. We use ratio described in Equation 4 to evaluate the improved performance with the training size ranges from 20 to 180.

$$ratio = \frac{AUC_k}{AUC_{opt}} \tag{4}$$

Figure 7 shows the results of our experiment. We note that the ratio decreases fast with the growing training set size. Then curve becomes almost stable after training set size up to 160.

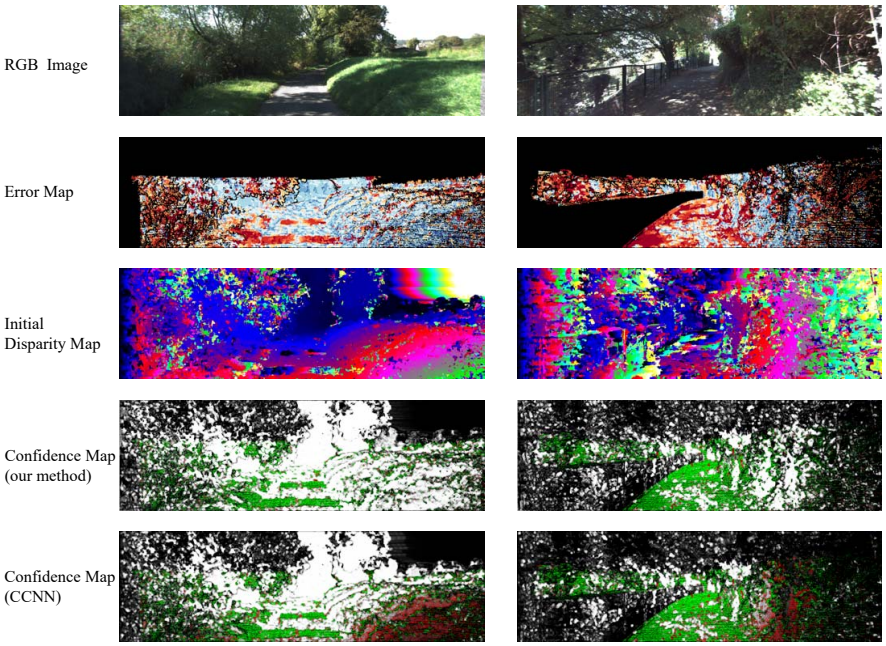

Fig. 6: The comparison of confidence maps with SAD on two most challenge K12 frames, which have the highest optimal AUC values. Notice that in error maps, red points are the low confidence labels. In the predicted confidence maps, darker points have lower confidence values. On each confidence map, we picked low confidence points using a threshold of 0.5. Then if the estimations are correct, we painted pixels in green, otherwise, we painted them in red.

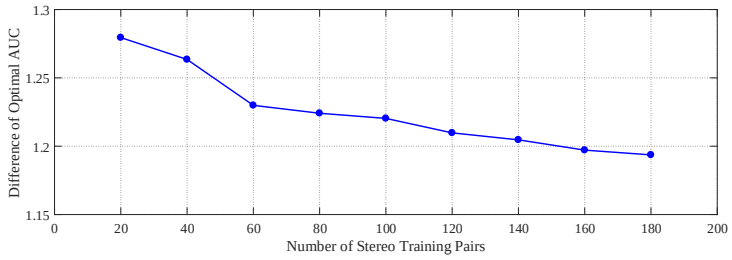

Fig. 7: The influence of training data size on performance, presenting as ratio between average AUC values of our method on K12 with SAD MVCs and average optimal AUC values.

4 Conclusion

In this paper, we explored the confidence prediction potential of different modalities and found both the initial disparity maps and the referenced RGB images have the capabilities. Based on this discovery, we proposed a multi-modal Convolutional Neural Network(CNN) for confidence estimation. We design and study on two architectures with different fusion stages and fusion methods. The experimental results show that the late fusion architecture achieves lower AUC values and has better generalization ability. It also has better performance while compared with several state-of-the-art methods. Overall, our approach shows the potential of feature fusion for confidence prediction in stereo matching, which is worthwhile for further research.

Acknowledgement We would like to thank Park and Yoon [7] for sharing his source code and also evaluation code. We also want to thank Seki and Pollefeys [16] for providing guidance on how to implement their algorithms.

References

[1] Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision **47**(1) (2002) 7–42 1

[2] Banks, J., Corke, P.: Quantitative Evaluation of Matching Methods and Validity Measures for Stereo Vision. The International Journal of Robotics Research **20**(7) (jul 2001) 512–532 2

[3] Egnal, G., Wildes, R.P.: Detecting binocular half-occlusions: empirical comparisons of five approaches. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**(8) (Aug 2002) 1127–1133 2

[4] Hu, X., Mordohai, P.: A quantitative evaluation of confidence measures for stereo vision. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(11) (Nov 2012) 2121–2133 2, 3, 7

[5] Haeusler, R., Nair, R., Kondermann, D.: Ensemble learning for confidence measures in stereo vision. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition. (June 2013) 305–312 2, 3, 7

[6] Spyropoulos, A., Komodakis, N., Mordohai, P.: Learning to Detect Ground Control Points for Improving the Accuracy of Stereo Matching. In: Computer Vision and Pattern Recognition 2014, Columbus, Ohio, United States (June 2014) 2, 3

[7] Park, M.G., Yoon, K.J.: Leveraging stereo matching with learning-based confidence measures. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2015) 101–109 2, 3, 7, 9, 12

[8] Spyropoulos, A., Mordohai, P.: Correctness prediction, accuracy improvement and generalization of stereo matching using supervised learning. International Journal of Computer Vision **118**(3) (2016) 300–318 2, 3, 7

[9] Poggi, M., Mattoccia, S.: Learning a general-purpose confidence measure based on O(1) features and a smarter aggregation strategy for semi global matching. Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016 (1) (2016) 509–518 2

[10] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. CVPR '14, Washington, DC, USA, IEEE Computer Society (2014) 1725–1732 3

[11] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. NIPS'12, USA, Curran Associates Inc. (2012) 1097–1105 3
[12] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. (June 2014) 1725–1732 3
[13] Dong, C., Loy, C.C., He, K., Tang, X. In: Learning a Deep Convolutional Network for Image Super-Resolution. Springer International Publishing, Cham (2014) 184–199 3
[14] Zhong, Z., Su, S., Cao, D., Li, S., Lv, Z.: Detecting ground control points via convolutional neural network for stereo matching. Multimedia Tools and Applications (2016) 1–16 3
[15] Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a "siamese" time delay neural network. In: Proceedings of the 6th International Conference on Neural Information Processing Systems. NIPS'93, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1993) 737–744 3
[16] Seki, A., Pollefeys, M.: Patch based confidence prediction for dense disparity map. In: British Machine Vision Conference (BMVC). Volume 10. (2016) 3, 4, 12
[17] Poggi, M., Mattoccia, S.: Learning from scratch a confidence measure. In: British Machine Vision Conference (BMVC). (2016) 3, 4, 7, 8, 9, 10
[18] Poggi, M., Mattoccia, S.: Learning to predict stereo reliability enforcing local consistency of confidence maps. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (2017) 3, 7, 8, 9
[19] Poggi, M., Tosi, F., Mattoccia, S.: Quantitative evaluation of confidence measures in a machine learning world. In: The IEEE International Conference on Computer Vision (ICCV). (Oct 2017) 3, 7
[20] Zbontar, J., LeCun, Y.: Stereo matching by training a convolutional neural network to compare image patches. Journal of Machine Learning Research **17** (2016) 1–32 5, 7
[21] Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS10). Society for Artificial Intelligence and Statistics. (2010) 6
[22] Collobert, R., Kavukcuoglu, K., Farabet, C.: Torch7: A Matlab-like Environment for Machine Learning. In: BigLearn, NIPS Workshop. (2011) 6
[23] Chetlur, S., Woolley, C., Vandermersch, P., Cohen, J., Tran, J., Catanzaro, B., Shelhamer, E.: cudnn: Efficient primitives for deep learning. CoRR **abs/1410.0759** (2014) 6
[24] Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. (June 2012) 3354–3361 6
[25] Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research **32**(11) (2013) 1231–1237 6
[26] Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2015) 6