Pixel-Wise Confidences for Stereo Disparities Using Recurrent Neural Networks

M. Shahzeb Khan Gul
muhammad.gul@iis.fraunhofer.de
Michel Bätz
michel.baetz@iis.fraunhofer.de
Joachim Keinert
joachim.keinert@iis.fraunhofer.de

Fraunhofer IIS
Moving Picture Technologies
91058 Erlangen, Germany

Abstract

One of the inherent problems with stereo disparity estimation algorithms is the lack of reliability information for the computed disparities. As a consequence, errors from the initial disparity maps are propagated to the following processing steps such as view rendering. Nowadays, confidence measures belong to the most popular techniques because of their capability to detect disparity outliers. Recently, convolutional neural network based confidence measures achieved best results by directly processing initial disparity maps. In contrast to existing convolutional neural network based methods, we propose a novel recurrent neural network architecture to compute confidences for different stereo matching algorithms. To maintain a low complexity the confidence for a given pixel is purely computed from its associated matching costs without considering any additional neighbouring pixels. As compared to the state-of-the-art confidence prediction methods leveraging convolutional neural networks, the proposed network is simpler and smaller in terms of size (reduction of the number of trainable parameters by almost 3-4 orders of magnitude). Moreover, the experimental results on three well-known datasets as well as with two popular stereo algorithms clearly highlight that the proposed approach outperforms state-of-the-art confidence estimation techniques.

1 Introduction

Stereo vision is an important and fundamental technique to understand the 3D structure of real-world imagery in the field of computer vision. Typically for a synchronized pair of images under different viewpoints of the same scene, stereo matching is used to find accurate corresponding points for each pixel to infer depth through simple triangulation. The distance between the corresponding points is called disparity and the set of all disparities in the image is called the disparity map. Stereo vision, despite being one of the most researched topics, lacks the ability to find out accurate disparities due to the ill-posed nature of the problem. This is particularly visible when dealing with occlusions, transparent or reflecting surfaces, and texture-less or repeated pattern regions [3]. Wrong disparity assignment limits the adaptability of stereo disparity estimation for practical systems. Over the years, different approaches have been proposed to address these limitations. Many of them focus primarily

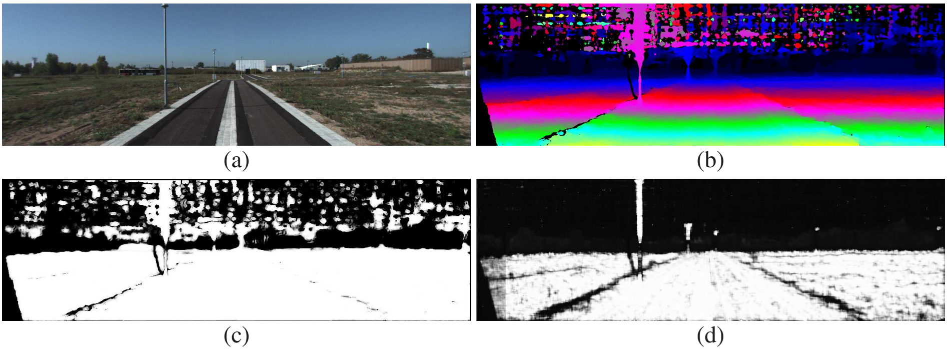


Figure 1: Qualitative result of 84th frame stereo pair from KITTI 2012 [8]. (a) reference image, (b) disparity map obtained using MC-CNN [34], (c) confidence map estimated using LGC-Net (LFN) [32], and (d) the proposed C-RNN method. Brighter values represents a higher confidence.

on the robust estimation of the matching cost measures [19, 34]. However, these robust cost measures are not able to fully solve this intrinsic problem of the stereo matching.

Nowadays, almost all stereo matching algorithms include post-processing steps for the refinement of the disparities. The very first step in this process is to filter out wrong assignments using confidence measures (CMs) aimed at providing a degree of uncertainty for each pixel and then correct them using reliable pixels [14, 30]. Conventional confidence measures are divided into the following three categories depending on the input data type:

- **Confidence measures based on matching cost volumes :** CMs in this group are dependent on the cost volume cues such as the minimum and the second minimum matching cost, or a combination of both. Examples of these CMs types are Native Peak Ration (PKRN) [15], Maximum Likehood Measure (MLM)[20] and Left-Right Difference (LRD) [15].

- **Confidence measures based on initial disparity maps :** Approaches relying on the cues from initial disparity maps are included in this category. For example, Left-Right Consistency (LRC) [6], Variance of the Disparity Value and the Median Deviation of Disparity Values (MDD) [23, 25, 30].

- **Confidence measures based on source images pairs :** Magnitude of the Image Gradients Measure [23, 30] and Distance to border (DB) [12, 23] are two examples for this kind of confidence measure.

Recently, researchers demonstrated that learning-based confidence measures [7, 24, 25, 29, 32] performed much better than the conventional confidence measures [15]. According to Poggi *et al*. [26], convolutional neural network (CNN) based confidence measures [24, 25, 29] which use information extracted from the disparity maps as input cue represent the state-of-the-art algorithm in terms of detection of correct matches and capability of adapting for different data. Early approaches as in [11], utilize conventional confidence measures within a random forest framework. However, recent CNNs achieved better results without relying on any hand-crafted features [26].

In this paper, we introduce a novel confidence measure which is inspired by both conventional and learning-based confidence measures. Recently, Op het Veld *et al*. [33] have proposed a confidence measure utilizing cues from the cost volumes and outperformed the state-of-the-art CNN-based confidence measure in [24]. Following the same strategy from [33], we have trained a recurrent neural network (RNN) on cost values of each pixel yielding better results. RNNs have internal memory, due to which they remember important things or cues about the input cost curve they received, such as number of local minima or distance between these local minima, which enables them to be very precise in predicting the confidence. Figure 1 shows the qualitative result of a frame from the KITTI 2012 dataset [8].

The contributions of this paper are as follows: We propose a novel RNN method named Confidence-RNN (C-RNN) which utilizes cost values for each pixel as input and has a very low complexity overall. Moreover, we extensively evaluated the performance of the proposed framework on three popular datasets, KITTI 2012 [8], Middlebury v3 quarter sized [28], and MPI Sintel [4] using two different stereo algorithms MC-CNN [34] and ADCensus [21]. Experimental results prove the reliability and accuracy of the proposed approach in successfully eliminating the wrong disparity values.

2 Related Work

2.1 Hand-Crafted Approaches

In literature, there are numerous methods estimating the reliability of the disparities based on hand-crafted features [6, 22]. In [15], an exhaustive review of such confidence measures has been conducted. According to the evaluation results, it is concluded that the confidence measures relying on a single feature would not perform well in confidence estimation. To alleviate the weakness of separate measures, there have been various approaches focusing on combining several features [11, 23, 25, 30]. They extract different cues from the estimated disparity map and cost volume, which are then used to train a simple classifier, for instance, a random forest. In [11], confidence features such as left-right consistency, image gradient, and disparity map variance are combined and trained within a random forest framework. A similar kind of approach is also proposed in [30]. The method proposed in [23] further improved the results by first selecting the optimal confidence features from multiple confidence features using a regression forest and then train the classifier using the selected pool of confidence features. Poggi and Mattoccia [25] utilize the confidence features extracted from the disparity map for the training and achieved better results within time complexity of **O(1)**. All above explained methods are working on pixel-level, whereas, [16] incorporated spatial context to estimate confidences at the superpixel-level.

In [33], a method for predicting pixel-wise reliability of the estimated disparity using the cues from the cost volume is proposed. The evaluation presented in [33] demonstrates much better results as compared to the current state-of-the-art CNN-based confidence measure [24].

2.2 CNN-Based Approaches

Deep learning has been successfully applied in different computer vision applications [5, 10, 18, 31, 35]. Due to the success of deep learning, researchers have also explored it for stereo disparity estimation, and in particular for confidence estimation [7, 24, 29, 32]. The convolutional neural network proposed in [24] estimates a confidence map using the left-to-right disparity map only. On the other hand, [29] predicts confidences on the basis of

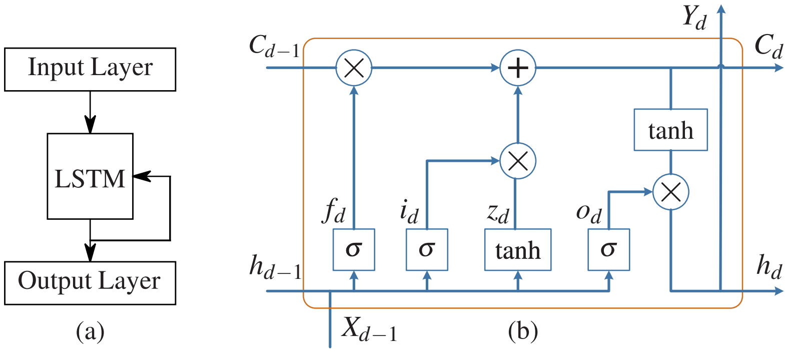


Figure 2: Proposed network architecture: (a) Network structure of the proposed recurrent neural network, and (b) LSTM memory block with a single memory cell.

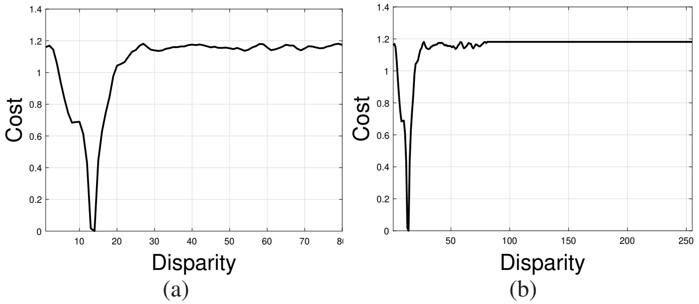


Figure 3: Cost curve modification for the training. (a) Raw cost curve (b) Padded cost curve.

both left-to-right and right-to-left disparity maps and hence, improves the accuracy of the prediction. Apart from straightforward CNN-based methods, [7] employed a multi-modal architecture which is trained on both the disparity map and the reference image. This strategy is then further extended by [32], exploiting more global context with local approaches and presenting a detailed study regarding early and late fusion of the predictions in the network. As opposed to all these methods, Kim *et al*. [17] proposed a convolutional neural network which takes a cost volume as input for the simultaneous prediction of a disparity map and a confidence map. Poggi *et al*. [26] extensively evaluated many of these confidence measures in comparison with conventional confidence measures and proved the efficiency of the deep learning-based approaches.

3 Proposed Method

In this section, the problem formation is given (in Section 3.1) before describing the proposed recurrent neural network for the prediction of confidences. Subsequently, the strategy for training the network is discussed (in Section 3.2).

3.1 Problem Formulation and Architecture

Given a stereo image pair $(I_{\mathrm{L}}, I_{\mathrm{R}})$. The objective of a stereo matching algorithm is to find out the correspondences between the stereo pairs and infer a disparity D_i for each pixel $i = [i_x, i_y]$. These stereo algorithms however, do not provide fully reliable disparity maps because of the ill-posed nature of the problem. Therefore, we aim to estimate confidences in a deep learning framework for the estimated disparity map so as to improve subsequent post-processing steps [11, 15, 22, 23, 30].

We have used existing stereo algorithms [21, 34] to obtain the matching cost volumes $C_{i,d}$ across a set of disparity candidates $d = 0, 1, 2, ...d_{\max}$, where $d_{\max}$ is the maximum disparity, and estimate its associated disparity map D_i. The obtained disparity map is then compared with the ground truth disparity map D^* to create ground truth confidence map Q^*. If the absolute difference between the estimated disparity D_i and the ground truth disparity D_i^* is smaller than a threshold τ then the ground truth confidence Q_i^* is set to 1 and otherwise to 0.

The architecture of the proposed network is rather shallow. The proposed network is a three layer Long Short-Term Memory (LSTM) [13] recurrent neural network consisting of one input layer, one output layer, and a hidden layer known as LSTM memory block as shown in Figure 2. The input of the network is a cost vector for a given pixel (storing all costs for all possible disparity values). The idea behind this architecture is derived from [33], i.e., if the matching cost vector of the pixel have multiple minima then it is highly probable that the disparity value of the pixel is not reliable. Therefore, we chose an LSTM RNN rather than using a CNN, because of its advantage and success for sequential data [9]. The LSTM memory block consists of a single memory cell and three multiplicative units (input, output, and forget gates) providing read, write, and reset operations for the cell. The number of hidden units – the size of the hidden state of the network – h_d is 5. In Figure 2, an LSTM memory block with a single LSTM cell is depicted. Each memory block has a recurrently self-connected linear unit, a constant error carousel (CEC), and the activation of the CEC indicates the cell state. The memory cell also mitigates the problem of exploding and vanishing gradients by allowing the state to remain unchanged from one time step to another for any outside interference. The functionality of the forget gate is to learn when to reset memory block (in case of an outdated state); this helps in preventing the cell state from growing without bounds. Likewise, the input gate is responsible for the modification of the cell state in response to the incoming signals, whereas the output gate decides whether the cell state should affect the other neurons or not.

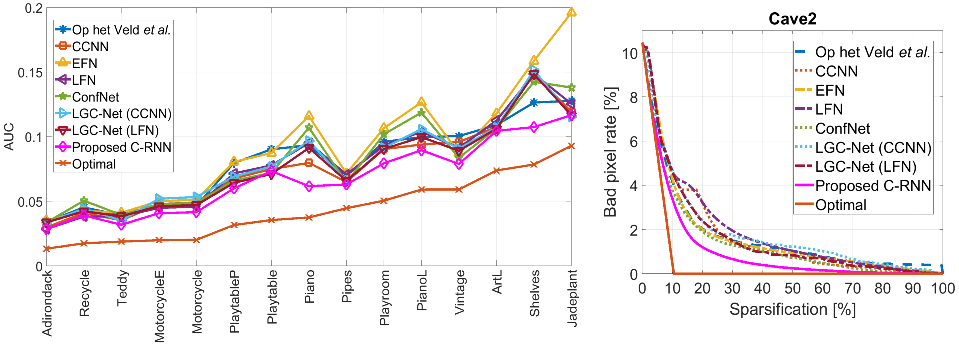
Let x_d be the input which corresponds to the cost value at a particular disparity d and y as the final output of the network, i.e., estimated confidence value Q for that pixel. The forward pass of the network is defined as

$$f_d = \sigma(W_f x_d + U_f h_{d-1} + b_f) \tag{1}$$

$$i_d = \sigma(W_i x_d + U_i h_{d-1} + b_i) \tag{2}$$

$$o_d = \sigma(W_o x_d + U_o h_{d-1} + b_o) \tag{3}$$

Equations 1-3 represent forget gate, input gate and output gate of the LSTM block, denoted by f_d, i_d, and o_d, respectively. W and U correspond to the weight matrices and b is the bias vector. Moreover, σ denotes the sigmoid activation function. The functionality of the input gate is to control the addition of new information into the cell, the forget gate is responsible for removing information from the cell state and the output gate selects the information from



(a) (b)

Figure 4: (a) AUC values computed for all the confidence measures on Middlebury v3 [28] sorted in the ascending order of optimal values, and (b) Comparison of the sparsification plots of the "Cave2" from MPI Sintel dataset [4].

CMs	CCNN [24]	EFN [7]	LFN [7]	ConfNet [32]	LGC-Net (CCNN) [32]	LGC-Net (LFN) [32]	Proposed C-RNN
# Trainable Parameters	128125	129853	247101	7855937	8347835	8466811	**146**

Table 1: Number of trainable parameters for different network architectures used for the confidence prediction of stereo pairs. The proposed C-RNN requires only a negligible amount of parameters compared to the state-of-the-art.

the current cell state to compute the output activation of the LSTM unit.

$$z_d = \tanh(W_z x_d + U_z h_{d-1} + b_z) \tag{4}$$

$$C_d = f_d \odot C_{d-1} + i_t \odot z_d \tag{5}$$

$$h_d = o_d \odot \tanh(C_d) \tag{6}$$

$$y = W_y h_{d_{max}} + b_y \tag{7}$$

z_d is the block input (4) which is a tanh layer and with the input gate, the two decide on the new information that should be stored in the cell state; whereas C_d and h_d denote the cell state and hidden state (or block output), (5) and (6) respectively. The symbol used for element-wise multiplication is $\odot$.

3.2 Training Procedure

The proposed network is implemented using the TensorFlow framework [2]. The network is trained on cost values obtained using different stereo algorithms. For each pixel, cost values over a disparity range is used as input; to estimate confidence as an output of that particular pixel. The disparity search range for each image is different, therefore we have to modify the length of the cost curve for each pixel. The length of the input sequence should be constant during training. For this reason, we have padded the cost curves with its maximum value as shown in Figure 3. The maximum sequence length (the disparity range) we used in our experiments is 255. Moreover, from the initial experiment, we noticed that if the minimum value of the cost curve is 0 then the training is faster and more stable. Therefore, we have subtracted the minimum value of the cost curve from each element. Besides that, a batch size of 1000 and a learning rate of 0.001 are used. We trained our network using RMSprop optimizer [1] with Binary Cross Entropy (BCE) between the estimated confidence Q_i and the ground truth confidence Q_i^* on each sample of the batch. During the training process, we noticed that the estimated confidence map usually shows errors in the region where the true confidence value is 0. This is because the overall data distribution is not uniform. There are more pixels with a confidence value of 1 as compared to pixels having a confidence value of 0 during training. Thus, the network is more inclined towards highly confident pixels. We address this problem by weighting the loss on the basis of the ground truth confidence value Q_i^*. The modified loss is defined as

$$\text{Loss} = -\frac{1}{n}\sum_i W_i \cdot \text{BCE}_i \tag{8}$$

with the weight vector W and BCE being defined as

$$W_i = \begin{cases} 1, & \text{if } Q_i^* = 0 \\ 0.1, & \text{otherwise} \end{cases} \tag{9}$$

$$\text{BCE}(Q_i, Q_i^*) = (Q_i^* \log(Q_i)) + (1 - Q_i^*)(\log(1 - Q_i)) \tag{10}$$

As shown in (9), if the ground truth confidence value of the pixel is 1 then the loss is scaled by 0.1, whereas the loss for the pixel with ground truth confidence value of 0 remains unchanged. In this way, we dealt with the uneven distribution of training data without reducing the number of samples and giving more weight to wrong disparity assignments for training. Hence, weighting of the loss function improves the overall accuracy of the network.

4 Experimental Results

To evaluate the performance of the proposed method, we have compared it with seven state-of-the-art methods, including the hand-crafted confidence measure proposed by Op het Veld *et al.* [33] and learning-based confidence measures such as CCNN [24], multi-modal networks (LFN and EFN) [7], and local-global confidence network (LGC-Net) [32]. We used three completely different datasets (KITTI 2012 [8], Middlebury v3[28], and MPI Sintel [4]) with two state-of-the-art stereo algorithms (MC-CNN and ADCensus) to assess the generalization or adaptability of our proposed network.

The evaluation strategy is explained in Section 4.1. In Sections 4.2 and 4.3, we have presented the quantitative and qualitative results for all three datasets, respectively, justifying the better accuracy achieved by the proposed neural network. In Table 1, the number of trainable parameters for each network is listed. It is evident that the proposed method only requires very few trainable parameters; 3-4 orders of magnitude less compared against state-of-the-art, with a clear increase in accuracy.

4.1 Evaluation Methodology

The sparsification curve and its area under the curve (AUC) is a well-known evaluation strategy used to benchmark the performance of the different confidence measures [15, 26].

[1] This is a part of TensorFlow framework [2].

Dataset	KITTI 2012 [8]		Middlebury v3 [28]	
Raw cost	MC-CNN [34]	ADCensus [21]	MC-CNN [34]	ADCensus [21]
Op het Veld *et al.* [33]	0.0091	0.0976	0.0801	0.1610
CCNN [24]	0.0074	0.0912	0.0764	0.1558
EFN [7]	0.0080	0.0904	0.0913	0.1700
LFN [7]	0.0077	0.0850	0.0791	0.1547
ConfNet [32]	0.0064	0.0912	0.0814	0.1620
LGC-Net (CCNN) [32]	0.0072	0.0845	0.0767	0.1592
LGC-Net (LFN) [32]	0.0065	0.0847	0.0792	0.1595
Proposed C-RNN	**0.0061**	**0.0839**	**0.0677**	**0.1537**
Optimal	0.0031	0.0531	0.0435	0.1101

Table 2: Experimental results on KITTI 2012 and Middlebury v3 datasets. For each row, average AUC achieved on the entire dataset (i.e., 161 out of 194 stereo pairs for KITTI2012 and 15 stereo pairs of Middlebury v3) is listed for different confidence measures. The measures are evaluated for both MC-CNN and ADCensus.

The area under the curve quantifies the accuracy of the method to successfully detect wrong disparity values while discarding only as few correct disparity values as possible, i.e., the lower the AUC the better. AUC is obtained by plotting the bad pixel rate (i.e., $|D_i - D_i^*| > \tau$) as a function of pixel density p sampled from the disparity map and sorted in descending order of the confidence as shown in Figure 4 (b). The optimal condition is achieved when all the pixels with incorrect disparity assignment are removed before rejecting correct ones. The optimal AUC (the theoretically best achievable value) is calculated using the following formula, where ε represents the bad pixel rate [15].

$$\mathrm{AUC}_{\mathrm{opt}} = \int_{1-\varepsilon}^{1} \frac{p-(1-\varepsilon)}{p} dp = \varepsilon + (1-\varepsilon)\ln(1-\varepsilon) \tag{11}$$

4.2 Evaluation on KITTI 2012 and Middlebury v3

For the validation of the proposed method, we trained and tested the network on two different datasets. From KITTI 2012 [8], we used the first 25 stereo pairs as well as the stereo pairs 43, 71, 82, 87, 95, 120, 122, and 180 following the conventional methods [11, 23, 29, 30] for training. The latter eight pairs have relatively more incorrect correspondences compared to the others. The remaining 161 (out of 194) stereo pairs are then used for quantitative evaluation. Similarly, we also trained and tested our neural network on the Middlebury dataset. For training, we used the datasets Middlebury 2005 [27] and 2006 [27] and tested it using Middlebury v3 [28].

In Table 2, average AUC values for both KITTI 2012 and Middlebury v3 test datasets using either MC-CNN or ADCensus for stereo estimation are reported. Our proposed method C-RNN yields the best results, while requiring the training of only few parameters. Figure 4 (a) illustrates the AUC values for Middlebury v3 dataset which are sorted with respect to ascending order of optimal AUC values. Moreover, it is also evident from the qualitative result shown in Figures 1 and 5 that our proposed network is more accurate in detecting outliers, especially in the sky region in Figure 1 as compared to the LGC-Net (LFN) [32].

Dataset	MPI Sintel [4]	
Raw cost	MC-CNN [34]	ADCensus [21]
Op het Veld *et al.* [33]	0.0701	**0.1339**
CCNN [24]	0.0757	0.1546
EFN [7]	0.0854	0.1537
LFN [7]	0.0755	0.1446
ConfNet [32]	0.0706	0.1546
LGC-Net (CCNN) [32]	0.0759	0.1475
LGC-Net (LFN) [32]	0.0734	0.1446
Proposed C-RNN	**0.0660**	0.1372
Optimal	0.0467	0.1223

Table 3: Cross validation results on MPI Sintel dataset. Average AUC obtained on 23 frames of MPI Sintel dataset evaluated for raw cost calculated using MC-CNN and ADCensus algorithm.

4.3 Cross-validation on MPI Sintel

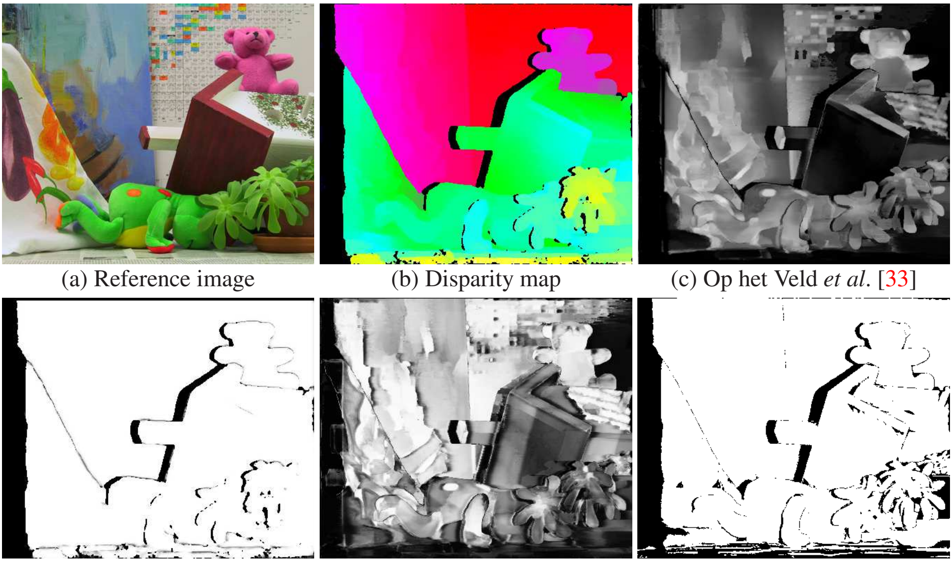
In this section, we have cross-validated our network on the artificial dataset MPI Sintel [4] in order to further demonstrate the ability to generalize the performance on a completely different dataset. This dataset is created using Blender [1]. We obtained confidence maps for the 23 frames in MPI Sintel dataset using our network trained on the Middlebury dataset. The Middlebury dataset consists of indoor scenes which is very different from the scene created using Blender [1] in MPI Sintel. Table 3 reports the average AUC values for all the considered confidence measures including the proposed method. The accuracy of the proposed network outperforms all other learning-based approaches. However, the performance of the hand-crafted method by Op het Veld *et al.* is slightly better for ADCensus. The qualitative result for frame *cave2* is shown in Figure 6 and its sparsification curves for all the CMs are shown in 4 (b). In Table 3, we can see the benefits of a hand-crafted method, as it generalizes more easily, while a learning-based method achieves better results, but must be retrained for every data set.

5 Conclusions

In this paper, we have introduced a novel algorithm for estimating confidence values for stereo disparity estimation. We have developed an LSTM recurrent neural network named C-RNN utilizing the cues from the cost curve for the elimination of the wrong disparity values. The proposed network requires very few trainable parameters as compared to state-of-the-art methods, consisting of a single memory block with a single LSTM cell. The matching cost curve is constructed by a disparity estimation method that already takes into account various priors such as spatial aggregation; the match cost curve resembles the distribution of disparity values for a particular pixel, thus, the distribution is very indicative of the confidence of the most probable disparity value. Moreover, the experimental results on three completely different datasets with two different stereo algorithms clearly show the efficiency and adaptability of the proposed method. In future, we would like to integrate our proposed confidence measure for multi-view-stereo disparity fusion.

6 Acknowledgment

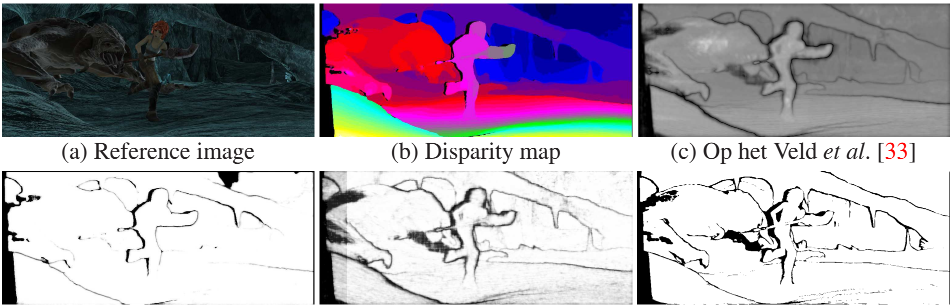
This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 765911 (RealVision).



(a) Reference image (b) Disparity map (c) Op het Veld *et al.* [33]

(d) LGC-Net (LFN) [32] (e) Proposed C-RNN (f) Ground-truth

Figure 5: Visual comparison for "Teddy" from Middlebury v3 [28]. The disparity map is obtained using ADCensus [21]. Brighter values represent a higher confidence.



(a) Reference image (b) Disparity map (c) Op het Veld *et al.* [33]

(d) LGC-Net (LFN) [32] (e) Proposed C-RNN (f) Ground-truth

Figure 6: Visual comparison for "Cave2" from MPI Sintel dataset [4]. The disparity map is obtained using MC-CNN [34]. Brighter values represent a higher confidence.

References

[1] Blender foundation, 2019. URL https://www.blender.org.

[2] Martin Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur,

Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation*, pages 265–283, 2016.

[3] Myron Z Brown, Darius Burschka, and Gregory D Hager. Advances in computational stereo. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (8):993–1008, 2003.

[4] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision*, pages 611–625, 2012.

[5] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European Conference on Computer Vision*, pages 184–199. Springer, 2014.

[6] Geoffrey Egnal, Max Mintz, and Richard P Wildes. A stereo confidence metric using single view imagery with comparison to five alternative approaches. *Image and Vision Computing*, 22(12):943–957, 2004.

[7] Z Fu, M Ardabilian, and G Stern. Stereo matching confidence learning based on multi-modal convolution neural networks. *Representation, Analysis and Recognition of Shape and Motion from Image Data (RFMI)*, 2017.

[8] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[9] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232, 2017.

[10] M Shahzeb Khan Gul and Bahadir K Gunturk. Spatial and angular resolution enhancement of light fields using convolutional neural networks. *IEEE Transactions on Image Processing*, 27(5):2146–2159, 2018.

[11] Ralf Haeusler, Rahul Nair, and Daniel Kondermann. Ensemble learning for confidence measures in stereo vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 305–312, 2013.

[12] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.

[13] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, November 1997.

[14] Asmaa Hosni, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):504–511, 2013.

[15] Xiaoyan Hu and Philippos Mordohai. A quantitative evaluation of confidence measures for stereo vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34 (11):2121–2133, 2012.

[16] Sunok Kim, Dongbo Min, Seungryong Kim, and Kwanghoon Sohn. Feature augmentation for learning confidence measure in stereo matching. *IEEE Transactions on Image Processing*, 26(12):6019–6033, 2017.

[17] Sunok Kim, Dongbo Min, Seungryong Kim, and Kwanghoon Sohn. Unified confidence estimation networks for robust stereo matching. *IEEE Transactions on Image Processing*, 28(3):1299–1313, 2019.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[19] Wenjie Luo, Alexander G Schwing, and Raquel Urtasun. Efficient deep learning for stereo matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5695–5703, 2016.

[20] Larry Matthies. Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation. *International Journal of Computer Vision*, 8(1):71–91, 1992.

[21] Xing Mei, Xun Sun, Mingcai Zhou, Shaohui Jiao, Haitao Wang, and Xiaopeng Zhang. On building an accurate stereo matching system on graphics hardware. In *IEEE International Conference on Computer Vision Workshops*, pages 467–474, 2011.

[22] Philippos Mordohai. The self-aware matching measure for stereo. In *IEEE International Conference on Computer Vision*, pages 1841–1848, 2009.

[23] Min-Gyu Park and Kuk-Jin Yoon. Leveraging stereo matching with learning-based confidence measures. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 101–109, 2015.

[24] Matteo Poggi and Stefano Mattoccia. Learning from scratch a confidence measure. In *British Machine Vision Conference*, 2016.

[25] Matteo Poggi and Stefano Mattoccia. Learning a general-purpose confidence measure based on o (1) features and a smarter aggregation strategy for semi global matching. In *Fourth International Conference on 3D Vision*, pages 509–518. IEEE, 2016.

[26] Matteo Poggi, Fabio Tosi, and Stefano Mattoccia. Quantitative evaluation of confidence measures in a machine learning world. In *IEEE International Conference on Computer Vision*, pages 5228–5237, 2017.

[27] Daniel Scharstein and Chris Pal. Learning conditional random fields for stereo. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[28] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Conference on Pattern Recognition*, pages 31–42, 2014.

[29] Akihito Seki and Marc Pollefeys. Patch based confidence prediction for dense disparity map. In *British Machine Vision Conference*, volume 2, page 4, 2016.

[30] Aristotle Spyropoulos, Nikos Komodakis, and Philippos Mordohai. Learning to detect ground control points for improving the accuracy of stereo matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1621–1628, 2014.

[31] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *Advances in Neural Information Processing Systems*, pages 1988–1996, 2014.

[32] Fabio Tosi, Matteo Poggi, Antonio Benincasa, and Stefano Mattoccia. Beyond local reasoning for stereo confidence estimation with deep learning. In *European Conference on Computer Vision*, pages 319–334, 2018.

[33] Ron Op Het Veld, Tobias Jaschke, Michel Bätz, Luca Palmieri, and Joachim Keinert. A novel confidence measure for disparity maps by pixel-wise cost function analysis. In *IEEE International Conference on Image Processing*, pages 644–648, 2018.

[34] Jure Zbontar, Yann LeCun, et al. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17(1-32): 2, 2016.

[35] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. Part-based r-cnns for fine-grained category detection. In *European Conference on Computer Vision*, pages 834–849. Springer, 2014.