IMPROVING DISPARITY ESTIMATION BASED ON RESIDUAL COST VOLUME AND RECONSTRUCTION ERROR VOLUME

Junhua Kang [1,2,*], Lin Chen[2], Fei Deng [1], Christian Heipke [2]

[1] School of Geodesy and Geomatics, Wuhan University, Wuhan, PR.China, jhkang@whu.edu.cn, fdeng@sgg.whu.edu.cn
[2] Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Germany, (kang, chen, heipke)@ipi.uni-hannover.de

Commission II, WG 2

KEY WORDS: Stereo Matching, Disparity Refinement, Residual Cost Volume, Reconstruction Error

ABSTRACT:

Recently, great progress has been made in formulating dense disparity estimation as a pixel-wise learning task to be solved by deep convolutional neural networks. However, most resulting pixel-wise disparity maps only show little detail for small structures. In this paper, we propose a two-stage architecture: we first learn initial disparities using an initial network, and then employ a disparity refinement network, guided by the initial results, which directly learns disparity corrections. Based on the initial disparities, we construct a residual cost volume between shared left and right feature maps in a potential disparity residual interval, which can capture more detailed context information. Then, the right feature map is warped with the initial disparity and a reconstruction error volume is constructed between the warped right feature map and the original left feature map, which provides a measure of correctness of the initial disparities. The main contribution of this paper is to combine the residual cost volume and the reconstruction error volume to guide training of the refinement network. We use a shallow encoder-decoder module in the refinement network and do learning from coarse to fine, which simplifies the learning problem. We evaluate our method on several challenging stereo datasets. Experimental results demonstrate that our refinement network can significantly improve the overall accuracy by reducing the estimation error by 30% compared with our initial network. Moreover, our network also achieves competitive performance compared with other CNN-based methods.

1. INTRODUCTION

Stereo matching has been investigated for many years and still remains to be a challenging task in photogrammetry and computer vision. The task is to find correspondences, often point-wise, between at least two images, and thus to calculate the disparity of corresponding points between images, which is a pre-requisite for computing 3D coordinates needed in many applications, such as mapping, autonomous driving, robotics and navigation. Traditional stereo matching methods have been well studied and in many cases provide efficient solutions (e.g. Heipke 1997; Haala and Rothermel, 2012); they mostly follow the traditional pipeline, namely matching cost computation, cost aggregation, optimization and disparity refinement (Scharstein and Szeliski, 2002).

In order to obtain sub-pixel accuracy, traditional methods usually employ least-squares matching (Förstner 1984) or some post-processing steps, such as left-right consistency check (Hannah, 1989; Bolles et al., 1993), filtering (Tomasi and Manduchi, 1998), and interpolation operations to refine and improve disparities. However, least-squares matching is known to need rather accurate initial values and ad hoc local post-processing ignores the global image context, which can result in noisy disparity estimation. Moreover, most traditional methods include some hidden assumptions about the geometry of the 3D surface to be reconstructed and thus have limited performance in more challenging scenes, especially for large depth variations and in fine structure areas.

Recently, deep learning techniques have shown powerful capability for stereo matching by using convolutional neural networks (CNN) to solve one or more of the four traditional steps. For example, MC-CNN (Žbontar and LeCun, 2016) was the first to use CNN to learn matching costs between two image patches. However, although this method out-performs some of the traditional approaches, it only focuses on the first step, namely matching cost computation. Several researchers have proposed to learn disparity by integrating all steps into an end-to-end network. DispNet (Mayer et al., 2015) is the first such end-to-end learning framework. It takes rectified stereo images as input and uses a deep encoder-decoder module to directly regress disparities from coarse to fine. Several other CNN-based methods (Chang and Chen, 2018; Kendall et al., 2017) employ 3D convolutional operations on cost volume optimization to aggregate more global context information, which achieves impressive performance. However, the improvements of these end-to-end methods mainly lie in a more global accurate estimation of the scene surface at the cost of losing local structure details. In our prior work (Kang et al., 2019), based on DispNet, we propose a context pyramidal network and introduce a gradient regularizer to preserve small structure detail. This method can estimate clear boundaries in large depth discontinuity areas and is considered the *initial network* in this paper. Nevertheless, when carefully inspecting the output, the predicted disparity still suffers from some local errors, which appear near small objects. This observation motivates us to integrate refinements more explicitly into the whole network for tackling this problem.

* Corresponding author

In this paper we address the problem of preserving details based on the concept of residual learning (He et al., 2016). We add two networks: after the initial network we adopt a second one which is guided by the initial results. We use shared feature maps and derive initial disparities to construct both, a residual cost volume and a reconstruction error volume. We then train a residual network, guided by the residual cost volume and the reconstruction error volume, to learn disparity residuals and estimate the final depth map by adding the learned residuals to the initial disparity. In this way, the refinement sub-net can concentrate on learning more accurate results, especially in problem areas where the initial network fails. Compared to the initial network, the residual cost volume takes into consideration a significantly shorter range of disparity with finer resolution, thus the complexity of learning is lower than learning the disparity for these pixels directly. For this reason, we can employ a shallow encoder-decoder module in our refinement sub-net, and we learn multiple residuals from coarse to fine, which allows our approach to also correct errors and refine details from coarse to fine.

In summary, the contributions of this paper are as follows:

- We propose a new guided refinement network to update the initial disparity estimates by incorporating shared feature maps from the initial network.
- We introduce two interpretable inputs, namely the residual cost volume and the reconstruction error volume as guidance for learning disparity details. These two volumes contain detailed context information and disparity correctness cues, respectively, which provide helpful guidance for disparity refinement.
- We propose a shallow encoder-decoder residual network to fuse guidance information for learning the residuals with explicit supervision at each scale, which is easier than directly learning entire disparity values.

This paper is organized as follows: we review the related work in Section 2 and present the details of our methodology in section 3; followed by experimental results and an analysis in Section 4, before concluding our work in Section 5.

2. RELATED WORK

For a long time stereo matching has continuously been an active research area in photogrammetry and computer vision. Here, we restrict the review to the categories most relevant in our context.

Traditional stereo methods. As mentioned above, most traditional methods follow the classical four-step pipeline. A well-known algorithm of this group is Semi-Global Matching (SGM) (Hirschmuller, 2008). SGM calculates the matching cost using Mutual Information (Viola and Wells III, 1997) and seeks an optimal disparity assignment by combining various 1D optimizations of a global energy function in different directions in image space using dynamic programming. Most global traditional stereo matching approaches typically use post-processing to obtain complete and sub-pixel disparities. For example, many employ the left-right consistency check (Hannah, 1989; Bolles et al., 1993) to detect occlusion areas and fill affected pixels by interpolation. Since these refinement steps typically do not consider global image context, the performance is limited.

Matching cost learning based on CNN. These methods mainly focus on learning matching cost between two image patches using CNN. MC-CNN (Žbontar and Lecun, 2016) is a Siamese network composed of a series of stacked convolutional layers to extract descriptors of each image patch, followed by a simple dot product (MC-CNN-fst) or a number of fully-connected layers (MC-CNN-art) to derive the similarity measure. Luo et al. (2016) expanded MC-CNN and propose a notably faster Siamese network to learn a probability distribution over all possible disparities without manually pairing patch candidates. Li and Yu (2018) introduced dilated convolutions to enlarge the receptive field of view when computing the matching cost. These patch based methods indeed outperform most traditional stereo algorithms. However, they still require subsequent heuristic steps, including cost optimization to produce complete results.

End-to-End disparity learning without refinement. Approaches in this category normally develop a fully learnable architecture without any further refinement processing, regressing disparity by training the whole network end-to-end. DispNet (Mayer et al., 2015) was the first end-to-end network for stereo matching, which has a structure similar to that of FlowNet (Dosovitskiy et al., 2015). DispNet utilizes a deep encoder-decoder architecture for disparity regression, has achieved prominent performance and has become a baseline network in stereo matching. Following the same basic architecture, GC-Net (Kendall et al., 2017) employs 3D convolutions for cost volume regularization to incorporate more context, and finally regresses the disparity through a differentiable "soft-argmin" operation. Similar to GC-Net, PSM-Net (Chang and Chen, 2018) uses spatial pyramid pooling and 3D convolutions to capture global context on different scales. However, employing high-dimensional features based on 3D convolution is computationally expensive. Instead of using 3D convolutions, Kang et al. (2019) introduced dilated convolutions to exploit multi scale context cues and proposed a gradient regression loss for regularizing disparity changes in a supervised way, which can preserve local detail in depth discontinuity areas.

End-to-End disparity learning with refinement. In this category, disparity refinement has been taken into consideration in CNN approaches. In the so called DRR (detection, replacement and refinement) approach (Gidaris and Komodakis, 2017) two sub-networks are used to detect initial errors and replace large mistakes with new values in the initial disparities, before refining minor errors by using an additional sub-network. In a similar way, DispNet_css (Ilg et al., 2018) combines three separate networks (each of them similar to DispNet) with residual connections to refine disparities. Jie et al. (2018) integrate the left-right consistency check as soft guidance into a recurrent neural network to refine unreliable disparities. Batsos and Mordohai (2018) also use a recurrent refinement network to learn different types of errors by combining residuals in different scales. However, recurrent neural network are difficult to train. Most recently, ResDepth, a deep network (Stucker and Schindler, 2020) was proposed to improve the depth map for high-quality dense stereo reconstruction. The inputs of this network are the initial depth map and the warped images, and a standard U-net is used to learn residuals.

The work most closely related to our work is CRL (Pang et al., 2018). This is a cascade residual learning network, which stacks an advanced DispNet and a residual network to learn residuals

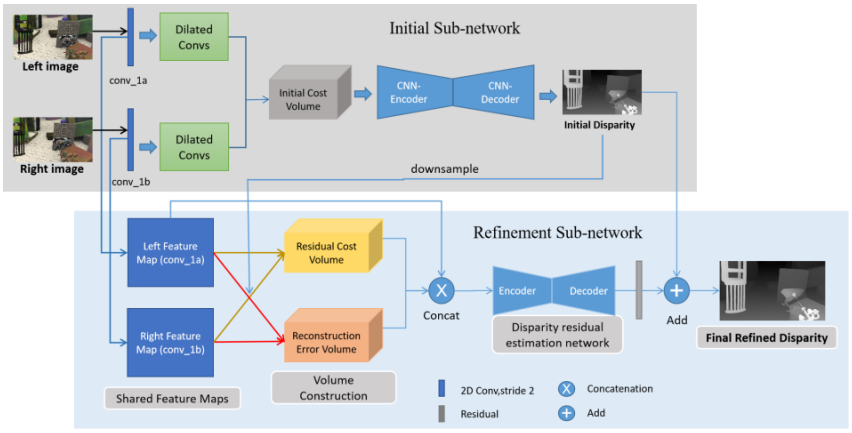


Figure 1. The overall architecture of our network. It consists of two parts, the initial sub-net and the refinement sub-net. The inputs are two rectified images and the output is the final refined disparity map

between the coarse initial disparity and the ground truth disparity and thus to explicitly refine the initial disparity. The inputs of this residual learning network are two original images, a warped right image, an error map and an initial disparity, which is somewhat redundant information. While stacking more networks together indeed improves accuracy, it lacks interpretability and leads to expensive computations.

In contrast, in our approach, we propose a residual cost volume and a reconstruction error volume, which we argue can be better interpreted as inputs for residual learning. The residual cost volume contains detailed information of the correlation between the two images and the reconstruction error volume reflects cues of uncorrected disparities, which are helpful to guide our network to learn accurate residuals. In addition, we use shared feature maps instead of using original images in the refinement sub-net, which reduces the number of learning parameters.

3. METHODOLOGY

3.1 Overall network architecture

The structure of the solution we propose is depicted in Fig. 1. In this study, the goal is to improve initial disparity quality by adding a refinement step to an end-to-end network. From Fig. 1, it can be observed that we cascade this refinement network as a sub-network to our initial CNN-based stereo matching network. We use the lower levels of our shared feature maps and disparities, both of which come from the initial network, as input for the refinement. The output is the final refined disparity. The initial network results in a pixel-wise disparity map for a pair of rectified stereo images. For more details about the initial network, please refer to our prior work (Kang et al., 2019). In this work, we focus on the details of the proposed refinement part.

3.2 Residual cost volume construction

In our initial sub-net, we learn the initial disparity from the initial cost volume, which is constructed between feature maps after several convolutional layers (see Fig. 1). These layers are necessary to increase the receptive field of view and capture more global context, but they lead to loosing small structures and reducing the spatial resolution of the feature maps. Therefore, in the initial cost volume, some detail will be lost.
In our refinement sub-net, we first take two shared feature maps F_L, F_R from the first convolutional layer of the initial sub-net as inputs, which provide enough local context information. Then, we construct a residual cost volume $C_{residual}$between these two feature maps at this fine resolution to capture more detailed correlation information. The basic idea behind the residual cost volume construction is shown in Fig. 2: for a pixel x_L on the left feature map F_L, let the initial disparity be d_L. The corresponding initial matching point $x_R^0 = x_L + d_L$ is calculated using the initial disparity d_L . However, the initial disparity d_L is imprecise and the correct corresponding point to x_L is be calculated as follows:

$$x_R^{\Delta d} = x_L + d_L + \Delta d \quad (1)$$

where $x_R^{\Delta d}$ is the corresponding point in the right feature map; $\Delta d \in [-d_{offset}, d_{offset}]$ represents the disparity residual interval. In this paper, we use sub-pixel steps (0.5 pixels) within this interval to obtain sub-pixel accuracy; d_{offset} reflects the accuracy of initial matching and must be given a priori. For every residual Δd, the matching cost feature map $C^{\Delta d}$ is created by convolving the left and right feature maps. The correlation of two patches centred at x_L in F_L and $x_R^{\Delta d}$ in F_R is defined as:

$$C^{\Delta d}\left(x_L, x_R^{\Delta d}\right) = \sum_{o\in[-k,k]\times[-k,k]}[F_L(x_L + o) \otimes F_R(x_R + o)] \quad (2)$$

where k is an index, $K = 2k + 1$ is the patch size and $\otimes$ denotes the convolution operation. Then, as Fig. 2 shows, the final residual cost volume $C_{residual}$ is constructed by concatenating all cost feature maps across the disparity residual interval. In this way, the refinement network can learn correct disparity residuals by using the residual cost volume as guiding information. As F_L and F_R are derived after the first convolutional layer with stride 2, the size of the residual cost volume is $1/2\,W \times 1/2\,H \times C_r$, where W, H representing the width and height of the original image, and C_r is the number channels of $C_{residual}$.

3.3 Reconstruction Error Volume Construction

As the relative orientation of the two images is known, a warped version of the right image can be reconstructed by texture remapping with the corresponding disparity map. The warping technique has been in use in photogrammetry for a long time (Norvelle, 1992) and is also employed for processing of the Mars HRSC images (Schmidt, 2008). In the ideal case, the left image and the warped right image are identical in non-occluded areas, and the difference of conjugate grey values is 0. If, on the other hand, this difference is large, the estimated disparity is more likely incorrect or stems from occluded areas. Thus, this difference, called reconstruction error here, provides cues of how to improve the disparity.

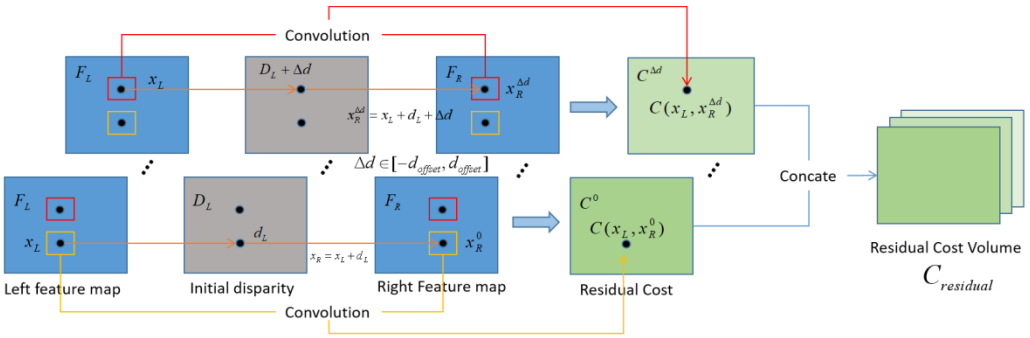


Figure 2. Construction process of the residual cost volume. The red and yellow boxes represent context windows of different points.

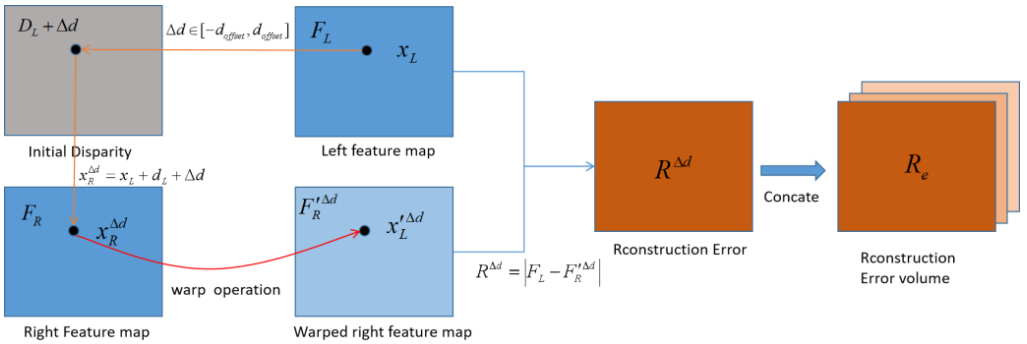


Figure 3. Construction process of the reconstruction error volume

In our refinement part, we compute the reconstruction error volumes between the left feature map and the warped right feature map for the disparity residual interval. The construction process is shown in Fig. 3: for the left feature map F_L, using the initial disparity map D_L and the residual value Δd, we can reconstruct a warped right feature map $F'^{\Delta d}_R$ by remapping pixels from the right feature map. Then the reconstruction error can be obtained by calculating the absolute difference:

$$R_{\Delta d} = \left| F_L - F'^{\Delta d}_R \right| \tag{3}$$

where $R_{\Delta d}$ is the reconstruction error map, which measures the correctness of disparity in feature space. Similar to the construction of the residual cost volume, we concatenate all reconstruction error maps along the disparity residual interval and obtain the reconstruction error volume R_e. This volume is also a crucial factor for guiding the refinement network.

3.4 Disparity residual estimation network

After calculating the residual cost volume and the reconstruction error volume, we concatenate these two volumes and the left feature map as inputs into the disparity residual estimation network. Fig. 4 shows its basic architecture.

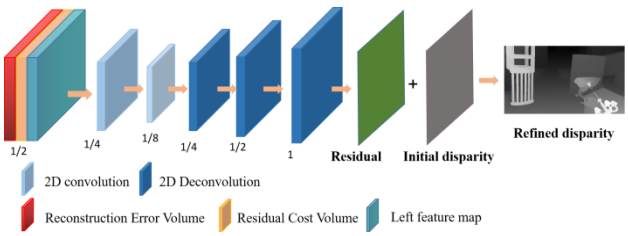


Figure 4. Disparity residual estimation network

Since we employ the shared feature maps from the initial sub-net, we do not have to extract features again from the original images, thus the refinement sub-net can be designed with less layers. As Fig.4 shows, we use a shallow encoder-decoder architecture to recover disparity details from coarse to fine. Instead of directly learning disparity values for every pixel, we chose to learn disparity residuals, which is easier. Only two groups of convolutional layers are stacked in the encoder to preserve more spatial context. Each group contains two 3 × 3 convolutions with strides 2 and 1, respectively, achieving an encoded feature map with dimension (W⁄8×H⁄8×C) where W, H, represent the width and height in original resolution, and C represents the number of channels of the feature map. In order to obtain dense per-pixel predictions with the original input resolution, we apply three up-sampling blocks corresponding to four scales (1/8, 1/4, 1/2 and 1 × the original size) in the decoder part. Each block consists of a 4 × 4 deconvolution layer with stride 2 to up-sample the residual output map. The network outputs refined disparities in different scales by adding the learned residuals with the initial disparity:

$$D^s_{refined} = D^s_{initial} + R^s \tag{4}$$

where $D^s_{refined}$ represents the refined disparity map in s scale ($s \in 1/8, 1/4, 1/2, 1$), $D^s_{initial}$ is the initial disparity and R^s means the residual map in s scale.

3.5 Loss

We train our network in a fully supervised manner by using a disparity regression loss. We adopt the ℓ_1 norm to measure the absolute difference between the disparity D predicted by the model and the ground truth disparity $\hat{\mathrm{D}}$. As ground truth disparity maps are sometimes sparse (e.g. KITTI dataset, see Geiger et al. (2012; Menze et al., 2018), we average our loss over the valid pixels N_v, for which ground truth labels are available. Thus the loss function for the scale s is defined as:

$$\mathcal{L}_s = \frac{1}{N_v} \sum_{i,j} \left\| D_{i,j} - \hat{D}_{i,j} \right\|_1 \tag{5}$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm, v represents all valid disparity pixels in $\widehat{D}$ and N_v is the number of valid pixels. The total loss function E is a weighted sum of losses of all scales:

$$E = \sum_s \lambda^s \mathcal{L}_s \quad (6)$$

where $\mathcal{L}_s$ is the loss from Eq. (5), evaluated at scale s, and λ^s denotes the weighting factor for this scale.

4. EXPERIMENTS AND RESULTS

4.1 Dataset

In this work, we have carried out several experiments to assess the performance of our method in a qualitative and quantitative way. A number of public synthetic and real stereo datasets are used: Scene Flow (Mayer et al., 2015), MPI Sintel (Wulff et al., 2012) and KITTI 2015 (Menze et al., 2018; Menze and Geiger, 2015), which all contain rectified stereo images and ground truth disparity. Scene Flow is a large synthetic dataset and provides accurate sup-pixel dense ground-truth disparities; it contains more than 39,000 stereo frames in 960 × 540 pixel resolution. We use it to train our network end-to-end. MPI Sintel is also an entirely synthetic dataset, which has 1064 training frames in 1024 × 436 pixel resolution and provides dense ground truth with large displacement. We use it to test the performance of our pre-trained model. The KITTI 2015 dataset is a real world dataset and contains various outdoor street views captured from a car driving in an urban area. It provides about 200 stereo pairs in 1242 × 375 pixel resolution for training with sparse ground truth obtained from a 3D laser scanner; only approximately 30% of pixels have ground truth disparity values.

4.2 Implementation details

Training: The Tensorflow framework is used in our work and all experiments are conducted on a Titan X GPU. We optimized our model end-to-end by choosing the Adam optimizer with default momentum parameters, β1 = 0.9 and β2 = 0.999. We trained our model on the Scene Flow dataset in three stages. First, we trained the initial sub network (for training details, refer to our prior work, Kang et al. (2019). Then we fixed the parameters of the initial sub-network and trained the refinement sub-net with a learning rate of 1e-4 for the first 80k iterations and 1e-5 for the remaining 120k iterations. Finally, we jointly refined the whole network with a learning rate of 1e-5 for the first 80k and 5e-6 for the remaining 120k iterations. We used fixed weights for the different scales in the loss function during training; these weights $(\lambda^1, \lambda^2, \lambda^3, \lambda^4)$ were set to (1, 0.5, 0.2, 0.2). We fine-tuned the pre-trained model on the KITTI 2015 training dataset with a learning rate of 1e-5 for 20k iterations. Due to the GPU limitation, we set the batch size to 2 for training.

Testing: we evaluate our model on different datasets with two metrics. One is the End-point-Error (EPE), which calculates the average Euclidean distance between predicted and ground truth disparity along all valid pixels. The other one is t-pixel error, which computes the percentage of "bad" pixels among all valid pixels. A bad pixel is a pixel with an absolute disparity error larger than a threshold t.

4.3 Error analysis for the initial network

By way of example, we first investigate the error distribution for the initial network to obtain the disparity error range, which is a very important factor to guide the refinement. To do so, we analyse the empirical error distribution of the initial disparity prediction (excluding the disparities in occluded areas) on the training samples of the well-known Scene Flow dataset (Mayer et al., 2015), see Fig. 5. From this figure, it can be observed that small errors occur with much higher probability than larger errors. We also provide the results in logarithmic scale to better show the percentage of large errors.

From Fig. 5, we observe that in this example 95% of the initial disparity errors are smaller than 2.3 pixels and 99% are smaller than 7.8 pixels. The distribution reveals that the majority of errors of the initial disparity can be interpreted as random errors, rather than as systematic or gross errors. Therefore, under the assumption that these results are representative, only considering a limited range of disparity in the residual network is a meaningful option. According to the statistical confidence theory, we set the potential residual range of disparity $d_{residual}$ as 10 pixels. This factor is used in the part of constructing the residual cost volume and reconstruction error volume.

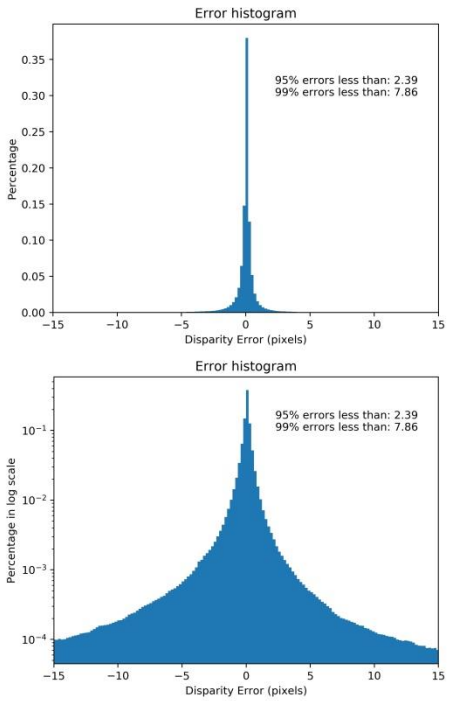


Figure 5. Disparity error distribution of the initial network in basic scale and logarithmic scale of the frequency, respectively.

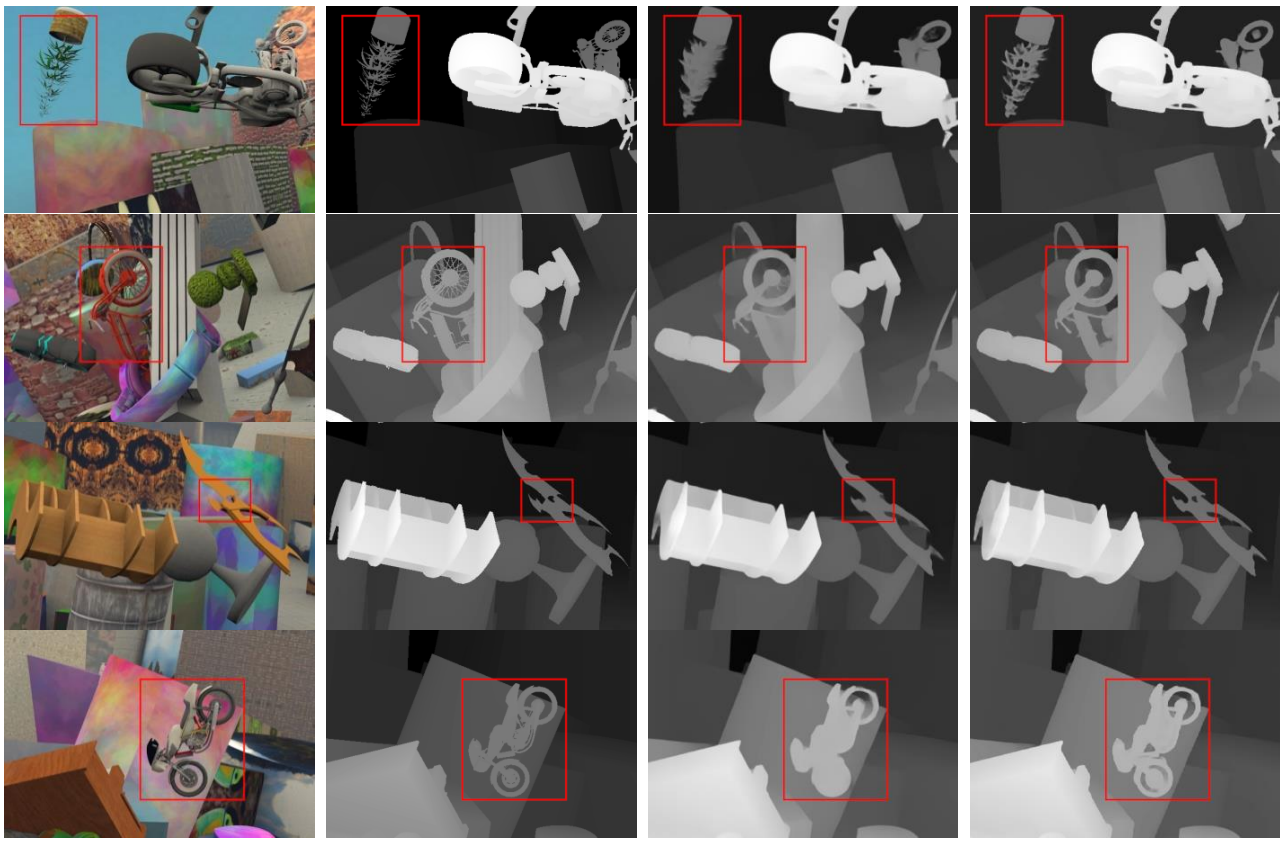
4.4 Results

4.3.1 Ablation experiments: In this section, in order to explore the effectiveness of our refinement sub-net, we compare the results on the Scene Flow dataset and the Sintel dataset when varying the refinement network structures. As listed in table 1, we use "DispNetC" as our baseline network; "DispGradNet" is our initial network which is modified based on the baseline network. "RCV" represents the residual cost volume module and "REV" means the reconstruction error volume in the refinement part. "Joint refinement" means we jointly trained the initial and the refinement network together.

As shown in Table 1, our initial network (Model_1) outperforms the baseline network with the EPE being reduced from 1.68 to 1.43 for Scene Flow, and from 5.66 to 3.06 for MPI Sintel. It can also be seen that our initial end-to-end network can predict more accurate initial disparities than the baseline network. To demon-

Name	Network setting				Test Datasets							
	Initial Sub-net	Refinement Sub-net			Scene Flow				MPI Sintel			
		RCV	REV	Joint refinement	>1px [%]	>3px [%]	>5px [%]	EPE [px]	>1px [%]	>3px [%]	>5px [%]	EPE [px]
Model_0	DispNetC	--	--	--	23.33	9.45	6.22	1.68	47.84	22.90	17.47	5.66
Model_1	DispGradNet	--	--	--	19.36	7.86	5.19	1.43	31.58	14.35	9.45	3.06
Model_2	DispGradNet	√	×	×	13.31	6.45	4.55	1.20	19.67	11.33	8.30	2.68
Model_3	DispGradNet	√	√	×	12.17	6.04	4.29	1.14	**17.81**	**10.35**	**7.63**	**2.58**
Model_4	DispGradNet	√	√	√	**10.77**	**5.28**	**3.76**	**1.02**	18.01	10.83	8.10	2.79

Table 1. Results achieved on the Scene Flow dataset and Sintel dataset when employing different network structures.



(1) Left image (2) Ground Truth (3) Initial Network (4) Our final model

Figure 6. Visualization results of disparity estimation for Scene Flow. Colum 1: Left image; Colum 2: ground truth; Colum 3: results predicted by the initial network (without the refinement). Colum 4: results predicted by our final model (with the refinement).

strate the effectiveness of the residual cost volume, we compared the results with and without this module (Model_1 vs. Model_2). As shown in Table 1, adding the residual cost volume achieves an improvement for EPE for Scene Flow; similar results were obtained for MPI Sintel. This demonstrates that using the residual cost volume indeed improves disparity qualities.

As introduced in Section 3.4, we also employ the reconstruction error volume as another additional guidance for the refinement sub-net. Comparing the results for Model_2 and Model_3 in Table 1, considering this volume leads to better results. Thus, the reconstruction error volume can indeed provide cues for erroneous areas, the results of which are subsequently improved.

In addition, jointly refining the whole network (Model_4 in Table 1) can further slightly improve the results for Scene Flow, but does not do so for MPI Sintel. Thus, some generalization abilities are lost. In summary, compared with the initial network, our refinement sub-net can decrease the estimation error by about 30%, which we consider to be significant.

We also show visualization results regarding the initial network and the refinement network on the Scene Flow dataset, see Fig. 6. Although this dataset is synthetic, the images for evaluation are still very challenging due do the presence of occlusions and thin structures. Compared with the initial network, as illustrated in the red box, in the small structure area, our method can recover richer detail. It can be seen that our refinement sub-net can significantly correct errors of the initial disparity and produce consistent disparities in homogeneous regions.

4.3.2 Comparison with other methods: In this section, we investigate how well our method performs when compared with some state of art methods on the Scene Flow dataset, namely DispNet (Mayer et al., 2015), DispNet_css (Ilg et al., 2018), CRL (Pang et al., 2018), PSM-Net (Chang and Chen, 2018), GC-NET (Kendall et al., 2017), iResNet (Liang et al., 2018) and StereoNet (Khamis et al., 2018). The results are shown in Tab.2.

As is shown in the Tab. 2, the end point error (EPE) of our method is 1.02 pixels, which is the smallest of all values.

	DispNetC	DispNet_css	CRL	PSM-Net	GC-NET	iResNet	StereoNet	Ours
EPE [px]	1.68	1.34	1.32	1.09	2.51	1.40	1.10	**1.02**
>3PX [%]	9.45	7.73	6.20	--	7.20	**4.57**	--	5.36

Table 2 Comparison results of different stereo matching methods on the Scene Flow dataset.

Method	**All pixels**			**Non-occluded pixels**			**Runtime**
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all	
M2S_CSPN (Cheng et al., 2018b)	1.51	**2.88**	**1.74**	1.40	**2.67**	**1.61**	1000 ms
GANet-deep (Zhang et al., 2019)	**1.48**	3.46	1.81	**1.34**	3.11	1.63	1800 ms
PSM-Net	1.86	4.62	2.32	1.71	4.31	2.14	410 ms
CRL	2.48	3.59	2.67	2.32	3.12	2.45	470 ms
GC-Net	2.21	6.16	2.87	2.02	5.58	2.61	900 ms
DispNetC（baseline）	4.32	4.41	4.34	4.11	3.72	4.05	60 ms
Initial Net	3.61	7.14	4.20	3.41	6.59	3.94	140 ms
Our Refinement	2.83	6.88	**3.51**	2.61	5.99	**3.17**	230 ms

Table 3 Comparison results of our model with other methods on KITTI2015 benchmark

As mentioned before, CRL is the method closest to ours, however, the authors simply stack two networks on top of each other to learn the refined disparity map. From the comparison, we see that our method outperforms CRL in terms of EPE and 3-pixel error. The main reason is probably that the initial and refinement sub-nets of CRL are loosely stacked, making it more difficult for the network to learn refined disparities. However, our method uses two interpretable inputs to capture detailed correspondences and error information, which makes our refinement network focus on learning accurate residuals. We also notice that iResNet achieves the best performance in terms of 3-pixel error and that of our result is slightly larger. Compared to our method, the authors of iResNet used multi-scale feature maps in their network to calculate reconstruction error maps and employ iterative strategies, which they argue are effective for improving accuracy in terms of 3-pixel error.

4.3.3 Fine-tuning on KITTI 2015 datasets: Furthermore, we randomly split the whole training set of KITTI2015 into the training subset (90%) and validation subset (10%) and fine-tuned our network on the training subset. Note that we have excluded one image pair from the KITTI2015 training dataset since its illumination condition is very black and thus not representative.

We then submitted the results to the KITTI online leader board for performance evaluation. The results are shown in Tab. 3. “D1-bg” means the 3-pixel error in the background and “D1-fg” means the 3-pixel error in the foreground. “D1-all” represents the 3-pixel error for all pixels. From Tab. 3, we can see that the 3-pixel error for all pixels of our method is 3.51%, which outperforms DispNet (4.34%) and the initial network (4.20%). This means, that adding the refinement part improves the performance on KITTI as well. Compared to current state-of-the- art methods in terms of speed, our method can predict disparity faster, being almost two times faster than CRL. However, our method is still slightly inferior to CRL in terms of accuracy, especially in the foreground of the image. This may be because in our initial network, we employ a disparity gradient regression loss to regularize disparity change, which requires the dataset to have dense ground truth. As the ground truth labels of KITTI are sparse, it is impossible to obtain accurate ground truth disparity gradients.

5. CONCLUSION

In this paper, we propose a new refinement network to estimate a detailed disparity map from stereo images, which incorporates a residual cost volume and a reconstruction error volume as guiding information. The residual cost volume provides more detailed correspondence information between the left and right image and the reconstruction error volume reflects the correctness of initial disparities; both are helpful to guide the network to improve disparity quality. Using these two volumes and the shared features as the inputs, the refinement network adopts a shallow encoder-decoder module to learn disparity residuals and output the final refined disparity map. Extensive qualitative and quantitative experiments on different datasets demonstrate that our refinement network can significantly reduce the disparity error and predict fine structures. Compared with state-of-the-art stereo matching methods, our method can achieve competitive performance if datasets provide dense ground truth, however, has limited accuracy in terms of 3-pixel error. This limitation may be mitigated by adding an iterative refinement.

In future work, we will focus on evaluating our network with different hyper-parameters (e.g., the parameter for sub-pixel accuracy) and refinement strategies. Furthermore, we also plan to employ our refinement network to predict dense depth for high resolution aerial or satellite image datasets. Finally, we strive to adapt our network to multi-view stereo matching, which is essential for dense 3D reconstruction.

ACKNOWLEDGEMENTS

The author Junhua Kang would like to thank the China Scholarship Council (CSC) for financially supporting her study at the Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Germany, as a visiting PhD student. Furthermore, we gratefully acknowledge the support of NVIDIA Corporation for the donation of GPUs used for this research

REFERENCE

Batsos, K., Mordohai, P., 2018. Recresnet: A recurrent residual cnn architecture for disparity map enhancement, in: 2018 Int. Conference on 3D Vision (3DV). IEEE, 238–247.

Bolles, R., Woodfill, J., Kanade, T., Paul, R., 1993. Spatiotemporal consistency checking of passive range data, in: Robotics Research: The Sixth International Symposium. International Foundation for Robotics Research, 165–183.

Chang, J.-R., Chen, Y.-S., 2018. Pyramid stereo matching network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 5410–5418.

Dosovitskiy, A., Fischery, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Smagt, P. Van Der, Cremers, D., Brox, T., 2015. FlowNet: Learning optical flow with convolutional networks. Proc. IEEE Int. Conf. Comput. Vis. 2015, 2758–2766.

Förstner, W., 1984. Quality assessment of object location and point transfer using digital image correlation techniques, The International Archives of Photogrammetry and Remote Sensing, (25) A3a, 197-219.

Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference On. IEEE, 3354–3361.

Gidaris, S., Komodakis, N., 2017. Detect, replace, refine: Deep structured prediction for pixel wise labeling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5248–5257.

Haala, N., Rothermel, M., 2012. Dense Multi-Stereo Matching for High Quality Digital Elevation Models, in: PFG Photogrammetrie, Fernerkundung, Geoinformation (4) 2012, 331-343.

Hannah M.J., 1989: A system for digital stereo image matching. Photogrammetric Engineering & Remote Sensing (55)12, 1765-1770.

Heipke C., 1997: Automation of interior, relative and absolute orientation. ISPRS J. Photogramm. Remote Sens. (52) 1, 1-19.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings, IEEE Conference on Computer Vision and Pattern Recognition, 770–778

Hirschmuller, H., 2008. Stereo processing by semiglobal matching and mutual information. IEEE Trans. Pattern Anal. Mach. Intell. 30, 328–341.

Ilg, E., Saikia, T., Keuper, M., Brox, T., 2018. Occlusions, Motion and Depth Boundaries with a Generic Network for Disparity, Optical Flow or Scene Flow Estimation BT - Computer Vision – ECCV 2018, in: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Eds.), . Springer International Publishing, Cham, 626–643.

Jie, Z., Wang, P., Ling, Y., Zhao, B., Wei, Y., Feng, J., Liu, W., 2018. Left-right comparative recurrent model for stereo matching, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3838–3846.

Kang, J., Chen, L., Deng, F., Heipke, C., 2019. Context pyramidal network for stereo matching regularized by disparity gradients. ISPRS J. Photogramm. Remote Sens. 157, 201–215.

Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A., 2017. End-to-End Learning of Geometry and Context for Deep Stereo Regression. Proc. IEEE Int. Conf. Comput. Vis. 2017-Octob, 66–75.

Khamis, S., Fanello, S., Rhemann, C., Kowdle, A., Valentin, J., Izadi, S., 2018. StereoNet: Guided hierarchical refinement for real-time edge-aware depth prediction. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 11219 LNCS, 596–613.

Li, Z., Yu, L., 2018. Compare Stereo Patches Using Atrous Convolutional Neural Networks, in: Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval. ACM, 473–480.

Liang, Z., Feng, Y., Guo, Y., Liu, H., Chen, W., Qiao, L., Zhou, L., Zhang, J., 2018. Learning for disparity estimation through feature constancy, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2811–2820.

Luo, W., Schwing, A.G., Urtasun, R., 2016. Efficient deep learning for stereo matching, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 5695–5703.

Mayer, N., Ilg, E., Häusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T., 2015. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation 4040–4048.

Menze, M., Geiger, A., 2015. Object scene flow for autonomous vehicles, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3061–3070.

Menze, M., Heipke, C., Geiger, A., 2018. Object scene flow. ISPRS J. Photogramm. Remote Sens. 140, 60–76.

Norvelle, R.R., 1992: Stereo correlation: window shaping and DEM corrections. PE&RS (58) 1, 111-115.

Pang, J., Sun, W., Ren, J.S.J., Yang, C., Yan, Q., 2018. Cascade Residual Learning: A Two-Stage Convolutional Neural Network for Stereo Matching, in: Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017. 878–886.

Scharstein, D., Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vis. 47, 7–42.

Schmidt, R. 2008. Automatische Bestimmung von Verknüpfungspunkten für HRSC-Bilder der Mars Express-Mission. Thesis. DGK-C 623, Beck Verlag, München.

Stucker, C., Schindler, K., 2020. ResDepth: Learned Residual Stereo Reconstruction. arXiv Prepr. arXiv2001.08026.

Tomasi, C., Manduchi, R., 1998. Bilateral filtering for gray and color images, in: Sixth Int. Conf. on Computer Vision (IEEE Cat. No. 98CH36271). IEEE, 839–846.

Viola, P., Wells III, W.M., 1997. Alignment by maximization of mutual information. Int. J. Comput. Vis. 24, 137–154.

Wulff, J., Butler, D.J., Stanley, G.B., Black, M.J., 2012. Lessons and insights from creating a synthetic optical flow benchmark, in: European Conference on Computer Vision. Springer, 168–177.

Žbontar, J., Lecun, Y., 2016. Stereo matching by training a convolutional neural network to compare image patches. J. Mach. Learn. Res. 17, 2287–2318.