remote sensing

MDPI

Letter

A Unified Framework for Depth Prediction from a Single Image and Binocular Stereo Matching

Wei Chen, Xin Luo *, Zhengfa Liang, Chen Li, Mingfei Wu, Yuanming Gao and Xiaogang Jia

College of Computer, National University of Defense Technology, Changsha 410073, China; chenwei@nudt.edu.cn (W.C.); liangzhengfa10@nudt.edu.cn (Z.L.); lichen14@nudt.edu.cn (C.L.); wumingfei10@nudt.edu.cn (M.W.); hylu@nudt.edu.cn (Y.G.); jiaxiaogang@nudt.edu.cn (X.J.)

* Correspondence: luoxin13@nudt.edu.cn; Tel.: +86-0731-8702-3510

Received: 24 December 2019; Accepted: 6 February 2020; Published: 10 February 2020

Abstract: Depth information has long been an important issue in computer vision. The methods for this can be categorized into (1) depth prediction from a single image and (2) binocular stereo matching. However, these two methods are generally regarded as separate tasks, which are accomplished in different network architectures when using deep learning-based methods. This study argues that these two tasks can be achieved using only one network with the same weights. We modify existing networks for stereo matching to perform the two tasks. We first enable the network capable of accepting both a single image and an image pair by duplicating the left image when the right image is absent. Then, we introduce a training procedure that alternatively selects training samples of depth prediction from a single image and binocular stereo matching. In this manner, the trained network can perform both tasks and single-image depth prediction even benefits from stereo matching to achieve better performance. Experimental results on KITTI raw dataset show that our model achieves state-of-the-art performances for accomplishing depth prediction from a single image and binocular stereo matching in the same architecture.

Keywords: depth prediction; binocular stereo matching; network architecture

1. Introduction

Effective methods have long been pursued in various disciplines that can extract or estimate accurate depth information from camera images. Depth information is the distance from images to objections. It plays import roles in modeling 3D environments. Therefore, accurate depth information is critical in many vision-related applications, e.g., autonomous driving [1], augmented reality [2], mixed reality [3], etc.

Some methods use sensors, e.g., laser radar [4] or structured light cameras [5], to measure depth directly. These methods are expensive and are highly dependent on the environment. Direct methods are limited to specific scenes, and most of them obtain sparse depth maps [6], which need further completion before being usable. Depth information can also be estimated from camera images [7,8]. Traditional methods extract various features from camera images and aggregate these features to map from images to depth information [9,10]. However, these methods often obtain blurry and inaccurate results.

Deep-learning-based methods have been widely used to estimate depth from images. Most of these methods are categorized into two classes, namely (1) depth prediction from a single image [11–14] and (2) binocular stereo matching [15–18]. Depth prediction from a single image generates a depth map for a single-view image, whereas binocular stereo matching takes as input a rectified image pair and outputs its disparity map. Note that depth z and disparity d can be transformed mutually

according to Equation (1), where f is the focal length of the camera and B is the distance between the camera centers.

$$z = \frac{fB}{d} \tag{1}$$

Recently, fundamental innovations have been made to benefit one task from the other. In the context of these innovations, stereo matching methods are used to enhance monocular methods [19]. Stereo cues help monocular methods to eliminate the influence of few dense ground-truth depth data. Considering depth z and disparity d can be transformed mutually, we can estimate disparity for both tasks uniformly, with f and B known. However, two research challenges remain before binocular stereo matching and depth prediction from a single image can be accomplished in a unified framework:

- Most deep learning frameworks take as input a fixed number of inputs. However, the number of input images is different for these two tasks.
- A single framework can perform well in one specific task. However, it is hard to guarantee the performance for another task. In other words, it is nontrivial to ensure the framework is optimized towards both tasks [20,21].

To tackle these challenges, an appropriate solution should be able (1) to effectively handle a different number of input images and, at the same time, (2) to unify stereo matching and depth prediction from a single image in the same architecture to ensure that the framework is trained towards both tasks. This study develops a unified framework for both depth prediction from a single image and binocular stereo matching, namely DoubleNet. The framework incorporates a module to handle different types of inputs and a unified architecture for both tasks. This study also introduces a novel training procedure to optimize the unified architecture. The proposed framework can accomplish stereo matching and depth prediction from a single image in the same architecture with the same parameters.

To evaluate the performance of DoubleNet, this study carried out a number of experiments on the challenging KITTI Raw dataset [22]. Experimental results demonstrate that the proposed method can perform both depth prediction from a single image and binocular stereo matching tasks simultaneously and achieves state-of-the-art performance. Moreover, single-image depth estimation benefits from stereo matching by achieving better performance than when treating these two tasks separately.

The main contributions of this study are three-fold:

- A solution has been fostered to accomplish depth prediction from a single image and binocular stereo matching simultaneously.
- This study explores the interaction between monocular and stereo methods. It proposes to make single-image depth estimation benefits from stereo matching.
- A number of experiments have been conducted to prove the effectiveness of the proposed unified framework. Experimental results have shown its performance.

2. Related Work

Successful attempts have been made to improve the performance of depth estimation from camera images, which has long been a notoriously ill-posed task. Studies undertaken for this purpose mainly focuses on (1) estimating the disparity for a pair of stereo images or (2) inferring the depth map for a single image.

2.1. Binocular Stereo Matching

Recently, deep learning methods are gaining popularity in stereo matching. Early work employs deep convolution neural networks (CNN) to calculate the matching cost between left and right images [23]. Its following methods try to extract better features from images [24], to aggregate different features [25], and to use various techniques to refine the estimation [17]. With the emergence of a

large scale of synthetic dense data, a great number of end-to-end methods are proposed for disparity estimation from camera images. These methods can roughly be categorized into two classes:

- In the first class of methods, the correlation between left and right feature maps is used to form a cost volume. The formed cost volume is processed by a series of convolution and trans-convolution layers, i.e., an encoder–decoder-like structure. DispNetC [15] is the first to suggest this paradigm. Built on top of DispNetC, more proposals were proposed, e.g., CRL [26], iResNet [17], DispNet3 [27], etc. Additionally, some similar networks used cues from edges or segmentation to enhance the estimation, e.g., EdgeStereo [28] and SegStereo [29].
- In the second class of methods, the concatenation or difference of left and right features are used to form a 3D cost volume. From this 3D cost volume, 3D CNN layers are employed to extract disparity information. Typical architectures along this direction include GC-Net [16] and PSMNet [18].

Regardless of the different methods to calculate cost volumes, these two classes of methods encode joint information from left and right images. This information is crucial for inferring disparity, which is usually accomplished using one or more encode–decoder-like structures.

Besides, unsupervised methods are also explored in stereo matching. Joung et al. [30] proposed to unsupervisedly use CNNs to compute the matching cost. They combined image domain learning with stereo epipolar constraints to obtain state-of-the-art performance.

2.2. Depth Prediction from a Single Image

Deep learning methods have helped reduce the heavy work of feature engineering [31,32]. In the context of deep learning methods, depth prediction from a single image often uses an encoder-like network to extract a feature map for a single image. Then, this feature map is upsampled and used to regress the depth map.

Eigen et al. [33] first applied a multi-scale CNN architecture to predict depth maps from monocular images, which helps capture image details. Following this, some other CNN-based methods [34] were proposed to estimate monocular depth. Xu et al. [35] combined CNN and conditional random field to improve the smoothness of estimated depth maps.

In Reference [36], the authors introduced a novel term to constrain depth and to occlude contour predictions. They used synthetic datasets for training and real datasets for finetuning. Their method has brought better accuracy along the occluding contours.

To extract high-level structure from single images, Osuna-Coutiño et al. [37] proposed to utilize region-wise analysis for better depth estimation. They segmented the depth in semantic orientations to extract high-level structures. Their work contributes to depth estimation from a single image, which has few parallax constraints.

Considering the difficulty of obtaining dense depth maps, some researches used unsupervised methods to estimate monocular depth maps. Garg et al. [38] and Clement et al. [39] proposed to use the estimated inverse depth map and the right image to reconstruct the left image. The reconstruction error was used as loss function to train the network. Luo et al. [19] first synthesized a right image from the input left image using a view-point synthesis network. Then, they used another network to perform stereo matching on this image pair to generate inverse depth map. Kuznietsov et al. [40] used deep networks to produce photo-consistent dense depth maps in a stereo setup using a direct image alignment loss. Their semi-supervised method estimates reliable depth images in realistic dynamic outdoor environments. Godard et al. [41] proposed a minimum reprojection loss to handle occlusion areas for better depth estimation. To reduce visual artifacts, they also designed a novel multi-scale sampling method. Their improvements estimated better depth maps, both quantitatively and qualitatively.

Goldman et al. [42] proposed a novel self-supervised method for depth estimation. During training, they used two twin networks to predict depth maps for both left and right images. During

testing, only one of the two networks are used. Their method provided better self-supervised estimation results, and they also performed well in unseen datasets.

2.3. Comparison between Binocular and Monocular Methods

2.3.1. Similarity

The input and output for two tasks are similar. Both of them estimate depth maps from camera images. This accounts for the similarity of functionality.

From the perspective of network structure, these two tasks are similar. Both of them use encoder–decoder-like structures. They extract feature maps from input images and upsample the feature maps to regress depth maps. This accounts for the similarity of implementation.

The similarity of functionality and implementation indicates that two tasks can be accomplished in the same architecture.

2.3.2. Differences

Although they have similar functionality and implementation, the binocular and monocular methods are different. Their differences need to be considered to accomplish them in only one architecture.

Binocular stereo matching networks take as input left and right images, while monocular networks only take single left images. This means that encoders of the two architectures are different. For binocular networks, they encode information from both left and right images whereas monocular networks only encode left image information.

The way to regress depth or disparity maps is also different. In binocular stereo matching networks, disparity maps are regressed according to the cost volume. The cost volume contains the relationship a pair of pixels, which contributes to the regression of disparity maps. However, monocular networks regress a depth map directly from the feature map, which makes the estimation ill-posed.

The two methods are different in the number of inputs and the way to regress the final results. These differences need to be tackled before the two tasks can be accomplished in the same architecture.

3. Method

Depth maps are estimated by either binocular stereo matching or from a single image. These two tasks are similar but have their differences. This section describes how the two tasks are accomplished in the same architecture using the same parameters.

3.1. Overview of the Proposed Unified Framework

Although different from each other, the two kinds of networks enjoy similarities in functionality and implementation. Figure 1 illustrates an overview of the proposed framework.

The framework consists of two functional modules, i.e., F_1 and F_2. F_1 is the module to handle different number of input camera images, and F_2 is the module to perform depth or disparity regression.

To accomplish them simultaneously, an architecture must be capable of dealing with different types of input, i.e., one single image or a pair of images. Moreover, it is necessary to design a regression sub-architecture. This sub-architecture can regress the results from either a feature map for one image or a cost volume for an image pair.

F_1 module detects whether the input is a monocular image or a rectified pair of stereo images. Then, it separates them as left and (fake) right images. These two images are taken as input in F_2 module.

For F_2 and F_3 modules, two approaches are available to accomplish both tasks in one architecture, which correspond to two kinds of networks. On the one hand, binocular networks take as input a pair of images. This study regards monocular methods as binocular ones without right images.

To accomplish the monocular task, the absent input needs to be filled. On the other hand, monocular networks are fed with only single images. To accomplish binocular task, an extra branch is needed for right images. Considering the former is easier, this study decides to use binocular stereo matching networks as the basic framework, i.e., DispNetC and PSMNet. Figures 2 and 3 illustrate detailed architectures used in this study. The details of DispNetC and PSMNet can be found in their papers and codes.

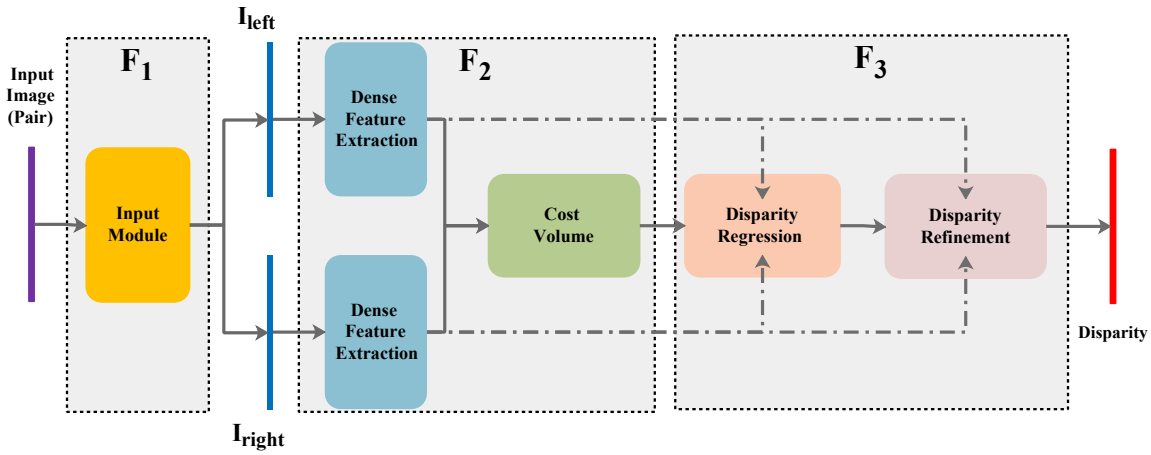

Figure 1. Frameworks of the proposed method.

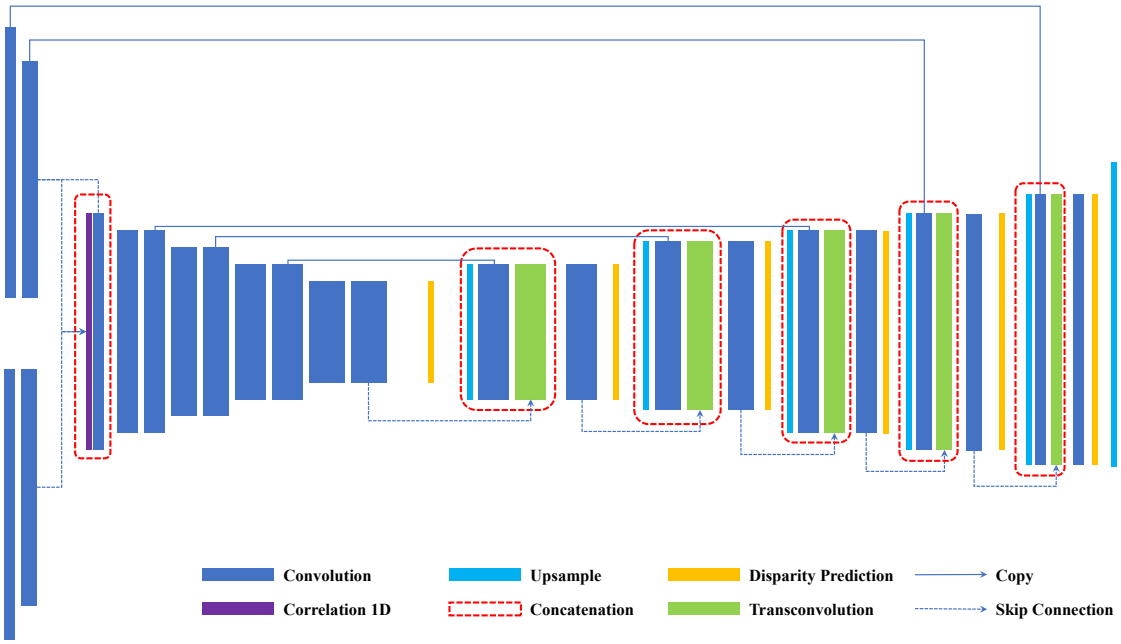

Figure 2. Architecture of DispNetC.

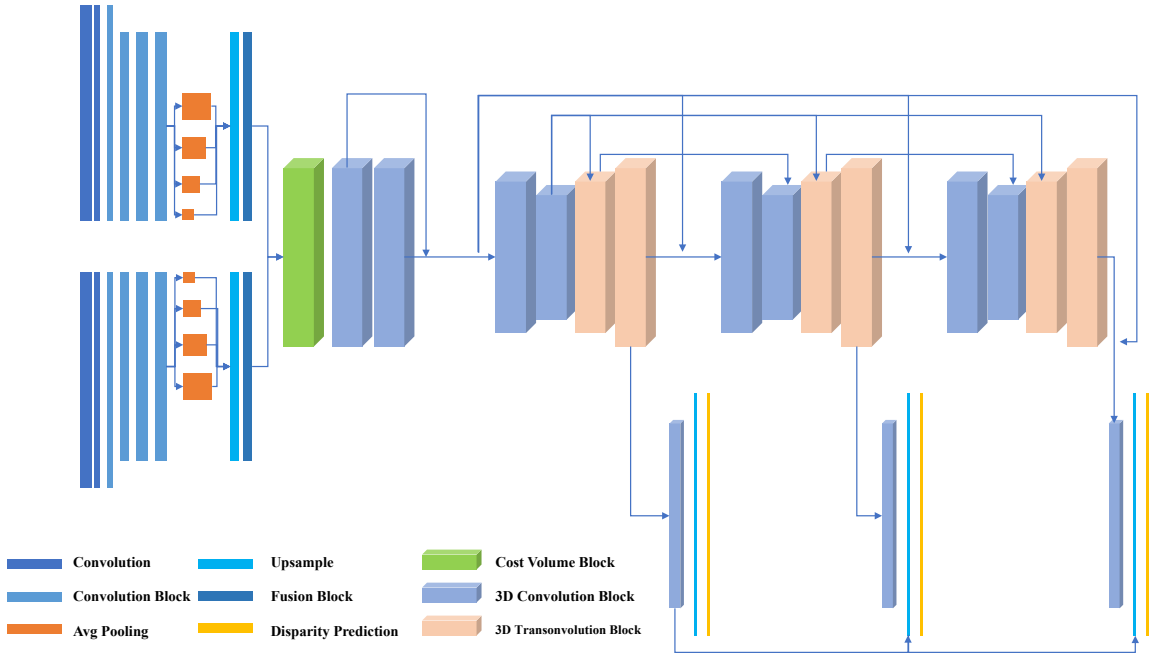

Figure 3. Architecture of PSMNet.

3.1.1. Module to Handle Both Types of Inputs

As illustrated in Figure 1, the F_1 module assures that the network can accept both types of inputs. It is fed with either a pair of images or a single image. Assuming the shape of an input image is $(H, W, 3)$, the output of F_1 module is a tensor T shaped like $(H, W, 6)$, where $T[0:H, 0:W, 0:3]$ represents th eleft image and $T[0:H, 0:W, 3:6]$ represents the right image. For a pair of images, the F_1 module simply concatenates left and right images to form the output. For single images, the F_1 module takes one of the two policies: (1) duplicating the left image to use as the right one and (2) filling the right image with a certain value, e.g., 0. With these two policies, the F_1 module outputs a $(H, W, 6)$ tensor regardless of the input.

3.1.2. Module to Form a Cost Volume

The proposed framework regards the $H * W * 6$ input as the combination of two images. For stereo matching, they are left and right images. For depth prediction from a single image, they are left image and corresponding learning cues. F_2 module focuses on how to form a cost volume, i.e., a fair feature map for both tasks. Available schemes include (1) stacking left and right feature maps, e.g., PSMNet, and (2) calculating the correlation between two feature maps, e.g., DispNetC.

PSMNet forms a 4D cost volume by concatenating left feature maps with their corresponding right feature maps across each disparity level. Then, it uses 3D convolution layers to learn matching cost estimation. For stereo matching tasks, this approach performs well. However, for depth prediction from a single image, the sources to form a cost volume are the left image and its duplicate or a constant-value tensor. This makes it hard to learn a cost volume by only rearranging the order of pixels.

An alternative approach is to explicitly calculate the correlation between two feature maps, as in DispNetC. The explicit calculation is effective for both types of inputs. When it is fed with a pair of stereo images, the F_2 module calculates the correlation between them and outputs their matching cost as the cost volume. For a single left image and its duplicate, the F_2 module calculates their self-similarity to form a cost volume.

3.1.3. Module to Regress a Depth Map

The F_1 module enables a stereo matching network to accept both types of inputs. The F_2 module forms a cost volume as the final feature to regress the result. To regress depth maps, a module, F_3, needs to upsample the cost volume and to handle the lost details. Actually, the decoder parts are almost the same for binocular and monocular networks. Therefore, this study does not adjust the network structure for regressing depth maps. Instead, the difference of regressing depth maps from a cost volume and from a single feature map is handled by the training procedure.

3.2. Procedure to Train the Network

This study aims to accomplish both tasks using the same architecture and the same parameters. Stereo matching networks are used as backbone architecture. The different inputs are handled by introducing the F_1 module. The training procedure makes the F_3 module capable of learning fair parameters to extract features and to regress depth maps towards both tasks.

With the rapid development of big data [43,44], considering that learning-based methods are highly dependent on data themselves, this study dynamically changes data for training a unified framework for both tasks. Specifically, the architecture alternatively selects training examples from two tasks during each iteration. This ensures that the model is optimized towards both tasks. From the aspect of functionality, F_3 can be regarded as a many-to-one mapping function, which uses the cost volume as the input and produces depth as its output. In F_3, our method adds additional regularization for optimizing the model compared with the original model, which is optimized towards only one task.

3.3. Loss Function

The framework was trained with both supervised and unsupervised loss functions, as in Equation (2), where λ is the hyper-parameter to adjust the weight of supervised loss and unsupervised loss.

$$Loss = \lambda Loss_{supervised} + (1 - \lambda) Loss_{unsupervised} \tag{2}$$

The supervised loss is defined in Equation (3), where N is the number of valid pixels in the ground-truth depth map, d is the ground-truth depth map, and $\hat{d}$ is the predicted one.

$$Loss_{supervised} = \frac{1}{N} \sum_{i=1}^{N} |d_i - \hat{d}_i| \tag{3}$$

The real ground-truth data are often sparse. This promotes the need for unsupervised methods. To improve the results, this study adopts unsupervised loss to train the framework. The unsupervised loss is defined in Equation (4), where I^{left} and I^{right} represent the image pair and where $Warp(I, d)$ means the warped image from image I and disparity d.

$$Loss_{unsupervised} = \frac{1}{N} \sum_{i=1}^{N} |I_i^{left} - Warp(I^{right}, \hat{d})_i| \tag{4}$$

4. Results

Experimental studies were conducted (1) to select the best input scheme for the proposed framework, (2) to compare different training procedures of the proposed framework and to evaluate the depth estimation performance of the proposed framework, and (3) to validate that the proposed framework can accomplish both depth prediction from a single image and binocular stereo matching in the same architecture using the same parameters. First, we give the implementation details of the proposed framework. Then, the evaluation metrics are introduced. Finally, the quantitative and qualitative results are given to illustrate the effectiveness of the proposed framework.

4.1. Implementation Details

We implemented the proposed framework based on the platform *PyTorch*. The experiments were conducted on a single NVIDIA Titan XP GPU with 12 GB GPU memory. The model was first pretrained on the large synthetic Scene Flow dataset, which contains about 30,000 training image pairs and 4370 test image pairs. Then, we fine-tuned the network on the training split of the KITTI dataset (used by Eigen et al.) to obtain the finally model. The model was optimized using the Adam method with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. It was trained using the loss functions defined in Section 3.3, with $\lambda = 0.5$.

We chose *DispNetC* and *PSMNet* as backbone architectures. We used our proposed training procedure to train *DispNetC* on Scene Flow dataset for 100 epochs and fine-tuned it on KITTI raw dataset for 80 epochs. The learning rate was set as 10^{-4} for the first 50 epochs and as 10^{-5} for the other epochs. *PSMNet* was trained on Scene Flow dataset for 20 epochs and fine-tuned on KITTI raw dataset for 40 epochs. The learning rate was set as 10^{-3} for the first 10 epochs and as 10^{-4} for the other epochs.

4.2. Evaluation Metrics

To align with the method in Reference [33], we evaluated our framework with the following error metrics:

$$\begin{array}{ccc} \text{(Absolute Relative Difference)} & ARD & = \frac{1}{|N|}\sum_i ||d_i - \hat{d}_i||/d_i \\ \text{(Squared Relative Difference)} & SRD & = \frac{1}{|N|}\sum_i ||d_i - \hat{d}_i||^2/d_i \\ \text{(Linear RMSE)} & RMSE & = \sqrt{\frac{1}{|N|}\sum_{y\in T} ||d_i - \hat{d}_i||^2} \\ \text{(Log RMSE)} & RLog & = \sqrt{\frac{1}{|N|}\sum_i ||\log d_i - \log \hat{d}_i||^2} \end{array}$$

where d and $\hat{d}$ denote the ground-truth depth map and the predicted depth map, respectively; N denotes the total number of pixels; and i denotes the pixel positions.

We also measured the accuracy, for which higher is better. Assuming t_i as the threshold value for pixel i, i.e., $t_i = max(\frac{d_i}{\hat{d}_i}, \frac{\hat{d}_i}{d_i})$, three metrics ($\delta_1$, δ_2 and δ_3) were defined as in Equation (5).

$$\delta_k = \frac{1}{N} * (Number\ of\ pixels\ with\ t < 1.25^k), k \in [1,2,3] \tag{5}$$

4.3. Comparison between Different Input Schemes

As mentioned in Section 3.1.1, there are two schemes for the proposed framework to handle the missing right input image, i.e., (1) duplicating left one and (2) filling the right one with a certain value, e.g., 0.

The performance of depth prediction from a single image with different input schemes is shown in Table 1. Note that the architecture with suffix *Cross* means duplicating the left image as the right one, whereas *Zero* means filling the right image input with zero. It can be observed that, for both $DispNetC$ and $PSMNet$, duplicating the left image as the right input achieves much a better performance than setting it to zero in terms of all the metrics. Besides, $DispNetC$ is better than $PSMNet$. This is because F_2 in $DispNetC$ takes as input the concatenation of cost volume (calculated using correlation) and left features and the network can learn depth information from those left features. However, in $PSMNet$, $F2$ takes as input only the cost volume (calculated using concatenation of left and right features) and applies 3D convolution upon the cost volume, which is intuitively and experimentally not suitable for learning depth information.

Table 1. Comparison between different input schemes

Architecture		ARD	SRD	RMSE	RLog	δ_1	δ_2	δ_3
		(Lower Is Better)				(Higher Is Better)		
DispNetC	(Zero)	0.744	11.566	12.312	0.639	0.267	0.502	0.689
DispNetC	(Cross)	**0.114**	**0.821**	**4.687**	**0.192**	**0.854**	**0.947**	**0.979**
PSMNet	(Zero)	0.716	10.513	17.15	1.528	0.009	0.026	0.059
PSMNet	(Cross)	**0.176**	**1.228**	**5.862**	**0.252**	**0.741**	**0.916**	**0.968**

The bolded statistics are the better results.

4.4. Performance for Depth Prediction from a Single Image

Table 2 compares our method with some typical methods for depth prediction from a single image. It is noted that the LRC method uses a similar U-net architecture as *DispNetC*, but LRC uses more convolution layers and outputs disparity maps of half resolution. Therefore, LRC outperforms *DispNetC(Mono)*, which is trained only with left images. However, when using our proposed training procedure, the performance of *DispNetC(Cross)* is better than that of LRC. This result indicates that our training procedure encodes knowledge from binocular stereo matching, which is helpful for single-image depth estimation. On the contrary, *PSMNet(Mono)* performs better than *PSMNet(Cross)*. This means that the stereo knowledge learned by *PSMNet* could not be well exploited for single-image depth estimation.

Table 2. Quantitative results for depth prediction from a single image on the test set of KITTI raw dataset.

Architecture		ARD	SRD	RMSE	RLog	δ_1	δ_2	δ_3
		(Lower Is Better)				(Higher Is Better)		
Make3D [7]	(Mono)	0.28	3.012	8.734	0.361	0.601	0.82	0.926
Eigen et al. [33]	(Mono)	0.19	1.515	7.156	0.27	0.692	0.899	0.967
Liu et al. [8]	(Mono)	0.217	1.841	6.986	0.289	0.647	0.882	0.961
LRC [39]	(Mono)	0.114	0.898	4.935	0.206	0.861	0.949	0.976
DORN [45]	(Mono)	**0.072**	**0.307**	**2.727**	**0.120**	**0.932**	**0.984**	**0.994**
DispNetC	(Mono)	0.191	1.369	5.743	0.259	0.73	0.903	0.964
DispNetC	(Cross)	**0.114**	**0.821**	**4.687**	**0.192**	**0.854**	**0.947**	**0.979**
PSMNet	(Mono)	**0.146**	**1.071**	**5.809**	**0.237**	**0.779**	**0.919**	**0.968**
PSMNet	(Cross)	0.176	1.228	5.862	0.252	0.741	0.916	0.968

The bolded statistics are the better results.

The proposed method did not perform better than DORN [45], this can be explained by more complicated structure employed in DORN. However, the main purpose of this study is to offer a unified framework for both single-image depth estimation and stereo matching. So it is possible to design a more flexible and well-designed stereo matching network to obtain better performance.

4.5. Accomplishing Both Tasks in the Same Architecture

As shown in Table 3, *(Stereo)* results were obtained using original trained model provided by the authors, whereas *(Cross)* results were obtained using models trained by our method. The performances for our method and the original models are similar. This demonstrates that our method enables stereo matching networks capable of depth prediction from a single image while keeping their performance for stereo matching.

Table 3. Quantitative results of stereo matching.

Architecture		ARD	SRD	RMSE	RLog	δ_1	δ_2	δ_3
		(Lower Is Better)				(Higher Is Better)		
DispNetC	(Cross)	0.081	**0.468**	**3.417**	**0.143**	**0.938**	**0.976**	**0.989**
DispNetC	(Stereo)	**0.070**	0.489	3.666	0.168	0.937	0.969	0.981
PSMNet	(Cross)	0.079	0.557	3.892	0.163	0.920	0.965	**0.983**
PSMNet	(Stereo)	**0.060**	**0.441**	**3.376**	**0.165**	**0.944**	**0.968**	0.981

The bolded statistics are the better results.

Figure 4 shows the qualitative results of the proposed method from the KITTI raw dataset. The disparity maps from ground truth are interpolated for better visualization. It is observed that, with the proposed method, the model can estimate depth from both a single image and a pair of images. Using a pair of images generates better results, since it utilizes more information than using a single image.

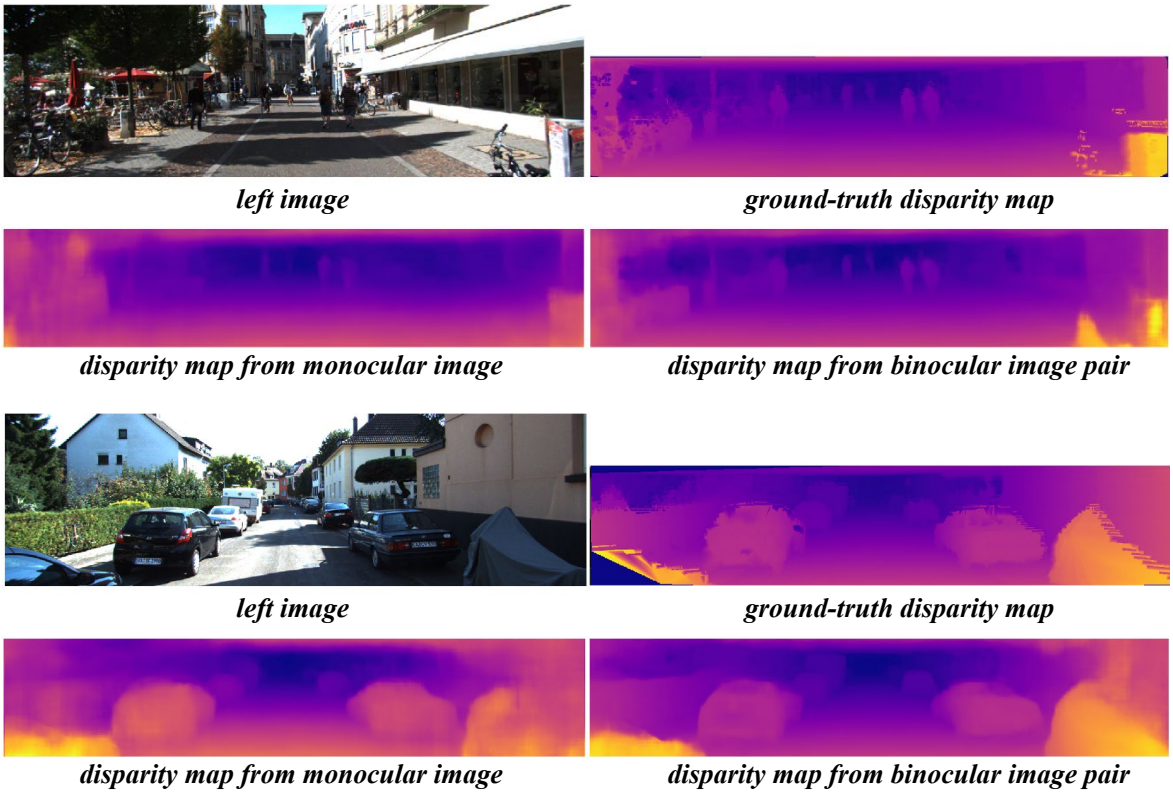

Figure 4. Qualitative results on KITTI dataset.

5. Conclusions

This study developed a unified framework to estimate depth from either single images or binocular image pairs, namely DoubleNet. The DoubleNet framework can accomplish both depth prediction from a single image and binocular stereo matching using the same architecture with the same parameters.

The DoubleNet framework employed typical stereo matching architectures as backbone. These architectures were modified to accept different types of inputs, with the proposed module for input handling. The modified architectures were trained using a novel training procedure, i.e., alternatively selecting monocular and binocular inputs during training iterations. This ensured that the architecture was optimized towards both tasks.

Experimental results indicated that the DoubleNet could reach state-of-the-art performance for the unified task of both single-image depth estimation and binocular stereo matching. In other words, the trained model could perform well in depth prediction from a single image and it could still reach similar accuracy for binocular stereo matching without extra training or adaptation.

Furthermore, DoubleNet is designed to explore the similarity between single-image depth estimation and binocular stereo matching. Experimental results also demonstrate that single-image depth estimation could benefit from stereo matching. This indicates that one of these tasks can be promoted by the other using the proposed training procedure.

Overall, the work paved the way to probing the unified framework for different but similar tasks by designing an architecture to accept different training samples for these tasks. The unified framework was trained by mixing different training samples. The work also explores the potential to benefit a specific task from another similar task.

Author Contributions: Conceptualization, Z.L. and W.C.; methodology, Z.L. and X.L.; software, X.L.; validation, W.C., C.L. and X.J.; formal analysis, W.C.; investigation, Y.G. and M.W.; resources, M.W.; data curation, C.L.; writing–original draft preparation, X.L. and W.C.; writing–review and editing, X.L. and Z.L.; visualization, X.L.; supervision, W.C. and Z.L.; project administration, W.C.; funding acquisition, W.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported by the Key Technologies Research and Development Program of China under grant No. 2018YFB0803501. It is also supported by National Key Research and Development Program of China under grant No. 2018YFB0204301.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, C.; Seff, A.; Kornhauser, A.; Xiao, J. DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 2722–2730. [CrossRef]
2. Kalia, M.; Navab, N.; Salcudean, T. A Real-Time Interactive Augmented Reality Depth Estimation Technique for Surgical Robotics. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019. [CrossRef]
3. Kim, H.; Sohn, K. Hierarchical Depth Estimation for Image Synthesis in Mixed Reality. *Proc. SPIE Int. Soc. Opt. Eng.* **2003**, *5006*. [CrossRef]
4. Zhang, X.; Khan, M. *Functions of Laser Radar in Intelligent Cars*; Springer: Singapore, 2019; pp. 263–274. [CrossRef]
5. Olaya, E.; Berry, F.; Mezouar, Y. *A robotic structured light camera*. In Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Besacon, France, 8–11 July 2014; pp. 727–734. [CrossRef]
6. Gansbeke, W.; Neven, D.; Brabandere, B. Sparse and Noisy LiDAR Completion with RGB Guidance and Uncertainty. In Proceedings of the 2019 16th International Conference on Machine Vision Applications (MVA), Tokyo, Japan, 27–31 May 2019. [CrossRef]
7. Saxena, A.; Sun, M.; Ng, A. Learning 3-D Scene Structure from a Single Still Image. In Proceedings of the 2007 IEEE 11th International Conference on Computer Vision, Rio de Janeiro, Brazil, 14–21 October 2007; pp. 1–8. [CrossRef]
8. Liu, F.; Shen, C.; Lin, G.; Reid, I. Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*. [CrossRef] [PubMed]
9. Rajagopalan, A.; Chaudhuri, S.; Uma, M. Depth Estimation and Image Restoration Using Defocused Stereo Pairs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 1521–1525. [CrossRef] [PubMed]
10. Bhavsar, A.; Rajagopalan, A. *Depth Estimation with a Practical Camera*; British Machine Vision Conference (BMVC): London, UK, 2009; pp. 104.1–104.11 [CrossRef]
11. Godard, C.; Aodha, O.M.; Brostow, G.J. Unsupervised Monocular Depth Estimation with Left-Right Consistency. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6602–6611. [CrossRef]

12. Mancini, M.; Costante, G.; Valigi, P.; Ciarfuglia, T.A.; Delmerico, J.; Scaramuzza, D. Toward Domain Independence for Learning-Based Monocular Depth Estimation. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1778–1785. [CrossRef]
13. Ye, X.; Zhang, M.; Xu, R.; Zhong, W.; Fan, X.; Liu, Z.; Zhang, J. Unsupervised Monocular Depth Estimation Based on Dual Attention Mechanism and Depth-Aware Loss. In Proceedings of the 2019 IEEE International Conference on Multimedia and Expo (ICME), Shanghai, China, 8–12 July 2019; pp. 169–174. [CrossRef]
14. Kaushik, V.; Lall, B. UnDispNet: Unsupervised Learning for Multi-Stage Monocular Depth Prediction. In Proceedings of the 2019 International Conference on 3D Vision (3DV), Québec City, QC, Canada, 16–19 September 2019; pp. 633–642. [CrossRef]
15. Mayer, N.; Ilg, E.; Hausser, P.; Fischer, P.; Cremers, D.; Dosovitskiy, A.; Brox, T. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4040–4048. [CrossRef]
16. Kendall, A.; Martirosyan, H.; Dasgupta, S.; Henry, P. End-to-End Learning of Geometry and Context for Deep Stereo Regression. In Proceedings of the2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 66–75. [CrossRef]
17. Liang, Z.; Feng, Y.; Guo, Y.; Liu, H.; Chen, W.; Qiao, L.; Zhou, L.; Zhang, J. Learning for Disparity Estimation Through Feature Constancy. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2811–2820. [CrossRef]
18. Chang, J.; Chen, Y. Pyramid Stereo Matching Network. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5410–5418. [CrossRef]
19. Luo, Y.; Ren, J.; Lin, M.; Pang, J.; Sun, W.; Li, H.; Lin, L. Single View Stereo Matching. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 155–163. [CrossRef]
20. Ke, H.; Chen, D.; Shi, B.; Zhang, J.; Liu, X.; Zhang, X.; Li, X. Improving Brain E-health Services via High-Performance EEG Classification with Grouping Bayesian Optimization. *IEEE Trans. Serv. Comput.* **2019**, 1–14. [CrossRef]
21. Tang, Y.; Chen, D.; Wang, L.; Zomaya, A.Y.; Chen, J.; Liu, H. Bayesian tensor factorization for multi-way analysis of multi-dimensional EEG. *Neurocomputing* **2018**, *318*, 162–174. [CrossRef]
22. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets Robotics: The KITTI Dataset. *IJRR* **2013**, *32*. [CrossRef]
23. Zbontar, J.; LeCun, Y. Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. *J. Mach. Learn. Res.* **2016**, *17*, 2.
24. Luo, W.; Schwing, A.G.; Urtasun, R. Efficient Deep Learning for Stereo Matching. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 5695–5703. [CrossRef]
25. Cheng, X.; Wang, P.; Yang, R. Learning Depth with Convolutional Spatial Propagation Network. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, 1. [CrossRef] [PubMed]
26. Pang, J.; Sun, W.; Ren, J.S.; Yang, C.; Yan, Q. Cascade Residual Learning: A Two-Stage Convolutional Neural Network for Stereo Matching. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 878–886. [CrossRef]
27. Ilg, E.; Saikia, T.; Keuper, M.; Brox, T. Occlusions, Motion and Depth Boundaries with a Generic Network for Disparity, Optical Flow or Scene Flow Estimation. In Proceedings of the 15th European Conference, Munich, Germany, 8–14 September 2018; pp. 626–643. [CrossRef]
28. Song, X.; Zhao, X.; Hu, H.; Fang, L. EdgeStereo: A Context Integrated Residual Pyramid Network for Stereo Matching. In Proceedings of the Computer Vision (ACCV 2018), Perth, Australia, 2–6 December 2018; pp. 20–35. [CrossRef]
29. Yang, G.; Zhao, H.; Shi, J.; Deng, Z.; Jia, J. SegStereo: Exploiting Semantic Information for Disparity Estimation. In Proceedings of the 15th European Conference, Munich, Germany, 8–14 September 2018; pp. 660–676. [CrossRef]
30. Joung, S.; Kim, S.; Park, K.; Sohn, K. Unsupervised Stereo Matching Using Confidential Correspondence Consistency. *IEEE Trans. Intell. Transp. Syst.* **2019**, 1–14. [CrossRef]

31. Chen, D.; Tang, Y.; Zhang, H.; Wang, L.; Li, X. Incremental Factorization of Big Time Series Data with Blind Factor Approximation. *IEEE Trans. Knowl. Data Eng.* **2019**, 1. [CrossRef]
32. Hengjin, k.; Chen, D.; Shah, T.; Liu, X.; Zhang, X.; Zhang, L.; Li, X. Cloud-aided online EEG classification system for brain healthcare: A case study of depression evaluation with a lightweight CNN. *Softw. Pract. Exp.* **2018**. [CrossRef]
33. Eigen, D.; Puhrsch, C.; Fergus, R. Depth Map Prediction from a Single Image Using a Multi-scale Deep Network. In Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS'14), Montreal, QC, Canada, 8–13 December 2014; MIT Press: Cambridge, MA, USA, 2014; Volume 2, pp. 2366–2374.
34. Liu, J.; Zhang, Y.; Cui, J.; Feng, Y.; Pang, L. Fully Convolutional Multi-scale Dense Networks for Monocular Depth Estimation. *IET Computer Vision* **2019**, *13*. [CrossRef]
35. Xu, D.; Ricci, E.; Ouyang, W.; Wang, X.; Sebe, N. Monocular Depth Estimation Using Multi-Scale Continuous CRFs as Sequential Deep Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 1426–1440. [CrossRef] [PubMed]
36. Ramamonjisoa, M.; Lepetit, V. Sharpnet: Fast and accurate recovery of occluding contours in monocular depth estimation. *arXiv* **2019**, arXiv:1905.08598
37. Osuna-Coutiño, J.; Martinez-Carranza, J. High Level 3D Structure Extraction from a Single Image Using a CNN-Based Approach. *Sensors* **2019**, *19*, 563. [CrossRef] [PubMed]
38. Garg, R.; Kumar, B.V.; Carneiro, G.; Reid, I. Unsupervised CNN for single view depth estimation: Geometry to the rescue. In Proceedings of the 14 th European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Amsterdam, The Netherlands, 2016; pp. 740–756.
39. Godard, C.; Aodha, O.; Brostow, G. Unsupervised Monocular Depth Estimation with Left-Right Consistency. *Softw. Pract. Exp.* **2017**. [CrossRef]
40. Kuznietsov, Y.; Stuckler, J.; Leibe, B. Semi-supervised deep learning for monocular depth map prediction. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6647–6655.
41. Godard, C.; Mac Aodha, O.; Firman, M.; Brostow, G.J. Digging into self-supervised monocular depth estimation. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–3 November 2019; pp. 3828–3838.
42. Goldman, M.; Hassner, T.; Avidan, S. Learn stereo, infer mono: Siamese networks for self-supervised, monocular, depth estimation. *arXiv* **2019**, arXiv:1905.00401.
43. Chen, D.; Li, X.; Wang, L.; Khan, S.; Juan, W.; Zeng, K.; Cai, C. Fast and Scalable Multi-Way Analysis of Massive Neural Data. *IEEE Trans. Comput.* **2015**, *64*, 707–719. [CrossRef]
44. Chen, D.; Hu, Y.; Wang, L.; Zomaya, A.Y.; Li, X. H-PARAFAC: Hierarchical Parallel Factor Analysis of Multidimensional Big Data. *IEEE Trans. Parallel Distrib. Syst.* **2017**, *28*, 1091–1104. [CrossRef]
45. Fu, H.; Gong, M.; Wang, C.; Batmanghelich, K.; Tao, D. Deep ordinal regression network for monocular depth estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2002–2011.