World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:16, No:2, 2022

Depth Estimation in DNN Using Stereo Thermal Image Pairs

Ahmet Faruk Akyuz, Hasan Sakir Bilge

***Abstract*—**Depth estimation using stereo images is a challenging problem in computer vision. Many different studies have been carried out to solve this problem. With advancing machine learning, tackling this problem is often done with neural network-based solutions. The images used in these studies are mostly in the visible spectrum. However, the need to use the Infrared (IR) spectrum for depth estimation has emerged because it gives better results than visible spectra in some conditions. At this point, we recommend using thermal-thermal (IR) image pairs for depth estimation. In this study, we used two well-known networks (PSMNet, FADNet) with minor modifications to demonstrate the viability of this idea.

***Keywords*—**Thermal stereo matching, depth estimation, deep neural networks, CNN

I. Introduction

DEPTH estimation is an important problem in computer vision applications. One way to obtain depth maps is to use stereo images. Stereo images are taken from two identical cameras that are calibrated the same way and horizontally aligned. Thus, it is determined how many pixels are shifted by inspecting a pixel taken from the first camera with respect to its corresponding position in the second camera. If shifting is larger, then the corresponding pixel is closer and vice versa. This process can be done using Deep Neural Networks (DNN) as well as traditional methods such as SGM [1].

Semi-global Matching (SGM) algorithms are done by traditional methods. Besides it is fast and applicable in real scenarios, its accuracy is not enough in many scenes. Textureless regions and repetitive pixels lie behind this accuracy problem. Therefore, a learnable structure was needed. This need was met with DNN.

DNN usage is a popular approach for depth map estimation. In recent studies, it has been observed that convolutional neural networks (CNNs) are very successful in applications such as feature extraction and similarity computation besides being more useful in terms of speed and consistency than traditional methods [2], [3]. There are many studies to estimate depth map using stereo matching networks such as PSMNet [4], FADNet [5], etc. These networks are based on supervised learning approaches where input and output relation is very important. The dataset should be given to the network correctly.

There are three dataset types used in stereo matching algorithms; synthetic, laboratory, and real, which are shown in Fig. 1. Synthetic datasets, such as scene flow dataset [6], are created using computers. However, such datasets cannot reflect the real-world data distribution, so they are not suitable for real scenarios. Lab datasets such as Middlebury [7] are created with real-world images but do not use a range sensor. Although they are closer to reality than synthetic ones, they are also not sufficient for real scenarios. Real datasets such as driving stereo dataset [8], KITTI [9] are originated from real cameras and scenarios with a range sensor. Therefore, such a dataset is suitable for real applications.

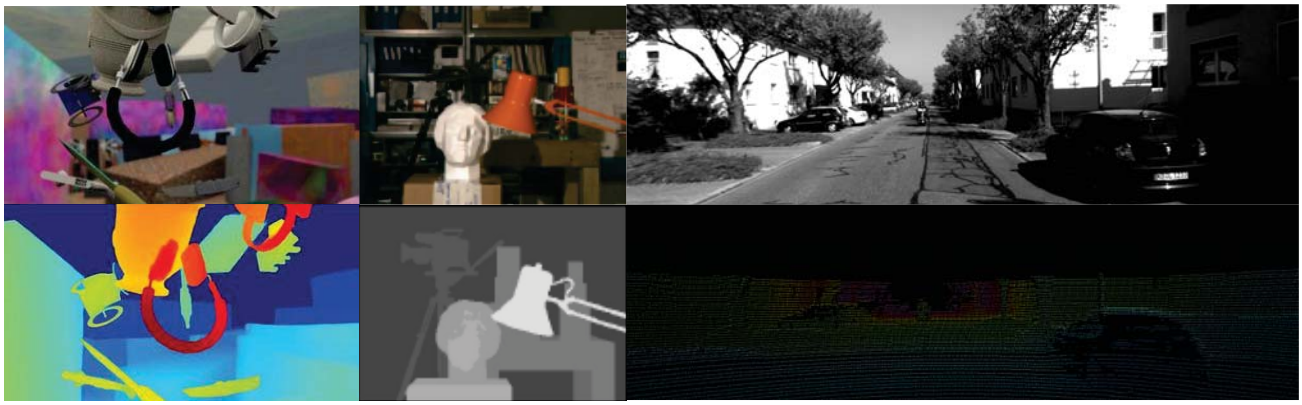
Real datasets should have some important features such as calibration, rectification, and registration to get good results and accuracy [9]. Calibrating the cameras among themselves is the first step. In this step, camera parameters are optimized to minimize the average reprojection error [10]. Secondly, since the origin of the range sensor and reference camera is different in the 3D coordinate system, the center location of the range sensor and reference camera is required to be in the same position to obtain the same view. Finally, point clouds taken from the range sensor should be registered according to the camera exposure. When these steps are ignored or not considered in supervised learning, it causes an erroneous samples and this leads for the neural network not learning or even learning incorrectly.

With the developing technology, the idea of using thermal-thermal image pairs for depth estimation comes from the invisibility problems in some conditions. Thermal cameras are known to offer a better view than visible cameras when the scene is foggy, misty, rainy, etc. [11]. For example, in autonomous driving or even in human-based driving, visibility range and texture quality decrease considerably, especially for night conditions. For this reason, thermal cameras are frequently used both for warning the driver and for automatic driving. Therefore, estimating depth map using thermal image pairs has an advantage compared to the visible spectrum.

There are some studies related to the usage of thermal-thermal stereo pairs for depth estimation. Arnab Dhua et al. proposed a method to calculate depth map with given uncalibrated thermal stereo pairs. They first compute a sparse disparity map using corner matching methods, then the computed map is improved with triangular constraints and epipolar geometrical constraints as the proposed method [12]. Massimo Bertozzi et al. proposed an algorithm to detect pedestrians using two identical thermal cameras. The algorithm first locates and estimates warm areas in the scene, then considers specific sizes and aspect ratios in areas with a similar positions, and finally, it uses the morphological and thermal features of a human head to list possible pedestrians [13]. Geoffroy et al. proposed a method of sub-pixel matching in low-resolution

A. F. Akyuz is with the Department of Electrical and Electronics Engineering, Gazi University and ASELSAN INC., Ankara, Turkey (e-mail: ahmetfaruk.akyuz@gazi.edu.tr, afakyuz@aselsan.com.tr).

H. S. Bilge is with the Department of Electrical and Electronics Engineering, Gazi University, Ankara, Turkey (e-mail: bilge@gazi.edu.tr).



(d) Scene Flow Dataset (e) Middlebury Dataset (f) KITTI dataset

Fig. 1: Dataset types. (d) Synthetic Dataset (e) Laboratory Dataset (f) Real Dataset

stereo thermal images. Firstly, they extract robust features based on phase congruency, then they match these features in pixel precision, and finally, they refine matching in sub-pixel accuracy based on local phase coherence [14]. None of these studies use neural network solutions. In this work, we show that depth estimation can be done by using thermal stereo pairs in existing neural network solutions such as FADNet, PSMNet.

The rest of the paper is organized as follows. We introduce network structures that we have examined in Section II. Section III shows preliminary work in both network structures and datasets. We demonstrate our experimental results in Section IV. Finally, we conclude the paper in Section V.

II. Network Structures

PSMNet benefits from different scales of receptive fields to extend pixel-level features to region-level features. This leads global and local feature clues forming the cost volume to get reliable disparity estimation. PSMNet also using a stacked hourglass that was applied repeatedly 3D CNN layers to regularize the cost volume.

On the other hand, FADNet effectuated 2D-based correlation layers with stacked blocks instead of 3D CNN's. This helps to preserve fast computation. In that way, it gets closer to be used in real applications. FADNet uses multi-scale predictions so that multi-scale weight scheduling training techniques can be applied to improve accuracy. The network structure of FADNet consists of two hands as DispNetC and DispNetS.

III. Preliminary Work

Before we train networks with a thermal-thermal stereo dataset, some preliminary work is required on both the structure of the network and the dataset. This preprocessing can be a subject for another study in order to be a difficult subject, but we have done this very simple way to show even though these preprocesses are not perfect, they are still quite good and bright for further studies.

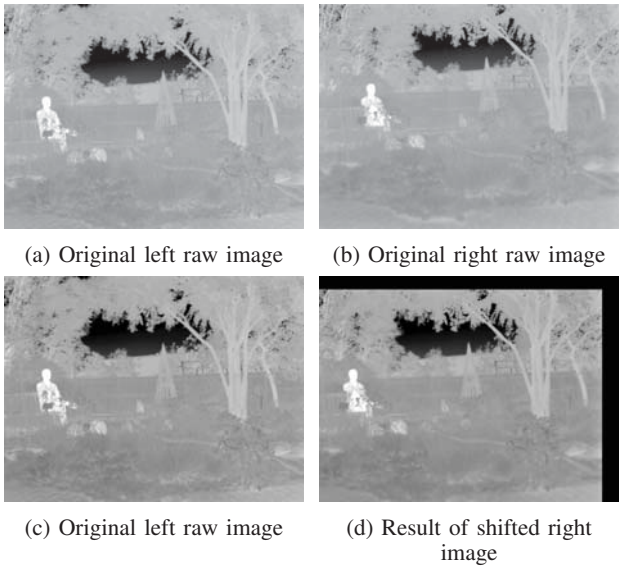
The first thing we did as preliminary work is to fit the network structure according to proper dataset feeding. Namely, current studies use stereo images as 3-channel because of taken in the visible spectrum (RGB images). In the infrared spectrum, images are gray-scale. Therefore, we should have changed the layers of both networks where RGB images were used to one channel. In this context, we made the following changes:

In FADNet, there are three modifications. First, in DispNetC hand, the first convolutional layer was changed from three filter sizes to one filter size. Secondly, on the same hand, the last convolutional layer was changed from 20 filter sizes to 18 filter sizes. The third modification is in DispNetS hand. We decreased input filter size from 11 to 5. In PSMNet, on the other hand, there is only one modification. In the feature extraction part, we decreased the first layer filter size from three to one. With these changes, the networks became ready to be fed single-channel images.

Due to the lack of datasets containing stereo thermal images with depth information, we were able to find only one useful dataset called CATS dataset [15]. This dataset provides stereo visible image, stereo thermal image, and depth information taken from range sensor (LiDAR) with 343 total samples. It has 100 indoor and 80 outdoor samples from various scenes in different environmental conditions including daytime, nighttime and foggy scenes. Since the scene of outdoor images is more obvious compared to indoor images in the thermal spectrum, we only dealt with outdoor images in this work. We divided outdoor images into 70 for training and 10 for validating.

Although CATS is a real dataset, the features of this dataset mentioned in the introduction section are not suitable for stereo algorithms, especially in neural network applications. Therefore, we need to perform rectification of images and registration between ground truth and the reference image for this dataset.

CATS has rectified image pairs themselves. However, these pairs are not proper for stereo algorithms because of rectification logic. Therefore, we decided to use original raw images to do the rectification process as shown in Fig. 2. We applied John Mallon and Paul F. Whelan's rectification method called *"Projective Rectification from the fundamental matrix"* [16].



(a) Original left raw image (b) Original right raw image

(c) Original left raw image (d) Result of shifted right image

Fig. 2: Image rectification process

We performed the following steps:

- detect the image which has a particular pixel at the deepest point since there should not be any shift of the deepest point
- match this pixel in horizontal and vertical axes to be in the same pixel location by shifting the right original image concerning original left image
- apply the same shifting amount for each image in both directions

We chose to use left images as the references, therefore we shifted only the right images in all samples. There should be a relation between input and output in the case of supervised learning as we mentioned before. With this purpose, we should project 3D point clouds ($\mathbf{p_{lid}}$) acquired from the range sensor with respect to the left camera that we chose as reference. To do so, we transformed (translation and rotation) the 3D point clouds according to the left camera's coordinate ($\mathbf{p_{cam}}$) since the center positions of the range sensor and left thermal camera are different. Transformation formula of 3D coordinates to camera coordinates as given as:

$$\mathbf{p_{cam}} = \mathbf{R}(\mathbf{p_{lid}} - \mathbf{t}) \tag{1}$$

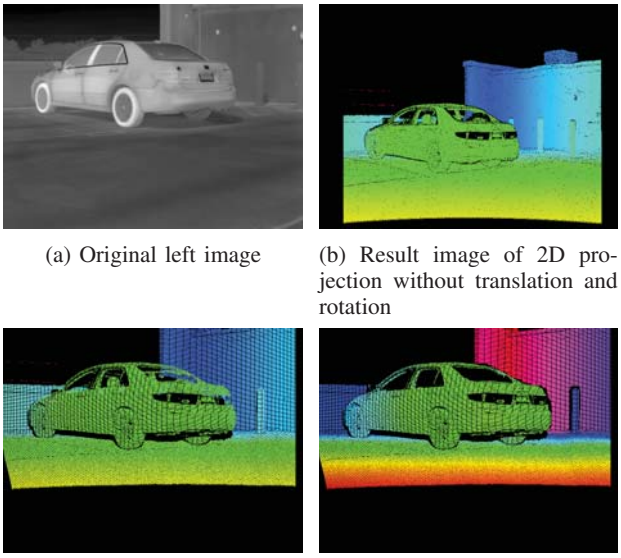
where $\mathbf{R}$ is the 3D rotation matrix and $\mathbf{t}$ is the shift vector.

After this transformation process, 3D coordinates is projected to the 2D image plane. We have used pinhole camera model with perspective projection [17]. Corresponding pixel positions of the 3D point $\mathbf{p_{cam}}$, u_x and u_y, are given as:

$$u_x = p^x_{cam} \frac{W}{2p^z_{cam} tan(\frac{\theta_x}{2})} \tag{2}$$

$$u_y = p^y_{cam} \frac{H}{2p^z_{cam} tan(\frac{\theta_y}{2})} \tag{3}$$

where p^x_{cam}, p^y_{cam} and p^z_{cam} are the coordinates in x, y and z directions respectively. W is the width of the image and H is the height of the image. θ_x and θ_y are the horizontal and vertical angles of field of view, respectively.



(a) Original left image (b) Result image of 2D projection without translation and rotation

(c) Result image of 2D projection with translation and rotation (d) Result image of removing conflict points

Fig. 3: An example of projection from 3D point cloud to 2D image plane

We manually projected the 3D point clouds onto the 2D image plane using Equation 2 and Equation 3 for each of the outdoor images. However, this action causes overlap due to discrete samples of the distance sensor. For example, a sample point taken from the wall may be located inside the sample points of the car object. It causes a conflict that leads to erroneous samples for the neural network. To resolve this issue, we manually eliminated the more distant points in the areas where overlap or collision occurred. These processes are illustrated through an example in Fig. 3.

IV. Experimental Results

We used a single Nvidia Quadro RTX 5000 GPU which has 16 GB of memory in this study. Both networks were implemented in Pytorch and were trained separately with Adam optimizer ($\beta_\mathbf{1} = 0.9, \beta_\mathbf{2} = 0.999$). Images were randomly cropped to $\mathbf{H = 256}$ and $\mathbf{W = 512}$ during training process. In PSMNet, we were able to set only two batch sizes because of the limitations of GPU memory. We trained the network up to 600 epochs to get enough good results. Learning rates began at 0.001 for the first 250 epochs and 0.0001 for the remaining 350 epochs. In FADNet, on the other hand, batch size was set at four. We were able to set two more because of FADNet uses less memory compared to PSMNet. We set the learning rate of FADNet as 0.001 for the first 100 epochs, then 0.0001 for the remaining 250 epochs.

As shown in Fig. 4, loss changes show that both networks are learning and losses are gradually decrease as the epoch goes. Since train loss and test loss curves are similar, i.e. no situation such as test loss curve increases while train loss curve

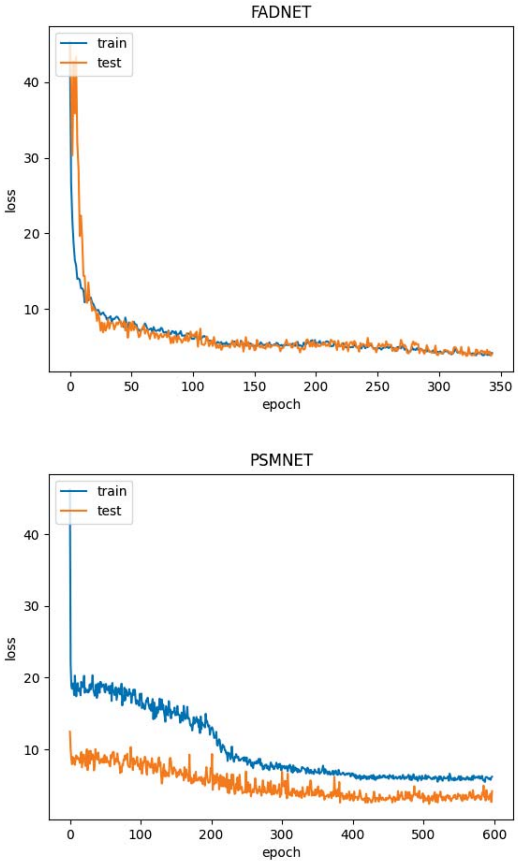


Fig. 4: Illustration of loss change in both networks

decreases, there is no over-fitting problem. As shown in Table I, the loss value of FADNet is smaller than PSMNet in both training and testing processes.

TABLE I: Average end point error (EPE) in both networks

	FADNet	*PSMNet*
Train Loss	4.16	5.7
Test Loss	3.8	4.17

We analyzed the results as quantitative and qualitative in the following sections.

A. Quantitative Results

We used KITTI 2015 benchmark for the evaluation results. We observed better quantitative results in both training set and test set as shown in Table II. Since there is some mismatch between referenced image (left thermal image) and the projected depth data in the dataset, these errors are big. This mismatch consists of two main reasons. The first reason, since the range sensor is used in the dataset, there might be some alignment problems, such that one scene in the dataset is scanned in about 8 minutes [15]. Therefore, some differences can exist between thermal image and 3D LiDAR coordinates. The other reason is that since we are manually projecting 3D points onto 2D image plane, we might have some matching errors. On the other hand, even if we are referencing normalization according to a particular point, this normalization may not be completely accurate for all points.

TABLE II: Evaluation Results on KITTI 2015 Benchmark

	>3 px		**>4 px**		**>5 px**	
	FADNet	PSMNet	FADNet	PSMNet	FADNet	PSMNet
Train set	**41.1 %**	42.9 %	**24.7 %**	29.9 %	**15.9 %**	22.2 %
Test set	**44.3 %**	45.6 %	**29.1 %**	30.0 %	**20.0 %**	22.0 %

B. Qualitative Results

By looking at the results of the validation set as shown in Fig. 5, FADNet is better resulted as in quantitative results. FADNet gives better details of objects like wheel of a bicycle or car, human shape etc. In most of the results, we can select close objects when we look at the output in both networks. As looking at distant objects, object detection in depth map getting harder, however, if regular things exist like a wall or road, still the results are good.

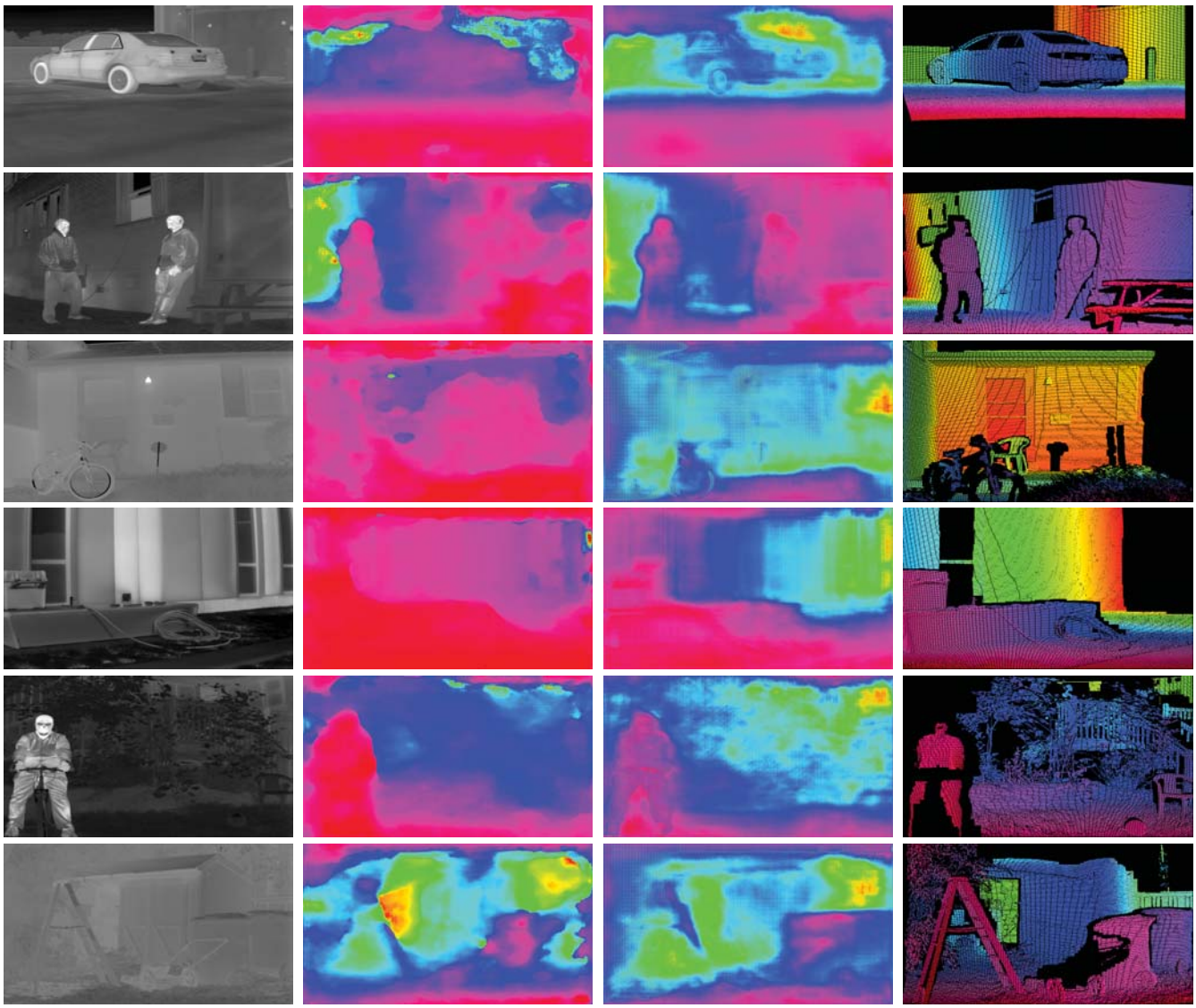
V. Discussion & Conclusion

Thermal images suffering from textureless regions and repetitive pixels, and this makes harder to refine depth map. Even though the situation is like this, we have shown that using thermal-thermal image pairs can be used for depth estimation besides color or gray-scale image pairs. The results are promising for further studies. We believe that the results would be much more satisfactory if we could feed the network with more examples during the training process. On the other hand, minimizing the error in the preliminary study of the dataset would not surprise us to get better results. If the network can be trained with a dataset generated from a particular scene, such as road images, as in KITTI, it will again give better results. In our case, the CATS dataset uses many different scenes consisting of many objects that complicate the learning process.

As a conclusion, we shed light on obtaining depth map estimation with the infrared spectrum as well as the visible spectrum. Namely, there is a potential to take advantage of using thermal stereo pairs. In this way, depth estimation can be done much better by using a combination of stereo visible images and stereo thermal images. On the other hand, it can be used for depth estimation in cross spectra such as color-thermal stereo pairs. In any case, a much more accurate depth estimation can be made if the neural network structure is configured using segmentation-based CNNs and/or detecting the morphological features of the edges of two images taken in different spectra.

VI. Acknowledgment

We would like to thank ASELSAN for their support.



(a) Left Thermal image (b) PSMNet output (c) FADNet output (d) Ground Truth

Fig. 5: Results of depth estimation. (a) shows the left input image of stereo image pair. For each input image, the depth maps obtained by (b) PSMNet and (c) FADNet as shown. (d) shows the ground truth

REFERENCES

[1] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[2] J. Žbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," 2016.

[3] A. Shaked and L. Wolf, "Improved stereo matching with constant highway networks and reflective confidence learning," 2016.

[4] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," 2018.

[5] Q. Wang, S. Shi, S. Zheng, K. Zhao, and X. Chu, "Fadnet: A fast and accurate network for disparity estimation," 2020.

[6] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," *CoRR*, vol. abs/1512.02134, 2015. [Online]. Available: http://arxiv.org/abs/1512.02134

[7] D. Scharstein, R. Szeliski, and R. Zabih, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," in *Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001)*, 2001, pp. 131–140.

[8] G. Yang, X. Song, C. Huang, Z. Deng, J. Shi, and B. Zhou, "Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 899–908.

[9] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361.

[10] A. Geiger, F. Moosmann, O. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 3936–3943.

[11] K. R. Beier, R. Boehl, J. Fries, W. Hahn, D. Hausamann, V. Tank, G. Wagner, and H. Weisser, "Measurement and modeling of infrared imaging systems at conditions of reduced visibility (fog) for traffic applications," in *Characterization and Propagation of Sources and Backgrounds*, W. R. Watkins and D. Clement, Eds., vol. 2223, International Society for Optics and Photonics. SPIE, 1994, pp. 175 – 186. [Online]. Available: https://doi.org/10.1117/12.177911

[12] A. Dhua, F. Cutu, R. Hammoud, and S. Kiselewich, "Triangulation based technique for efficient stereo computation in infrared images," in *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683)*, 2003, pp. 673–678.

[13] M. Bertozzi, A. Broggi, A. Lasagni, and M. Rose, "Infrared stereo vision-based pedestrian detection," in *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.*, 2005, pp. 24–29.
[14] Y. W. K. Zoetgnande, G. Cormier, A.-J. Fougères, and J.-L. Dillenseger, "Sub-pixel matching method for low-resolution thermal stereo images," 2019.
[15] W. Treible, P. Saponaro, S. Sorensen, A. Kolagunda, M. O'Neal, B. Phelan, K. Sherbondy, and C. Kambhamettu, "Cats: A color and thermal stereo benchmark," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
[16] J. Mallon and P. F. Whelan, "Projective rectification from the fundamental matrix," *Image Vision Comput.*, vol. 23, no. 7, p. 643–650, Jul. 2005. [Online]. Available: https://doi.org/10.1016/j.imavis.2005.03.002
[17] E. Trucco and A. Verri, *Introductory techniques for 3-D computer vision.*, 01 1998.