TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

STEREO DEPTH ESTIMATION USING CONVOLUTIONAL SPATIAL PROPAGATION NETWORKS

Paulami Banerjee

Department of Computer Science - Computer Vision

2022

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

INF-73-84-L-7

STEREO DEPTH ESTIMATION USING CONVOLUTIONAL SPATIAL PROPAGATION NETWORKS

Submitted in Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science
of the Technische Universität Kaiserslautern
under the supervision of
Dr. Muhammad Zeshan Afzal

by

Paulami Banerjee

Department of Computer Science - Computer Vision
2022

Abstract

In this project, we have attempted to implement the three-dimensional Convolutional Spatial Propagation Network (CSPN) as proposed by Cheng et al. in their work "Learning Depth with Convolutional Spatial Propagation Network" [1] to obtain disparity maps for Stereo Matching. CSPN is a robust and simple linear propagation model which accomplishes the propagation by a series of recurrent convolutional operations. A deep convolutional neural network (CNN) helps learn the affinity between neighboring pixels.

Acknowledgments

I'd like to express my heartfelt gratitude to everyone who assisted and supported me throughout this project. I am indebted to Prof. Dr. Muhammad Zeshan Afzal for providing me with the opportunity to do my project work in the Computer Vision group and for his support throughout the process. I could not have asked for a better supervisor during the entire period of working on my project. I am indebted to God Almighty and my family for their unwavering support throughout my life. I am also indebted to all my friends for their unwavering support, assistance, and motivation throughout the process.

Contents

Abstract i

Acknowledgments ii

Contents iii

List of Figures v

1 Introduction 1

1.1 Stereo Matching 1

1.2 Classical Stereo Matching Approaches 2

1.3 State-of-the-Art Approaches 3

1.4 Motivation 3

1.5 Report Structure 4

2 Relevant Work 5

3 Model Details 7

3.1 Disparity Calculation 7

3.2 Cost Volume 7

3.3 Pyramid Stereo Matching Network (PSMNet) 8

3.4 Convolutional Spatial Propagation Network (CSPN) 9

3.5 3D CSPN 10

4 Experiment 12

4.1 Implementation Details 12

4.2 Dataset 14

5 Conclusion **15**

5.1 Further Works on Stereo Matching 15

List of Figures

1.1 A Stereo Camera 2
1.2 An example of Left-Right Image Pair from a Stereo Camera. 3

3.1 Fundamental Structure of Stereo Vision 8
3.2 PSMNet Architecture Overview. Image Courtesy: [6] 9
3.3 2D CSPN. Image Courtesy [1] 10
3.4 3D CSPN. Image Courtesy [1] 11

4.1 Network Architecture for Stereo Matching Implementation 12
4.2 3D Module in Figure 4.1 13

Chapter 1

Introduction

Depth Estimation is the process of determining the distance between each pixel and the camera. The most widely used technique for estimating depth is LiDAR. This involves using a laser to determine ranges (variable distances) by targeting an object or a surface and recording the time it takes for the reflected light to return to the receiver. However, LiDAR is a costly process. The less expensive option is to estimate depth using a stereo camera and a technique called Stereo Matching.

1.1 Stereo Matching

Stereo Matching is a critical component of stereo vision because it allows for the recovery of 3D structures in the real world from 2D images. This field has inspired many researchers and mathematicians to develop novel algorithms to ensure the stereo systems' output is accurate. This system is especially advantageous in the area of robotics [2], augmented reality, and self-driving vehicles [3], among others. It provides them with a three-dimensional understanding of the scene by estimating object depths. Given a pair of rectified stereo images taken using a stereo camera, Stereo Matching helps calculate per pixel disparity in a reference image.

Stereo images are captured using a Stereo Camera. Such a camera consists of two or more image sensors. This enables the camera to simulate human binocular vision and

thus perceive depth. Figure 1.1 shows a generic stereo camera with two image sensors.

Figure 1.1: A Stereo Camera

1.2 Classical Stereo Matching Approaches

A conventional method for estimating disparity consists of the following steps:

1. Extrapolating image features to gain more informative data than raw color intensities and to enhance point matching.

2. Construction of the cost volume to estimate the degree to which the left and right feature maps match at various disparity levels. Absolute intensity differences or cross-correlation can also be used, for example.

3. Utilization of the disparity computation module to determine the disparity from the cost volume. For instance, it could be a brute-force algorithm that seeks the disparity level at which the left and right feature maps are most similar. Figure

Figure 1.2: An example of Left-Right Image Pair from a Stereo Camera.

1.2 shows a left and right image pair from a stereo camera.

4. Refining the initially predicted disparity map in case it is too coarse.

1.3 State-of-the-Art Approaches

The evolution of deep networks [4] has had a significant role in developing the current Stereo State-of-the-Art (SOTA) methods. GCNet [5] learns to include geometric context directly from the data by utilizing 3D Convolutions (3DConv) over the disparity cost volume. PSMNet [6] uses a similar concept but induces extensions at the scale-space level through appending a spatial feature pooling [7] module after the feature encoder and extracting multi-scale outputs from their stacked hourglass networks [8] via 3DConv.

1.4 Motivation

Cheng et al. proposed Convolutional Spatial Propagation Networks (CSPNs) that directly learn the affinity from images in their work [1]. It has been proved to be more robust than prior SOTA propagation technique [9] for depth estimation without sacrificing the linear propagation's stability. It extends CSPN to depth completion by

embedding the sparse depth samples provided into the propagation process, ensuring that the resulting depth map retains the sparse input depth values. The CPSN is then transformed into a three-dimensional CSPN for stereo depth estimation, investigating correlations in discrete disparity and scale space. It enables the recovered stereo depth to generate additional details and avoids matching errors caused by noisy appearances caused by sunlight or shadows. Additionally, the work discusses spatial pyramid pooling (SPP) from a CSPN perspective and proposes a more efficient SPP module that improves depth completion and stereo performance. We have attempted to implement the three-dimensional CSPN concept for stereo matching in this project.

1.5 Report Structure

The rest of this report is structured as follows. Chapter 2 discusses some of the related works on Stereo Matching. Chapter 3 discusses the CSPN module's structure and its transformation into a three-dimensional CSPN. Additionally, it discusses the structure of the base deep network that was used to generate the disparity maps. Chapter 4 discusses the integration of the 3D CSPN module into the base network. Chapter 5 is the concluding chapter and provides an overview of subsequent research on this subject.

Chapter 2

Relevant Work

Stereo depth estimation has been a key challenge in computer vision for an extended period of time. Scharstein and Szeliski [10] have traditionally provided a taxonomy of stereo algorithms. Zbontar and LeCun [4] pioneered stereo matching by introducing CNNs to replace the matching cost computation. Their method demonstrated that by incorporating CNNs into the matching process, it was possible to achieve SOTA results over KITTI Stereo benchmarks. But the networks continue to require post-processing refinement. Following that, several methods for increasing computational efficiency [11][12] were proposed, as well as for matching cost accuracy [13] with a more robust network and confidence predictions. Later works incorporated high-level knowledge from objects such as Displets [14] to focus on post-processing.

The need to create an entirely learnable architecture that does not require manual processing stems from this. By adding two corresponding frames, FlowNet [15] is capable of determining two-dimensional optical flow. It can easily be extended to stereo matching by restricting the search to the epipolar line. PWCNet [16] employs a similar strategy but calculates cost volumes using a pyramid warping strategy within a local region of size d. However, due to the epipolar constraint, one may only consider a limited range for disparity matching in stereo estimation. To model per-pixel disparity matching more accurately, GCNet [5] proposes to generate a 3D cost volume of size h by densely comparing the feature at pixel (i, j) in the reference image to all possible

matching pixels within the epipolar line in the target image. Through a soft-argmin operation, the network can determine the best matching disparity.

PSMNet [6], which incorporates semantic segmentation experience and exploits scale-space via pyramid spatial pooling and hourglass networks to capture global image context, outperforms GCNet. Admittedly, both GCNet and PSMNet add disparity space and scale space as a third dimension and get better results. Based on PSMNet, 3D CSPN considers the relationship between diffusion and their proposed new dimension, resulting in more robust results.

Chapter 3

Model Details

The model comprises two modules: a base network resembling the PSMNet and a 3D CSPN module. The researchers used the PSMNet as a baseline for their evaluations in the original publication. We will discuss disparity calculation and cost volume in greater detail in this chapter, followed by the structure of the PSMNet, CSPN, and how it is transformed into the three-dimensional CSPN.

3.1 Disparity Calculation

The stereo camera's perspective is likewise quite comparable to that of the human eye. The distance D between the target object and the cameras can be computed using the equation $D = \frac{f*B}{d}$, which involves the camera focal length, f, and disparity, d, where the disparity value, d, is determined by the location difference of the target object between the reference and target images. Figure 3.1 illustrates the basic structure of stereo vision. The output from a stereo matching algorithm is the disparity map.

3.2 Cost Volume

While looking for the most comparable sibling of a single 2D patch in another image, each pixel is a candidate. If we write their similarity in another image, we get a 2D image with similarities. We can refer to it as a similarity surface or a cost surface if we

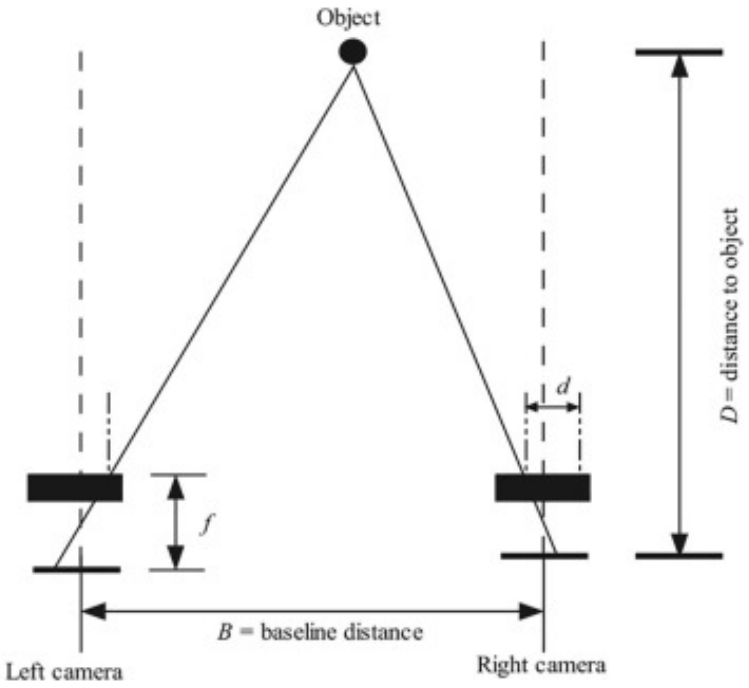

Figure 3.1: Fundamental Structure of Stereo Vision

include distances.

The cost, or distance, between a feature in one image and all the pixels in a window around it, is stored for a WxH image. Given that we have WxH pixels and a $DXxDY$ window, the complete array of expenses is $WxHxDXxDY$. Thus, it is four-dimensional but is referred to as a "cost volume" by analogy.

3.3 Pyramid Stereo Matching Network (PSMNet)

The Pyramid Stereo Matching network (PSMNet) [6] comprises two major modules: spatial pyramid pooling (SPP) and three-dimensional convolutional neural networks. The spatial pyramid pooling module leverages the global context's potential for information by collecting context at many scales and places to create a cost volume. The cost volume is regularized by the 3D CNN which learns due to the stacking of the hourglass networks along with interim supervision.

The PSMNet's architecture is depicted in Figure 3.2. The left and right stereo images are fed into two weight-sharing pipelines, each of which contains a CNN for calculating

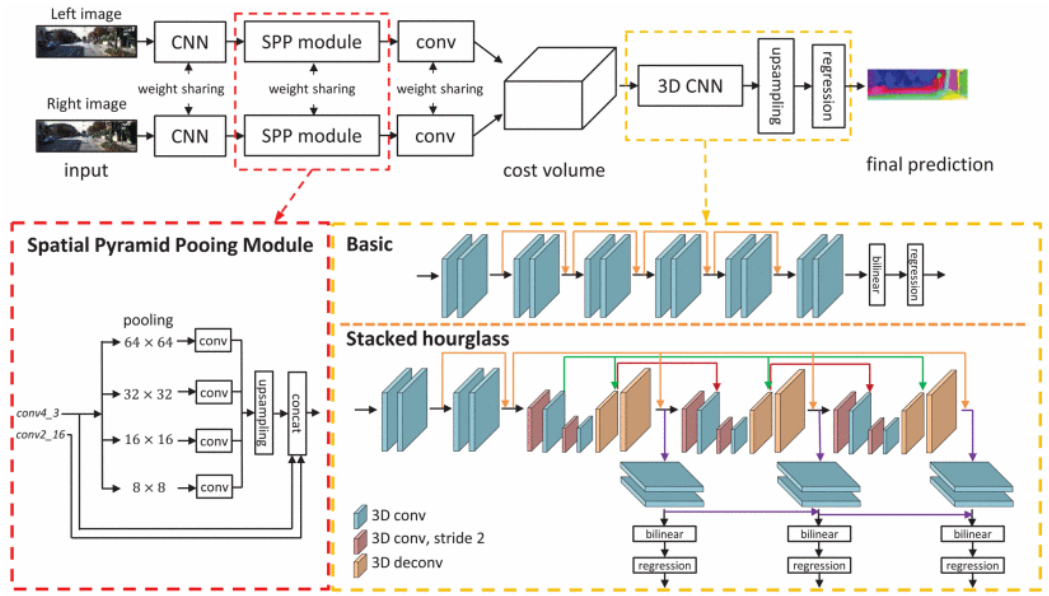

Figure 3.2: PSMNet Architecture Overview. Image Courtesy: [6]

feature maps, an SPP module for harvesting features by concatenating representations from sub-regions of varying sizes, and a convolution layer for feature fusion. The left and right image data are then combined to create a four-dimensional cost volume fed into a three-dimensional convolutional neural network for regularizing the cost volume and predicting disparity values.

3.4 Convolutional Spatial Propagation Network (CSPN)

The task is to update an existing depth map $D_0 \in \mathbf{R}^{mxn}$ and an image $X \in \mathbf{R}^{mxn}$ in N iteration steps to a new depth map D_n. This reveals additional structural information and improves the per-pixel depth calculation results.

Figure 3.3 illustrates this updating method in two dimensions. Without sacrificing generality, the depth map $D_0 \in \mathbf{R}^{mxn}$ can be formalized as embedded in some hidden space $H \in \mathbf{R}^{mxnxc}$, where c is the number of feature channels. For each time-step t, the convolutional transformation functional with a kernel size of k might be expressed as,

$$H_{i,j,t+1} = \kappa_{i,j}(0,0) \odot H_{i,j,0} + \sum_{a,b=-(k-1)/2}^{(k-1)/2} \kappa_{i,j}(a,b) \odot H_{i-a,j-b,t} \qquad (3.1)$$

where $a, b \neq 0$, $\kappa_{i,j}(a,b) = \frac{\hat{\kappa}_{i,j}(a,b)}{\sum_{a,b,a,b\neq 0} |\hat{\kappa}_{i,j}(a,b)|}$, and $\kappa_{i,j}(0,0) = 1 - \sum_{a,b,a,b\neq 0} \kappa_{i,j}(a,b)$.

The transformation kernel $\hat{\kappa}_{i,j} \in \mathbf{R}^{kxkxc}$ is the output of a spatially dependent affinity network on the input image. The kernel size k is typically an odd number to maintain the symmetry of the computational context surrounding the pixel (i, j). The kernel weights are normalized to the range $(-1, 1)$, as in Spatial Propagation Networks (SPN), so that the model can be stabilized when the requirement $\sum_{a,b,a,b\neq 0} |\kappa_{i,j}(a,b)| \leq 1$ is satisfied. After then, N iterations are undertaken to achieve a stable state.

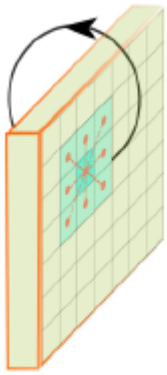

Figure 3.3: 2D CSPN. Image Courtesy [1]

3.5 3D CSPN

The CSPN is then extended to three dimensions in order to process the three-dimensional cost volumes that are frequently utilized for a stereo estimate. Figure 3.4 depicts the 3D CSPN. Similar to Equation 3.1, given a three-dimensional feature volume $H \in \mathbf{R}^{dxmxnxc}$, where d stands for the disparity space, the 3D CSPN can be formed as:

$$H_{i,j,l,t+1} = \kappa_{i,j,l}(0,0,0) \odot H_{i,j,l,0} + \sum_{a,b,c=-(k-1)/2}^{(k-1)/2} \kappa_{i,j,l}(a,b,c) \odot H_{i-a,j-b,l-c,t} \tag{3.2}$$

where $a, b, c \neq 0$, $\kappa_{i,j,l}(a,b,c) = \frac{\hat{\kappa}_{i,j,l}(a,b,c)}{\sum_{a,b,c|a,b,c\neq 0} |\hat{\kappa}_{i,j,l}(a,b,c)|}$, and

$\kappa_{i,j,l}(0,0,0) = 1 - \sum_{a,b,c|a,b,c\neq 0} \kappa_{i,j,l}(a,b,c)$.

If we compare this to Equation 3.1, we can see that the only difference it has is the addition of disparity as an additional dimension for propagation. All the original

theoretical features are preserved.

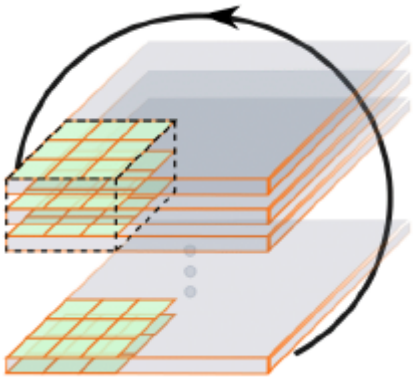

Figure 3.4: 3D CSPN. Image Courtesy [1]

Chapter 4

Experiment

The network design is analogous to that of the PSMNet [6], as seen in Figure 3.2. To reiterate, the left and right stereo images are fed into two weight-sharing pipelines, each of which contains a CNN for calculating feature maps, an SPP module for harvesting features by concatenating representations from sub-regions of varying sizes, and a convolution layer for feature fusion. The left and right image data are then combined to create a four-dimensional cost volume fed into a three-dimensional convolutional neural network for cost volume regularization and disparity regression. After the multi-scale outputs, the 3D CSPN module is added to this network.

4.1 Implementation Details

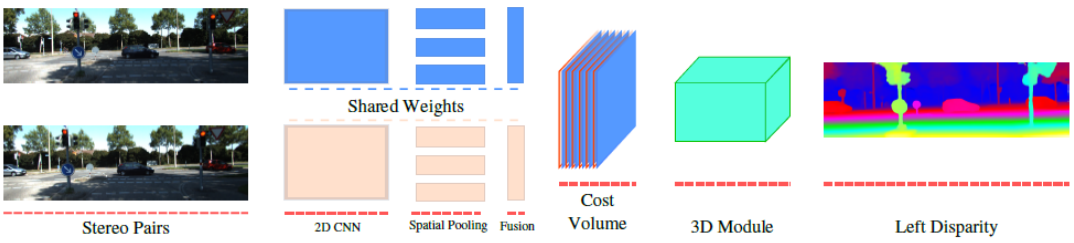

Figure 4.1: Network Architecture for Stereo Matching Implementation

Figure 4.1 and 4.2 illustrates the base network used for the implementation of the proposed Stereo Matching method and the corresponding 3D module in the network.

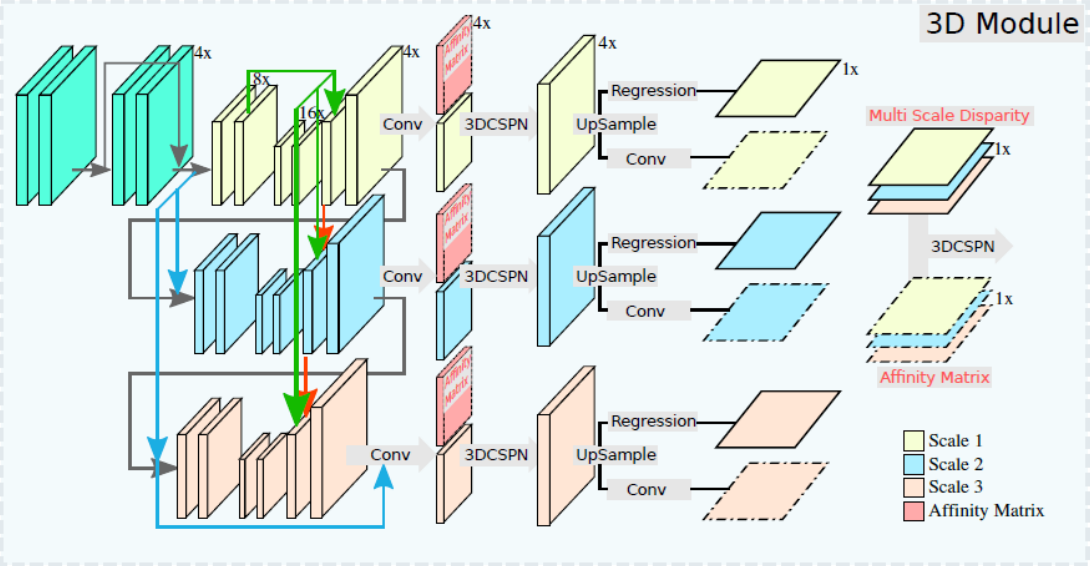

Figure 4.2: 3D Module in Figure 4.1

Each block in Figure 4.2 has the downsampling rate relative to the image size indicated in the right-top corner. For instance, $4x$ indicates that the feature map is $\frac{h}{4}x\frac{w}{4}$ in size, with hxw being the picture size. The red, green, and blue arrows represent skip connections, which indicate feature concatenation at a certain place, and are identical to those used in PSMNet.

The three disparity volumes of size $\frac{d}{4}x\frac{h}{4}x\frac{w}{4}x1$ are predicted as output by a stacked hourglass network at different stages of PSMNet. Here, d, h, w represent the input image's maximum disparity, height, and width, respectively. To merge the contexts from neighboring pixels, a 3D CSPN with kernel size $kxkxk$ is inserted, where the affinity matrix is learned from the same feature block as the outputs. An Affinity Matrix structures the similarities between a collection of data elements. Then, using trilinear upsampling, a disparity volume is upsampled to $dxhxwx1$ for disparity map regression, resulting in an output with the shape of $hxwx1$. 3DCSPN completes its processing of the disparity space at this point (ds).

PSMNet manually adjusts the weight to average the outputs of the numerous disparity maps from different phases. The numerous outputs are concatenated into a 4D volume with the dimensions $sxhxwx1$, where $s = 3$ is the number of disparity maps.

Similarly, to connect the multi-stage predictions, a 3D CSPN with a kernel size of

$sxkxk$ can be done. Finally, feature padding with a size of $[0, 1, 1]$ is performed to decrease the first dimension to one iteration and produce a single regressed disparity map with the shape $hxwx1$ for the final depth estimation. At this point, 3DCSPN completes its analysis of the scale-space (ss).

The entire network is trained using $soft - argmin$ disparity regression to transform the discrete values to continuous disparity values. Using the L1 loss, this continuous disparity value is compared to the ground truth. The loss function is:

$$L(d^*, \hat{d}) = \frac{1}{N} \sum_{i=1}^{N} ||d^* - \hat{d}||_1, \tag{4.1}$$

where the ground truth disparity and the predicted continuous disparity values are represented by d^* and $\hat{d}$, respectively.

4.2 Dataset

We used the Scene Flow Dataset for our experiment with 3D CSPN. This dataset consists of 35454 training and 4370 test stereo pairs, synthetically rendered in 960x540 pixel resolution. Color normalization is performed on the dataset as data preprocessing.

We encountered problems while reading the .pfm disparity files provided in the Scene Flow dataset. Hence we converted these files to their corresponding numpy arrays and continued.

Chapter 5

Conclusion

We attempted to implement the 3D CSPN module in this project by appending it to the Pyramid Spatial Network using the Scene Flow Dataset and the Tensorflow framework. The 3D CSPN can diffuse along both the disparity and scale dimensions.

5.1 Further Works on Stereo Matching

Stereo matching remains a complex problem to solve to this day. Xue et al. [17] present a multi-frame narrow-baseline stereo matching method based on edge extraction and matching across multiple frames. Edge matching enables the user to focus immediately on the essential features and deal with occlusion boundaries and untextured regions.

Fu and Liang [18] perform stereo matching in order to reconstruct a three-dimensional face. They used the spatial-temporal integral image (STII) to accelerate the computation of matching costs during the stereo matching process. A similar three-dimensional depth perception study was also conducted using the mantis vision system [19]. They discovered that as the resolution of the image decreases, insect stereopsis becomes more efficient and robust and more responsive to deviations in the pattern of luminance between the two eyes. When we take advantage of the unique characteristics of the natural elements around us, the research works mentioned above on stereo vision will pave the way for new frontiers in the stereo matching field.

Bibliography

[1] Xinjing Cheng, Peng Wang, and Ruigang Yang. "Learning Depth with Convolutional Spatial Propagation Network". In: *CoRR* abs/1810.02695 (2018). arXiv: 1810.02695. URL: http://arxiv.org/abs/1810.02695.

[2] Don Murray and J.J. Little. "Using Real-Time Stereo Vision for Mobile Robot Navigation". In: *Auton. Robots* 8 (Apr. 2000), pp. 161–171. DOI: 10.1023/A:1008987612352.

[3] Chenyi Chen et al. "DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 2722–2730. DOI: 10.1109/ICCV.2015.312.

[4] Jure Žbontar and Yann LeCun. "Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches". In: *J. Mach. Learn. Res.* 17.1 (Jan. 2016), pp. 2287–2318. ISSN: 1532-4435.

[5] Alex Kendall et al. "End-to-End Learning of Geometry and Context for Deep Stereo Regression". In: (Mar. 2017).

[6] Jia-Ren Chang and Yong-Sheng Chen. "Pyramid Stereo Matching Network". In: (Mar. 2018).

[7] Kaiming He et al. "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition". In: *Computer Vision – ECCV 2014*. Ed. by David Fleet et al. Cham: Springer International Publishing, 2014, pp. 346–361.

[8] Alejandro Newell, Kaiyu Yang, and Jia Deng. "Stacked Hourglass Networks for Human Pose Estimation". In: *Computer Vision – ECCV 2016*. Ed. by Bastian Leibe et al. Cham: Springer International Publishing, 2016, pp. 483–499.

[9] Sifei Liu et al. “Learning Affinity via Spatial Propagation Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/c22abfa379f38b5b0411bc11fa9bf92f-Paper.pdf.

[10] Daniel Scharstein and R. Szeliski. “A Taxonomy And Evaluation Of Dense Two-Frame Stereo Correspondence Algorithms”. In: *International Journal of Computer Vision - IJCV* 47 (Jan. 2000), pp. 7–42.

[11] Yiliu Feng, Zhengfa Liang, and Hengzhu Liu. “Efficient deep learning for stereo matching with larger image patches”. In: *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. 2017, pp. 1–5. DOI: 10.1109/CISP-BMEI.2017.8301999.

[12] Wenjie Luo, Alexander G. Schwing, and Raquel Urtasun. “Efficient Deep Learning for Stereo Matching”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 5695–5703. DOI: 10.1109/CVPR.2016.614.

[13] Amit Shaked and Lior Wolf. “Improved Stereo Matching with Constant Highway Networks and Reflective Confidence Learning”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 6901–6910. DOI: 10.1109/CVPR.2017.730.

[14] Fatma Güney and Andreas Geiger. “Displets: Resolving stereo ambiguities using object knowledge”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 4165–4175. DOI: 10.1109/CVPR.2015.7299044.

[15] Eddy Ilg et al. “FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1647–1655. DOI: 10.1109/CVPR.2017.179.

[16] Deqing Sun et al. “PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8934–8943. DOI: 10.1109/CVPR.2018.00931.

[17] Tianfan Xue et al. “Multi-frame stereo matching with edges, planes, and superpixels”. In: *Image and Vision Computing* 91 (2019), p. 103771. ISSN: 0262-8856.

DOI: https://doi.org/10.1016/j.imavis.2019.05.006. URL: https://www.sciencedirect.com/science/article/pii/S0262885619300745.

[18] Junwei Fu and Jun Liang. "Virtual View Generation Based on 3D-Dense-Attentive GAN Networks". In: *Sensors* 19.2 (2019). ISSN: 1424-8220. DOI: 10.3390/s19020344. URL: https://www.mdpi.com/1424-8220/19/2/344.

[19] Vivek Nityananda et al. "A Novel Form of Stereo Vision in the Praying Mantis". In: *Current Biology* 28.4 (2018), 588–593.e4. ISSN: 0960-9822. DOI: https://doi.org/10.1016/j.cub.2018.01.012. URL: https://www.sciencedirect.com/science/article/pii/S0960982218300149.