Unsupervised Monocular Depth Estimation for Autonomous Driving

Chih-Hsuan Huang, Wan-Nung Tsung, Wei-Jong Yang, Chin-Hsing Chen
Dept. of Electrical Engineering, National Cheng Kung University, Tainan, 701, Taiwan (R. O. C.)
Keywords: Autonomous Driving, Depth Estimation, Disparity, 3D image.

ABSTRACT

3D technology with range information has become a staple requirement in computer vision. For this reason, we believe that the depth information can effectively improve the vision capabilities for many applications. In this paper, we proposed an unsupervised monocular depth estimation network to extract the depth map of street views.

1 INTRODUCTION

To get depth maps from a single view image, which is known as monocular depth estimation, is an important technique in computer vision with a long history. It is often described as an ill-posed and inherently ambiguous problem. Most existing approaches treat the depth prediction as a supervised regression problem as a result, they require vast quantities of corresponding ground truth depth data for training [1]-[4]. However, to collect those ground truth data is time consuming. Motivated by [2], we adopt stereo images as our training data, and compute the reconstruction loss according to the disparity maps which are the output of the network. As the result, we use transfer learning to pretrain the network with limited edge ground truth, and then use large amount of stereo training data to fine the network in an unsupervised way during training. We can replace the use of explicit depth data with easier-to-obtain binocular stereo footage. Generating disparity images by training the network with an image reconstruction loss in an unsupervised way is more reasonable without complete data sets.

Moreover, the most existing developments of unsupervised monocular depth estimation (MDE) suffer from the problem of blurring depth maps. To overcome this problem, inspired by [3], we thus add some edge information of the ground truth depth to train the whole model in a semi-supervised way. As aforementioned, recording quality depth data is a challenging problem.

2 RELATED WORK

The depth estimation networks generally can be separated as supervised monocular depth estimation and unsupervised monocular depth estimation. The most common way is to treat it as a supervised regression problem. Eigen [1] has proposed a network that produces dense pixel depth maps using a two-scale DCNN, trained with images and their corresponding depth value. Several later proposed networks have been designed based on this method. The supervised method has great performance but needs a great deal of data.

In order to overcome the disadvantage of collection of ground truth data. Godard [2] has introduced an unsupervised network using the epipolar geometry constraints, which is the property of binocular stereo images, and have trained the network with an image reconstruction loss. The reconstruction process is to generate a synthetic right view image according to the left view and its estimated disparity with the technique of wrapping. Because the binocular visual database of street view is difficult to collect. So, we proposed an unsupervised depth estimation network with the edge guided network to overcome the problem.

3 THE PROPOSED SYSTEM

In the proposed system, as illustrate in Fig. 1, the depth estimation is based on two sub networks, the unsupervised depth estimation network and the edge guided network. I^L stands for the input image, and D^L, D^R, E^h, E^v are the corresponding output, which represent estimated disparity, estimated horizontal edge and vertical edge respectively.

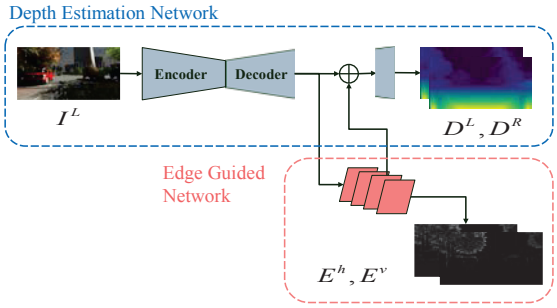

Fig. 1 Proposed edge guided semi-supervised monocular depth estimation system.

3.1 Unsupervised Depth Estimation Network

Inspired by [2], the target of our depth estimation network is to estimate two kinds of disparity maps, disparity of left and right view image, denoted as D^L = [D_1^L, D_2^L, D_3^L, D_4^L] and D^R = [D_1^R, D_2^R, D_3^R, D_4^R] separately. The disparity maps can reconstruct synthetic images by doing wrapping and further infer the depth maps as seen in Fig. 3.

To compute the differences between synthetic images and input images according to the reconstruction process, we treat our depth estimation network as an unsupervised network. The reconstruction process also

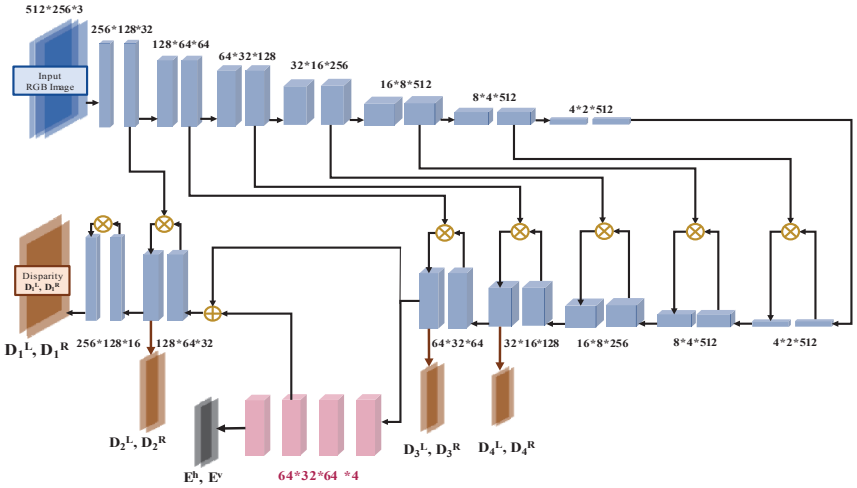

Fig. 2 Details of the whole network

shows that we need stereo input images during training but require only one-view images in testing. Furthermore, our depth estimation network, shown in Fig. 2, is composed of an hourglass network, which is divided into two parts - encoder and decoder. The upper part of our network is the depth estimation network, composed of an encoder modular with seven convolutional blocks, which implement downscaling each block with stride-two convolution, and a decoder modular with seven deconvolution blocks. The lower part is the edge-guided network with four convolutional layers. We also use the concept of U-net [5] to deploy skip connections between every pair of corresponding layers, which can avoid losing lower features during multiple convolutional layers. Following [6], we replaced the usual deconvolutions with a nearest neighbor upscaling followed by a convolutional layer to reduce the chessboard effect.

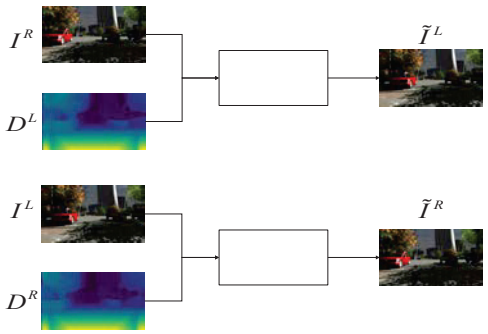

Fig. 3 The image reconstruction process of backward mapping

3.2 Edge Guided Network

The edge guided network is designed to emphasize contour regions, since our target is to reduce the blurry effect of objects' contour. Considering that we hope our network to learn the features of edge, instead of directly using limited depth ground truth, we generate the horizontal and vertical edge ground truth by Sobel edge detection. In remind of the fact that our architecture is mainly applied on autonomous driving, we need to keep the processing in real-time. We also suppose that extracting edge features might not be too complex in compares with extracting other features that can generate depth maps. Thus, we use only few convolutional layers, as illustrate in Fig. 1, pipelining with the decoder of depth estimation network to refine our final depth.

However, monocular depth estimation relies on large amount of training data to compensate the lack of geometry constrain. Training the whole network with a few binocular images and their corresponding ground truth such as Scene Flow Dataset, can't achieve proper results. As aforementioned, we then take the advantages of transfer leaning to freeze weights in edge guided network, and further train the whole network with larger datasets such as KITTI Dataset [7].

3.3 Training Loss

Our total loss L is a weighted sum of four losses, which are appearance matching loss, disparity smoothness loss, left-right disparity consistency loss, and edge feature loss. The value of each α are set as α_{ap} = 0.85, α_{ds} = 0.1, α_{lr} = 1, α_{ef} = 0.5, empirically.

$$L = \alpha_{ap}(C_{ap}^{L} + C_{ap}^{R}) + \alpha_{ds}(C_{ds}^{L} + C_{ds}^{R}) + \alpha_{lr}(C_{lr}^{L} + C_{lr}^{R}) + \alpha_{ef}C_{ef}, \quad (1)$$

where C_{ap}, C_{ds}, C_{lr}, and C_{ef} are appearance matching, disparity smoothness, left-right consistency, and edge feature lose respectively; R and L denote the right and left view. The detail will be discussing in following.

Appearance Matching Loss

We integrate the image sampler from the spatial transformer network (STN) [8] into our convolutional architecture. The STN, which features locally fully differentiable, can be used to sample input images using disparity maps according to bilinear sampling. To robust our network, the appearance matching loss of depth map is the weighted sum of four different resolutions, that leads to more stable convergence. Inspired by [9], we use a combination of an L1 loss and single scale

structural similarity index (SSIM) [10] term as our photometric image reconstruction cost. As L1 loss, we compare the input image I_{ij}^L with its reconstruction $\tilde{I}_{ij}^L$ as

$$C_{ap}^L = \frac{1}{N}\sum_{i,j}\alpha\frac{1-SSIM(I_{ij}^L,\tilde{I}_{ij}^L)}{2}+(1-\alpha)\left\|I_{ij}^L-\tilde{I}_{ij}^L\right\| \quad (2)$$

where N is the number of pixels and (i, j) denotes the location of reliable pixel and α is set to be 0.85.

Disparity Smoothness Loss

The disparity smoothness loss is designed to make depth map locally smooth.

$$C_{ds}^L = \frac{1}{N}\sum_{i,j}\left|\partial_x d_{ij}^L\right|e^{-\left\|\partial_x I_{ij}^L\right\|}+\left|\partial_y d_{ij}^L\right|e^{-\left\|\partial_y I_{ij}^L\right\|} \quad (3)$$

where d_{ij} denotes the disparity locate on pixel (i, j).

Edge Feature Loss

While generating the edge ground truth, gt^v and gt^h, we record the gradient value in vertical and horizontal direction. Since that, the value of areas which does not seem to be edge is set to be zero. The outputs of our depth estimation network and edge guided network are kind of mutual conflict. So, we ignore the zero-value regions instead of directly computing the loss with whole image.

$$C_{ef} = \frac{1}{N}\sum_{i,j}\left|E_{ij}^v\cdot gt_{ij}^v-gt_{ij}^{v2}\right|+\left|E_{ij}^h\cdot gt_{ij}^h-gt_{ij}^{h2}\right| \quad (5)$$

4 EXPERIMENT

The proposed network, which contains 31.7 million trainable parameters, is implemented in TensorFlow and takes about 9 hours to train. Using GPU-1080Ti on Scene Flow Dataset of 4 thousand images for 80 epochs with batch size 4. The inference is fast and takes less than 38 ms, or more than 26 frames per second, for a 512×256 image, which is nearly real-time. We used an initial learning rate of $\lambda=10^{-4}$, which is kept constant for the first 30 epochs before halving it every 10 epochs until the end.

During the wrapping process, it is instinct that we might face the problem of occlusion and disocclusion. We operate some post-processing on the output. For an input image I at test time, we also compute the disparity map D_l' for its horizontally flipped image I'. By flipping back this disparity map we obtain a disparity map D_l''. And for the final result, we assign the first 5% on the left of the image using D_l'' and the last 5% on the right to the disparities from D_l. The central part of the final disparity map is the average of D_l and D_l'. As illustrated in Fig. 4., after adding the edge guided network, all the contours in the images is much clearer.

We restore the pretrain weights of former network, and then freeze all the weights in edge guided network. In this part, we adopt nearly 30 thousand training data in KITTI Dataset as our training data. Following by its high resolution and large amount of data, we can improve the accuracy comparing to only training with Scene Flow Dataset. As showed in Fig. 5, the proposed network makes the contours of objects be more complete such that it leads to a more precise result.

5 CONCLUSIONS

The most existing developments of MDE have the problem of blurring depth maps. The proposed method adds some edge information to improve the completeness of feature data to obtain more precise depth maps. Furthermore, the proposed network can be trained the whole model in a semi-supervised way to reduce the data requirement.

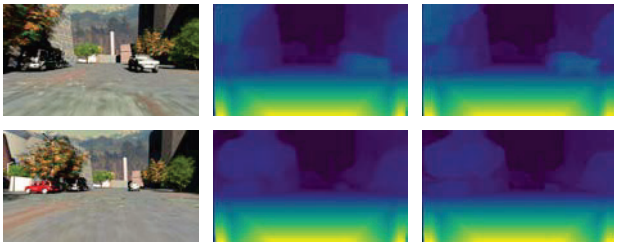

Fig. 4 Testing results after training 80 epochs on Scene Flow Dataset

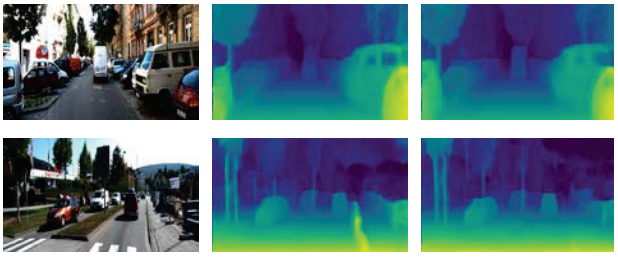

Fig. 5 Testing results after futrher training on KITTI Dataset.

REFERENCES

[1] Eigen, David, Christian Puhrsch, and Rob Fergus. "Depth map prediction from a single image using a multi-scale deep network." Advances in neural information processing systems. 2014.

[2] Godard, Clément, Oisin Mac Aodha, and Gabriel J. Brostow. "Unsupervised monocular depth estimation with left-right consistency." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[3] Kim, Youngjung, et al. "Deep monocular depth estimation via integration of global and local predictions." IEEE Transactions on Image Processing 27.8 (2018): 4131-4144.

[4] Roy, Anirban, and Sinisa Todorovic. "Monocular depth estimation using neural regression forest." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[5] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In Proc. Int. Conf. Medical Image Computing and Computer-Assisted Intervention, pages 234–241. Springer, 2015.

[6] A. Odena, V. Dumoulin, and C. Olah. Deconvolution and checkerboard artifacts. Distill, 2016. 5

[7] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In CVPR, 2012.
[8] Jaderberg, Max, Karen Simonyan, and Andrew Zisserman. "Spatial transformer networks." Advances in neural information processing systems. 2015.
[9] H. Zhao, O. Gallo, I. Frosio, and J. Kautz. Is l2 a good loss function for neural networks for image processing? arXiv preprint arXiv:1511.08861, 2015
[10] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. Transactions on Image Processing, 2004. The performance of S02 “Poznan Street”