2020 Joint 11th International Conference on Soft Computing and Intelligent Systems and 21st International Symposium on Advanced Intelligent Systems (SCIS-ISIS) | 978-1-7281-9732-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/SCISISIS50064.2020.9322780

Monocular Depth Estimation with a Multi-task and Multiple-input Architecture Using Depth Gradient

1st Michiru Takamine
Graduate School of Engineering and Science
University of the Ryukyus
Nishihara-town, Japan
k198579@ie.u-ryukyu.ac.jp

2nd Satoshi Endo
Faculty of Engineering, School of Engineering
University of the Ryukyus
Nishihara-town, Japan
endo@ie.u-ryukyu.ac.jp

***Abstract*—In this paper we address the monocular depth estimation task with a multi-task and multiple-input network architecture. The integrated network that we develop use depth gradient information and can be applied to supervised and unsupervised learning. We confirmed that our architecture improve depth estimation on the supervised learning and improve structure-from-motion on the unsupervised learning.**

***Index Terms*—monocular depth estimation, multi-task learning, multipule-input**

I. Introduction

Scene understanding is a central problem in computer vision that has many different aspects [1]. Specifically, depth estimation is one of the important applications in scene understanding, robotics and 3-D reconstruction. There are three major methods for obtaining depth information; recovering 3D structures from a couple of images based on geometric constraints; using depth sensors, like RGB-D cameras and LIDAR; and deep learning base methods. Due to the low cost, small size and wide applications of monocular cameras, estimating the dense depth map from a single image has received more attention. Moreover, monocular depth estimation has been well researched recently based on deep learning in end-to-end methods [2]. Eigen et al. achieved to show the baseline of monocular depth estimation and broke many assumptions on the input, particularly horizontal alignment of the ground plane [3] [1]. The model name is Multi-Scale Model:MSM. After a while, several models using multi-modal and surrounding knowledge were proposed. Li et al. proposed Two-Streamed Network:TwoNet [4] with multiple-input using depth gradient information. The TwoNet improves accuracies, but because each task uses separate networks, the model size is twice as large as the original MSM. Furthermore, The network cannot obtain features considering multiple tasks. Typical models that combine multi-task learning with depth estimation include Joint Refinement Network:JRN [5] and Geometric Neural Network:GeoNet [6]. These previous researches show the effect of using the surrounding knowledge in depth estimation. On the other hand, these results confirmed that ambiguity of label information adversely affects learning [7] [8]; the specific label area reduces the estimation accuracy. Therefore, multi-task learning and multiple-input using unlabeled information improve the estimation efficiently. However, there are few networks optimized for both of them.

Furthermore, since preparing the ground-true of a RGB-D image is very difficult, depth estimation requires development of unsupervised learning methods. Zhou et al. perform unsupervised monocular depth estimation with SfM Learner [9] using the concept of self-supervised. At that time, the SfM Learner recorded state-of-the-art in the monocular depth estimation, and it became an opportunity for unsupervised learning to attract attention. Meanwhile, unsupervised learning has the disadvantage of high learning costs. Additionally, the unsupervised model cannot reduce the amount of data needed for learning.

In this paper, we propose the integrated network of monocular depth estimation for multi-task and multiple-input using depth gradient information. The integrated network can be applied to supervised and unsupervised learning.

II. Related Work

A. Multi-Scale Model

Multi-Scale Model:MSM is a supervised monocular depth estimation model that was proposed by Eigen et al [1]. The MSM is a pioneer of depth estimation with deep learning based methods. The model has two network stacks; first one makes a coarse global prediction based on the entire image(Fig. 1. Red frame), and second one refines this prediction locally(Fig. 1. Green and blue frame). The names of the network are scale1, scale2, and scale3. In consequence, the monocular depth estimation broke many assumptions on the input, particularly horizontal alignment of the ground plane. In addition to depth, the network can estimate surface normals and semantic labels, uses multi-task learning for surface normals. We use this model as the origin for the supervised learning model of our proposed architecture.

B. Two-Streamed network

Two-Streamed network:TwoNet [4] is a supervised monocular depth estimation model using multiple-input with depth gradient information. The network using the MSM as origin model, and has two network streams; first one estimates depths(Fig. 2. Blue frame), and second one predicts depth gradients(Fig. 2. Red frame). TwoNet obtains refinement depths

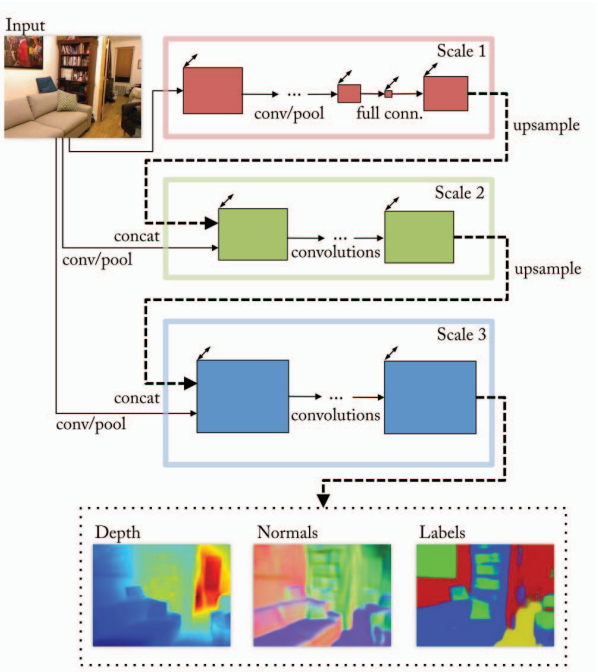

Fig. 1. Multi-Scale Model

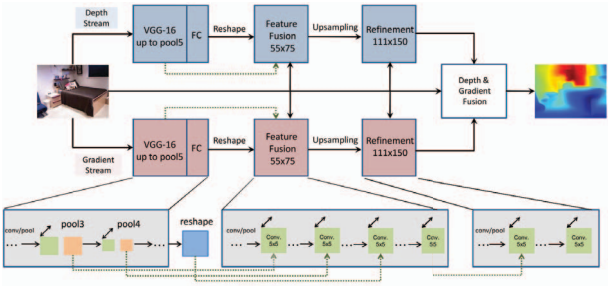

Fig. 2. Two-Streamed network

using both depths and depth gradients as multiple-input(Fig. 2. White frame). This model improves estimation, but because it contains two separate networks, model size is twice as large as the MSM. Furthermore, the network cannot get the features considering both a depth and a depth gradient at the same time. We make the TwoNet compatible with multi-task learning to obtain common features and reduce the model size.

C. SfM Learner

SfM Learner [9] is one of the self-supervised monocular depth estimation model. This model uses only monocular video as the training data and is the basis of many unsupervised monocular depth estimation. The SfM Learner learns reconstructing the scene based on the camera parameters and depths. Namely, the network restores the target image that is a moment after of an input image(Fig. 3. Pose CNN). Self-supervised indirectly estimates depths by setting the task that assumes depth maps as a middle output(Fig. 3. Depth CNN).

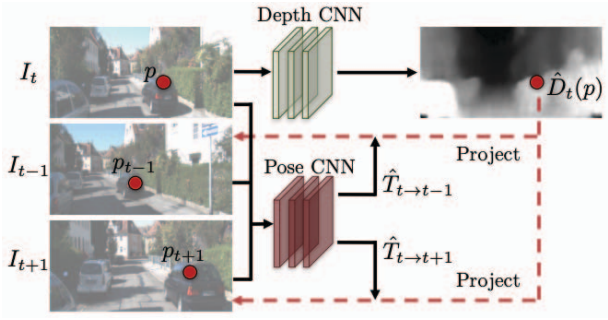

Fig. 3. SfM Learner

We use this model as the origin for the unsupervised learning model of our proposed architecture.

III. Model Architecture

Our model is integrated architecture that uses multi-task learning and multiple-input. The model learns using multi-task learning that is depth estimation and depth gradient estimation. Simultaneously, the model estimates the refinement depth that uses the estimation depth and the estimation depth gradient as multiple-input. Model size is reduced by using multi-task learning compared to the normal multiple-input model. Additionally, we set depth gradient information in the sub-task. In consequence, our architecture can be applied to supervised and unsupervised learning.

A. Depth and Depth Gradient Loss

We referred to the TwoNet and give the following formula(Eq. 1) in addition to the conventional loss when multi-task learning. The loss function maintains depth consistency by accounting for differences between depths and depth gradients. N is the number of valid depth pixels and G^p is estimation depth gradients. We apply the edge filter to the estimation depth, and obtains the generated depth gradient:∇D^p. Moreover, $\phi(x)$ is L1 norm; $\sqrt{x^2+10^{-4}}$.

$$\frac{1}{N}\sum_{p}^{N}[\phi(\nabla_x D^p - G_x^p) \times \phi(\nabla_y D^p - G_y^p)] \qquad (1)$$

B. Supervised learning model

We use the MSM [1] as the origin for the supervised learning model. First, our model branches the path of scale2, and the network obtains common features using multi-task learning. Each scale2 estimates depth maps and depth gradients(Fig. 4. Green frame). Simultaneously, scale2 estimates the depth gradients in two directions; x-axis and y-axis. Depth's scale2 network and grad's scale2 network share the weights of the middle layer, and features(Fig. 4. Pink frame). Next, we concatenates estimation depth maps and estimation depth gradients to make multiple-input scale3(Fig. 4. Orange frame). The model uses the depth and the depth gradient to complement the information, improving the final estimation accuracy.

978-1-7281-9732-6/20/$31.00 ©2020 IEEE

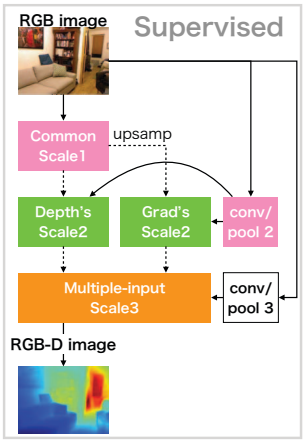

Fig. 4. Our model for supervised learning

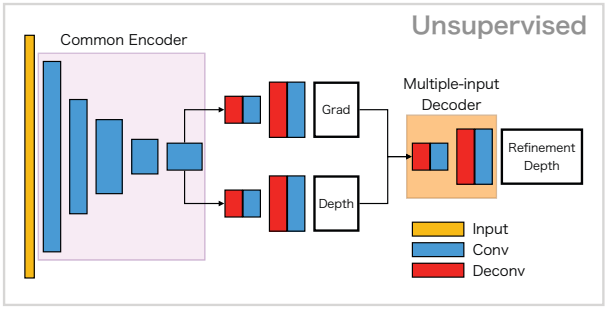

Fig. 5. Our model for unsupervised learning

C. Unsupervised learning model

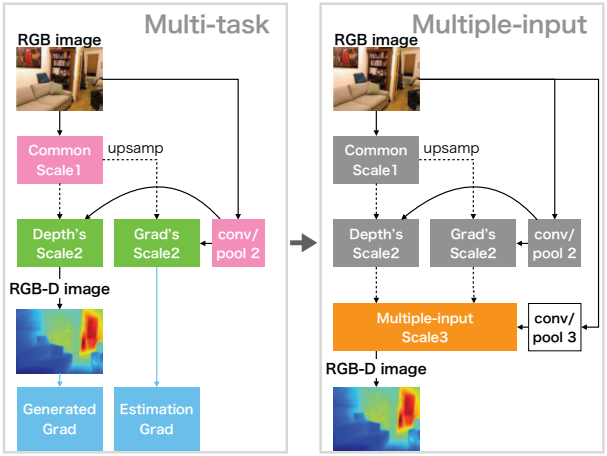

Fig. 6. Train process for our supervised model

We use the SfM Learner [9] as the origin for the unsupervised learning model. Similar to the case of supervised learning, our model branches the path of the feature extraction, and estimates the refined depth using the estimated depth and the depth gradient. In this case, the feature extractor is the common encoder(Fig. 5. Pink frame), and the network concatenates two middle outputs in the multiple-input decoder(Fig. 5. Orange frame). Refinement depth decoder's size is same that original decoder.

IV. Training Procedure

We will train the proposed model through two stages. In the first stage, the model gets common features by multi-task learning that is depth and depth gradient estimation. At the next stage, our model learns refinement depth estimation using multiple-input of estimation depths and depth gradients. Before the 2nd stage learning, we freeze the weights of feature extractor to save common features. In addition, the model uses the loss function (Eq. 1) for the multi-task learning. The constraint that the network has to consider the depth gradient keeps the depth consistent. Simultaneously, the estimation depth serves as a model for the depth gradient estimation. As a result, depth and depth gradient tasks learn from each other.

A. Training of supervised learning

In the first stage, we train our MSM's Scale 1 and 2, thereby the network learn multi-task that depth and depth gradient estimation(Fig. 6. Green frame). Simultaneously, we apply edge filters to the estimation depth and use the generated depth gradient to the loss:Eq. 1(Fig. 6. Blue frame). At this time, the common convolutional layer saves features considering depths and depth gradients(Fig. 6. Pink frame). When learning has converged, proceed to the next step. In the next stage, we train our MSM's Scale 3 using multiple-input. Before learning, we fix up the weights of middle layers to save the common features(Fig. 6. Gray frame). Additionally, we concatenate the estimation depths and estimation depth gradients. We set the concat to multiple-input of scale 3(Fig. 6. Orange frame), and estimate the refinement depth. The network uses the sc-inv error [1] and eq. 1 as the loss function for the first stage. Similarly, the network uses the sc-inv error [1] as the loss function by itself for the second stage.

978-1-7281-9732-6/20/$31.00 ©2020 IEEE

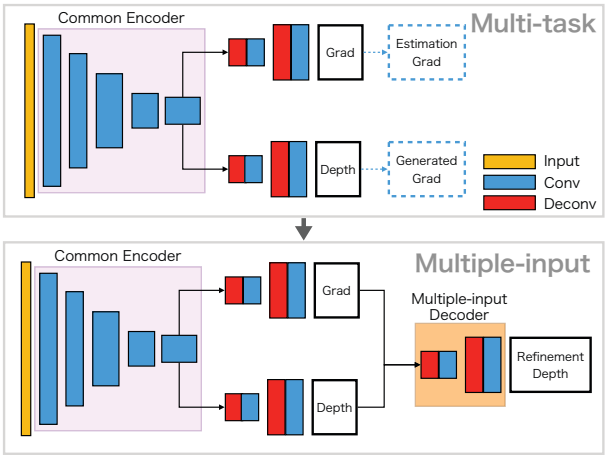

Fig. 7. SfM Learner

Supervised learning(Multi-task learning + Multiple-input)					
Normal evaluations			MAE in each section of every 1m		
	MSM	Our		MSM	Our
$\sigma < 1.25$	0.344	0.353	0m∼1m	1.156	1.224
$\sigma < 1.25^2$	0.626	0.648	1m∼2m	1.087	0.766
$\sigma < 1.25^3$	0.802	0.826	2m∼3m	1.020	0.346
abs. rel	0.648	0.412	3m∼4m	1.742	0.944
sqr. rel	10.511	0.730	4m∼5m	2.349	1.726
RMS(lin)	1.747	1.125	5m∼6m	2.601	2.191
RMS(log)	0.281	0.171	6m∼7m	3.456	2.787
sc-inv.	1.44	0.125	7m∼8m	3.427	3.00
MAE	1.75	1.125	8m∼9m	3.911	3.241
			9m∼10m	4.005	3.392

TABLE I
DEPTH ESTIMATION MEASUREMENTS. RIGHT TABLE IS EVALUATION FOR EACH METER WITH MAE. NOTE HIGHER IS BETTER FOR TOP ROWS OF THE LEFT TABLE, WHILE LOWER IS BETTER FOR THE BOTTOM.

Unsupervised learning(Multi-task learning)					
Normal evaluations			SfM's Losses		
	SfM	Our		SfM	Our
$\sigma < 1.25$	0.283	0.225			
$\sigma < 1.25^2$	0.521	0.439			
$\sigma < 1.25^3$	0.724	0.621			
abs. rel	0.475	0.627	recon. loss	0.152	0.140
sqr. rel	5.270	8.084	smooth. loss	0.043	0.021
RMS(lin)	12.554	13.603	mask loss	0.082	0.263
RMS(log)	0.617	0.734			

TABLE II
DEPTH ESTIMATION MEASUREMENTS.NOTE HIGHER IS BETTER FOR TOP ROWS, WHILE LOWER IS BETTER FOR THE BOTTOM.

B. Training of unsupervised learning

In training unsupervised learning, similar to the case of supervised learning, we get common features using multi-task learning. Furthermore, we train the refinement depth estimation using multiple-input. In this case, after training our Depth CNN's common decoder and normal decoder(Fig. 7. Top), train multiple-input decoder(Fig. 7. Bottom). The common decoder contains the features that considers depth and depth gradients(Fig. 7. Pink frame). The network uses the SfM Learner's loss [9] and eq. 1 as the loss function for the first stage. Similarly, the network uses the SfM Learner's loss [9] by itself for the second stage.

V. Performance Experiments

We trained the original model and the proposed model in the same environment; Training dataset, Augmentation set, Optimizer, Loss functions, etc.

A. Dataset & Evaluation

We used the NYU Depth v1 and v2 dataset's Labeled data [9] for supervised learning, and splitted the dataset's scenes; 70% train data, 20% validation data and 10% test data. Additionally, we used augmentation from Eigen et al [3], and train ground-truth depth map estimation. In contrast, we used KITTI dataset's raw data [10] for unsupervised learning. We trained the unsupervised model using only the data taken on September 26, 2011 to simplify the scale of the learning. The ratio of training data division is the same as in supervised learning. Especially in supervised learning , we evaluated each area of meters with MAE.

The following 10 functions are used for evaluation. The lower from the loss and error is the better, and the higher from accuracy is the better. The reconstruction loss(8), the smoothness loss(9), and the mask loss(10) is the SfM Learner's training functions. The reconstruction loss is the absolute error of the generated image and target image, and is used for the reconstruction task. The smoothness loss is a constraint term for flat points in the image; Second derivative of the depth between neighboring pixels. The mask loss is the size of the masking area for a moving object, and is calculated based on the PoseNet estimation.

1) thresholded accuracy: $\sigma < 1.25, \sigma < 1.25^2, \sigma < 1.25^3$
2) squared absolute error: abs. rel
3) squared relative error: sqr. rel
4) root mean squared error: RMS(lin)
5) root mean squared error log_{10}: RMS(log)
6) mean absolute error: MAE
7) sc-inv error: sc-inv. [3]
8) reconstruction loss: recon. loss [9]
9) smoothness loss: smooth. loss [9]
10) mask loss [9]

B. Result of supervised learning

We set train batch size to 8, and set optimizer to Adam with $\beta_1 = 0.9$, $\beta_2 = 0.999$, learning rate of 0.0001. We applied fine-tuning to Scale1; the VGG16 model with the imagenet weights.We computed the generated depth gradient using a simple matrix operation($[-1, 0, 1], [-1, 0, 1]^T$)(Fig. 6. Blue frame).

The accuracy of our depth estimates is compared with the origin method in Table 1. Our proposed model improved the accuracy of MSM in the normal evaluation function(Table 1. Left). From these results especially the MAE evaluation, we confirmed that our model improved accuracy by 50cm. Additionally, the proposed model improved the accuracy in all

978-1-7281-9732-6/20/$31.00 ©2020 IEEE

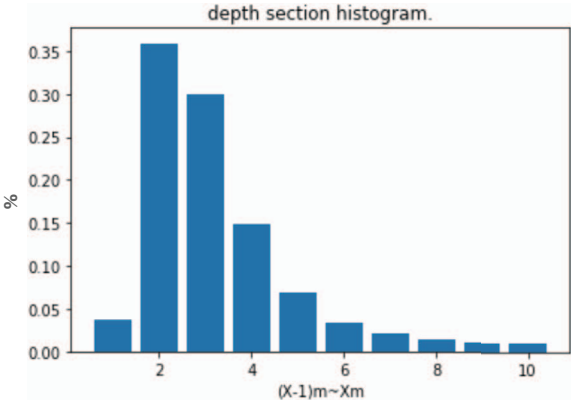

Fig. 8. Number of pixels that belong to the same section of depth. We normalized the graph that the sum of all frequencies is 1.

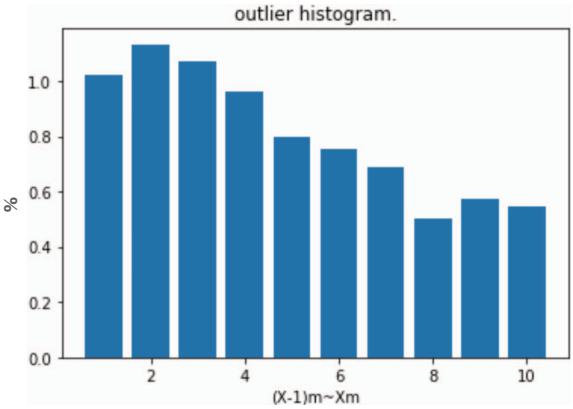

Fig. 9. Ratio of outliers to the number of pixels in same section of depth. We normalized the number of pixels in same section to 1.

sections except 0m to 1m(Table 1. Right). The results showed that the proposed model realized versatile depth estimation. On the other hand, the adverse effect of selecting depth gradient as additional information appeared in the estimation of the very close distance. Fig. 8. shows that the number of pixels in the section from 0m to 1m is significantly smaller than in other short distance sections. Moreover, Fig. 9. shows that the 0 m to 1 m section has more outliers than long distance sections, where there are about the same number of pixels.

C. Result of unsupervised learning

In the unsupervised learning, we trained only multi-task learning as a preliminary experiment of the integrated model(Fig. 7. Top). We set train batch size to 4, and used optimizer to Adam with $\beta_1 = 0.9$, $\beta_2 = 0.999$, learning rate of 0.0002. In addition, we set the parameters of the reconstruction loss, the smoothness loss and the mask loss to 1,0.1 and 0.2. We used a filter similar to supervised learning for loss function($[-1, 0, 1], [-1, 0, 1]^T$).

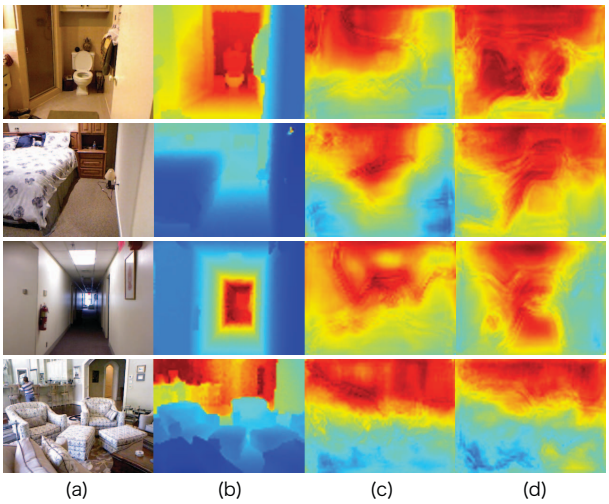

Fig. 10. Example depth results of supervised model. (a) RGB input; (b) ground truth; (c) our result; (d) result of the origin MSM. Note the color range of each image is individually scaled.

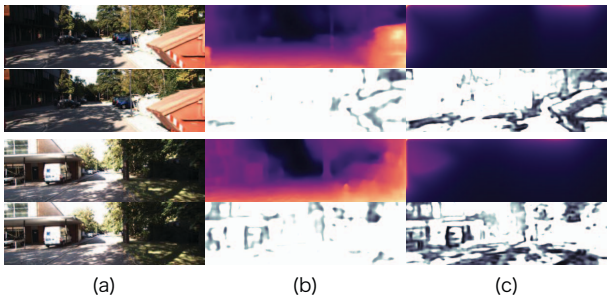

Fig. 11. Example results of unsupervised model. The first and third lines are depths, the second and fourth lines are mask of the mask loss. (a) RGB input; (b) result of the origin SfM Learner; (c) our result. Note the color range of each image is individually scaled.

The accuracy of our depth estimates is compared with the origin method in Table 2. Our proposed model didn't improve the accuracy of the SfM Learner in the normal evaluation function(Table 2. Left). Altogether, in depth estimation there was no part that was superior to the original. Nevertheless, the reconstruction loss and the smoothness loss have improved accuracy. The reconstruction loss is calculated on the assumption of the estimated depth. Therefore, the result is counterintuitive. However, note that the reconstruction loss depends on the estimated camera parameters as well as the estimated depth. In other words, it is highly possible that multi-task learning improved the camera parameters estimation and the model reduced depth dependency. In conclusion, depth-gradient multi-task learning improved 3-D reconstruction for the unsupervised learning.

VI. Discussions and Conclusion

We have proposed an integrated network architecture using depth gradient. Our architecture connected the output from the Multi-task learning to the Multiple-input model. Thereby, we

978-1-7281-9732-6/20/$31.00 ©2020 IEEE

strived to acquire general-purpose features, reduce the model size, and improve the accuracy. In addition, our architecture selects depth gradients as additional information, and can be applied to both the supervised and unsupervised learning. If the outliers are large in spite of the low frequency of occurrence of depth, there was an adverse effect. However, the result of supervised learning confirmed that it improves accuracy in about all section of depth. Contrary to this, preliminary experiments with unsupervised learning did not improve depth estimation. Nevertheless, unsupervised multi-task learning had the function of improving the accuracy of reconstruction tasks. There are a few possible reasons for we couldn't improve depth estimation in unsupervised. First, there is a possibility that competition between the smoothness loss and depth gradient loss adversely affected training. Next, there is a possibility that the estimation accuracy can be improved only after completing the training using multiple-input. Outlook for the future of research, we want to experiment with multiple-input of unsupervised learning and confirm the final accuracy (Fig. 11). Additionally, we want to train the supervised learning using a raw dataset of Nyu depth and compare result with our depth maps (Fig. 10).

REFERENCES

[1] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[2] Chaoqiang Zhao. Monocular Depth Estimation Based On Deep Learning: An Overview. 2020.

[3] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*, volume 3, pages 2366–2374. Neural information processing systems foundation, 2014.

[4] Jun Li, Reinhard Klein, and Angela Yao. A Two-Streamed Network for Estimating Fine-Scaled Depth Maps from Single RGB Images. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[5] Omid Hosseini Jafari, Oliver Groth, Alexander Kirillov, Michael Ying Yang, and Carsten Rother. Analyzing modular CNN architectures for joint depth prediction and semantic segmentation. *Proceedings - IEEE International Conference on Robotics and Automation*, (iv):4620–4627, 2017.

[6] Xiaojuan Qi, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia. GeoNet: Geometric Neural Network for Joint Depth and Surface Normal Estimation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018.

[7] Lubor Ladický, Jianbo Shi, and Marc Pollefeys. Pulling things out of perspective. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014.

[8] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan Yuille. Towards unified depth and semantic prediction from a single image. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015.

[9] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017.

[10] Uhrig Jonas, Schneider Nick, Schneider Lukas, Franke Uwe, Brox Thomas, and Geiger Andreas. Sparsity Invariant CNNs. *International Conference on 3D Vision (3DV)*, 2017.

978-1-7281-9732-6/20/$31.00 ©2020 IEEE