INTERNATIONAL JOURNAL OF CIRCUITS, SYSTEMS AND SIGNAL PROCESSING
DOI: 10.46300/9106.2021.15.59 Volume 15, 2021

Depth Estimation for Monocular Image Based on Convolutional Neural Networks

Binglin Niu[1], Mengxia Tang[2], Xuelin Chen*[3]
[1]School of information engineering, Xinyang Agriculture and Forestry University
No.1 North Circular Road, Pingqiao District, xingyang, 464000.
[2]School of technology, Beijing Forestry University
No.35 Tsinghua East Road, Haidian District, Beijing, 10086.
[3]School of finance and economics, Xinyang Agriculture and Forestry University
No.1 North Circular Road, Pingqiao District, xingyang, 464000.
China

Received: November 23, 2020. Revised: May 12, 2021. Accepted: June 2, 2021. Published: June 7, 2021.

***Abstract*— Perceiving the three-dimensional structure of the surrounding environment and analyzing it for autonomous movement is an indispensable element for robots to operate in scenes. Recovering depth information and the three-dimensional spatial structure from monocular images is a basic mission of computer vision. For the objects in the image, there are many scenes that may produce it. This paper proposes to use a supervised end-to-end network to perform depth estimation without relying on any subsequent processing operations, such as probabilistic graphic models and other extra fine steps. This paper uses an encoder-decoder structure with feature pyramid to complete the prediction of dense depth maps. The encoder adopts ResNeXt-50 network to achieve main features from the original image. The feature pyramid structure can merge high and low level information with each other, and the feature information is not lost. The decoder utilizes the transposed convolutional and the convolutional layer to connect as an up-sampling structure to expand the resolution of the output. The structure adopted in this paper is applied to the indoor dataset NYU Depth v2 to obtain better prediction results than other methods. The experimental results show that on the NYU Depth v2 dataset, our method achieves the best results on 5 indicators and the sub-optimal results on 1 indicator.**

***Keywords*—Three-dimensional structure, single image, convolutional neural networks, depth map.**

I. Introduction

Depth estimation provides help for understanding the three-dimensional relationship between objects in the image scene, thereby improving the accuracy of current recognition tasks [1] and can be applied to indoor scenes understanding, pose estimation, 3D reconstruction, robot navigation and virtual reality (VR) [2-9].

The main purpose of predicting the depth is to assign a one-to-one corresponding depth value to all pixels in a single image. Most of the methods for predicting depth are based on stereo vision [10,11] or motion, while there are fewer depth map prediction methods based on monocular vision. The reason is that the prediction method based on stereo vision can accurately restore depth information while providing accurate image correspondence and it can be achieved as long as the correspondence between image points is found through local appearance features. Monocular vision requires a global view of the scene to associate depth cues, and as far as a single image is concerned, there may be countless scenes that can produce it. As far as a single image is concerned, it mainly uses visual motion information [12], different shooting conditions [13,14], linear perspective and occlusion to complete the depth prediction. Since then, researchers have begun to study methods for estimating depth maps from monocular images.

The traditional methods [15-19] mainly use the combination of hand-craft features and probabilistic graphic models to solve the problem of prediction depth map for monocular images. The traditional methods mainly establish geometric assumptions to predict the spatial contours of indoor or outdoor scenes. However, the traditional method of predicting depth maps has many problems: probabilistic graphic models are difficult to train, and approximate methods are often used for calculation; hand-craft features are not as accurate as extracted by convolutional neural networks; some assumptions need to be established to complete prediction and usually only applicable to specific scenarios.

Recently, deep learning has demonstrated powerful feature representation capabilities in computer vision tasks. Deep learning methods are introduced into the task of depth estimation [20-25]. The convolutional neural network is fast in feature extraction, and the prediction results are more accurate. The supervised method is mainly divided into the use of improved convolutional neural network to extract features [25], the use of multi-scale feature fusion [21,24], and the use of probabilistic graphic models for subsequent processing [22,23].

The supervised method requires numerous ground-truth depth map to train the model, so that it has strong generalization ability, the real depth map is usually difficult to obtain, and the supervised depth map estimation method is considered to be an ill-posed problem because of the

ambiguity of the scale. Researchers have proposed many semi-supervised and unsupervised methods [26,27] to solve the problems of supervised prediction methods. However, the estimation output of semi-supervised and unsupervised methods are not as accurate as those of supervised methods, and internal settings such as camera calibration are required.

The paper proposes a novel single-image supervised depth prediction method. We use the encoder-decoder structure with feature pyramid to complete prediction of dense depth map. The encoder uses ResNeXt-50[28] as the basic network to obtain representative features from the raw image. The pyramid structure can make high-level and low-level information merge with each other, and feature information is not lost. The decoder uses the deconvolution and the convolution layer to connect as an up-sampling block to restore the size of the output feature. At the same time, we experiment with the proposed method on indoor datasets. The visual effects and qualitative indicators are all due to other methods.

II. DEPTH ESTIMATION METHODS

In this part, we introduce our depth map estimation method for a single image in detail. Above all, we use ResNeXt-50 as our encoder part, and the decoder section adopts a new up-sampling strategy to expand the size of the predicted output. The encoder and decoder are combined and arranged in the feature pyramid. Second, we propose a suitable training strategy to optimize a given task and achieve better prediction results.

A. *Network framework*

Inspired by U-net network structure [30], we designed a depth estimation network with an encoding-decoding structure, and regarded depth estimation as learning problem of deep regression. Fig. 1 shows the overall network structure of the proposed model.

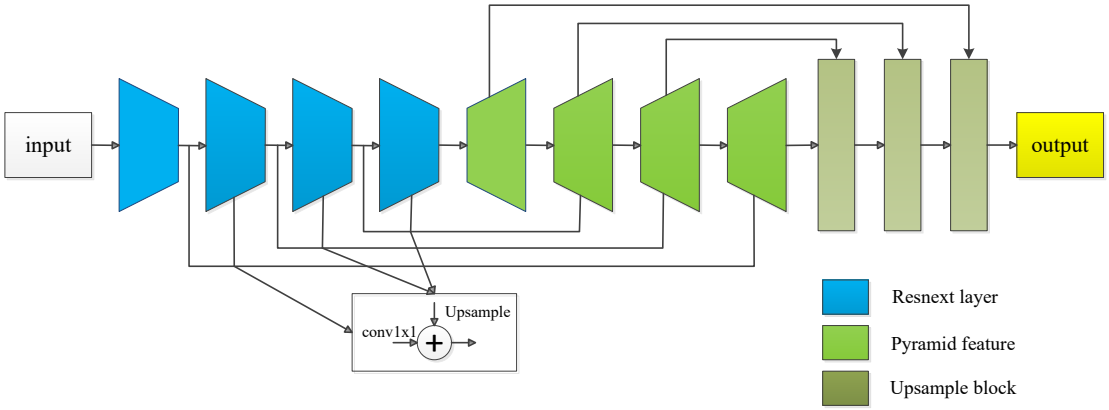


Fig.1 Our proposed encoder-decoder structure

The skeleton structure of the encoder is ResNeXt-50 (the blue block in the figure), and the decoder structure is composed of three deconvolution blocks (the green block in the figure). The encoders and decoders are arranged in the form of feature pyramid. The encoder part extracts the key features from the original data and the decoder section mainly adopts deconvolution operations to restore the resolution of the output feature map.

In the encoder module, we use ResNeXt-50 [28] to acquire multi-scale features of the original image, similar to [29]. The feature image size is reduced at the speed of ratio 2, and the feature image depth is zoomed in at the speed of ratio 2. The top-level feature image is the feature image with low resolution and strong semantic information. In the network structure, many blocks constitute a stage, and the final output of each stage is regarded as the first level of the multi-scale structure. For ResNeXt-50 as the basic skeleton, the biggest difference between the ResNeXt block and the residual block is that the intermediate dimensions are divided into multiple groups at the same time, and the dimensions of each group are very small, which reduces the running time. The characteristics of the ResNeXt block are shown in Fig. 2.The residual network mainly has four layers, which are composed of 3, 4, 6, and 3 residual blocks respectively. Each residual block consists of 1×1, 3×3, 1×1 convolutional layers and skip connections.

This structure divides the middle dimension into multiple groups and reduces the dimension of each group. As shown in the figure 2, the 128 dimensions are divided into 32 groups, and each group has a dimension of 4. Among them, different convolution kernels (1×1,3×3,1×1) have different functions, namely compression dimension, convolution processing, and restoration dimension. The network parameters are reduced and the speed is increased.

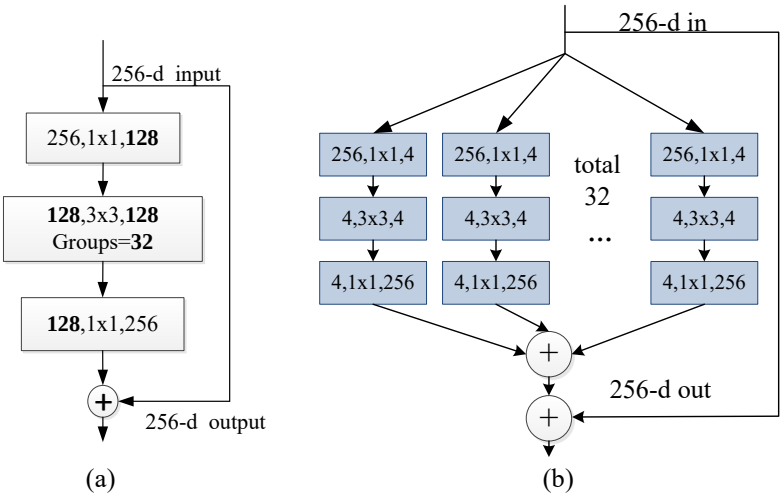


Fig.2 The basic residual block structure of ResNetXt-50. (a) is residual block; (b) is resnext block. 1x1 and 3x3represent that the convolution kernel is 1x1 and 3x3 size

The resolution from small to large path and internal connections: The higher levels are connected to the lower levels through horizontal connections and simple up-sampling operations to strengthen the reuse of features. The up-sampling operation expands the feature size of the high-level to be consistent with the feature of the low-level. We use bilinear interpolation method to up-sample the lower resolution features map by 2 times, and then the up-sampled feature map and the corresponding path of resolution from large to small feature maps (after 1×1 convolution for decreasing the number of channels and parameters) element-wise addition and merging, repeat this process until the highest resolution. Before starting the above operation, we can generate the roughest resolution map by convolving c5 with 1×1.

In the encoding-decoding structure, the decoder only performs up-sampling operations on the highest-level features extracted by the encoder, and the contextual semantic information in the bottom layer will be lost, and the highest-level features contain less information about small objects in the image, so the up-sampling operation is performed When restoring the feature resolution, the depth of small objects will be blurred or even ignored. The middle connection path of the feature pyramid merges multilevel feature The high resolution features reduce the number of parameters through convolution, and the high-level features increase the resolution to the same as the low-level through bilinear interpolation and up-sampling, and then the processed features are gradually processed. The intermediate connection path helps to retain detailed information from low-level features.

In the decoder stage, we designed a novel deconvolution block to increase the size of output feature map. Through the structure, the output feature map is up-sampling to a higher resolution, and the channel dimension is consistent with the feature map with higher resolution. Our decoder section contains four deconvolution block structures. The specific details of the deconvolution block structure are shown in Fig. 3.

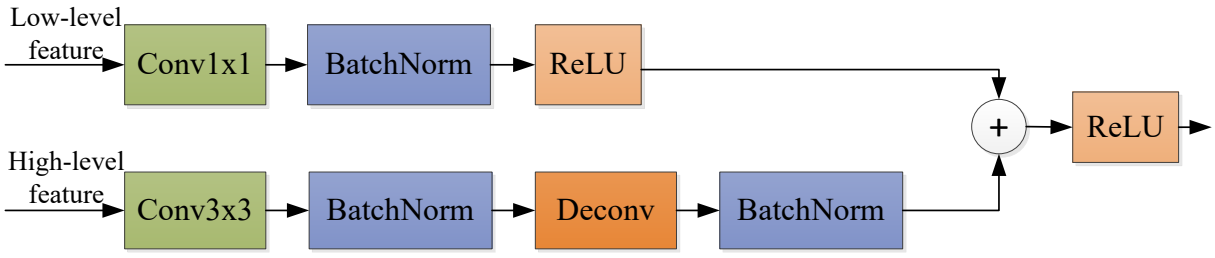


Fig. 3 The structure of deconvolution block. The size of the output of deconvolution block is the same as that of the low-level feature maps, and the dimension is the same.

First, we use 3×3 convolutional, regularization and deconvolution layers to keep the output high-level and low-level features the same size. The convolutional layer plays a role in reducing the number of channels of the input feature map, and deconvolution operation keeps the dimensionality unchanged. On the other hand, the low-level features are passed through the 1×1 convolutional layer, BN layer and activation layer (ReLU), and then the output of the low-level and the high-level features are added. The number of channels remains the same. Unlike cascading, since the number of channels of the feature map is not increased, the number of

parameters can be significantly reduced. Finally, we output the feature map through the ReLU layer.

III. LOSS FUNCTION

When training the regression model, it is necessary to design a suitable loss function to improve the accuracy and robustness of the prediction, and further explore the relationship between the predicted and ground-truth value. *Manhattan* distance (L_1) and *Euclidean* distance (L_2) are often used as the standard loss function for regression tasks. The *Manhattan* distance is the sum of the absolute values of the difference between the estimated and true depth point. The *Euclidean distance* loss minimizes the squared *Euclidean* term between the predicted and ground-truth value. Expressed by the following:

$$L_1 = \sum_{i=1}^{n} \left| y_i - y_i^* \right| \tag{1}$$

$$L_2 = \sqrt{\sum_{i=1}^{n} \left(y_i - y_i^* \right)^2} \tag{2}$$

where n is sum of valid pixels, i is the index. y_i and y_i^* is the i pixel value of the estimated and true depth map, separately.

From Eq.1 and Eq.2, we can get the depth loss L_d,

$$L_d = \begin{cases} | y_i - y_i^* | & | y_i - y_i^* | \le c, \\ [(y_i - y_i^*)^2 + c^2] / 2c & | y_i - y_i^* | > c. \end{cases} \tag{3}$$

where $c = \frac{1}{5} \max_i \left(\left| y_i - y_i^* \right| \right)$.

As shown in formula (3), when the absolute value of the error of the estimated and true depth value is greater than c, the L_d is changed to L_2 loss; otherwise, we take the L_1 loss function as the L_d. The L_2 term is more sensitive to pixels with high residual errors, which increases its weight; and the L_1 term has a greater impact on the pixel values with smaller errors. However, for edge structures with sudden changes in depth, the above loss is relatively insensitive to the offset in the horizontal and vertical directions and edge deformation and blurring. Therefore, we added gradient term to make the edge of the prediction result clearer, see Eq.4.

$$L_g = \frac{1}{n} \sum_{i=1}^{n} \left| \nabla y_i^* - \nabla y_i \right| \tag{4}$$

where ∇y_i^* and ∇y_i are the sum of the X-direction and Y-direction gradients for the real and output depth map, respectively. The gradient term is sensitive to the offset of the edges in the horizontal and vertical directions.

So as to deal with the depth of smaller objects and make further efforts the accuracy of the global depth map, we introduce the surface normal loss, (the measurement of the depth map relative to the ground-truth depth map and the accuracy of the estimated surface normal) into our training strategy.

$$L_n = \frac{1}{n} \sum_{i=1}^{n} \left(1 - \frac{\left\langle n_i^d, n_i^p \right\rangle}{\left\| n_i^d \right\|_2 \left\| n_i^p \right\|_2} \right) \tag{5}$$

where $n_i^d = \left[-\nabla_x(d_i), -\nabla_y(d_i), 1 \right]^T$ is the surface normal vectors and $n_i^p = \left[-\nabla_x(p_i), -\nabla_y(p_i), 1 \right]^T$ is the estimated depth map. $\nabla_x(\ )$ and $\nabla_y(\ )$ means X and Y direction gradients, $\langle \cdot, \cdot \rangle$ expresses the inner product of the two vectors. $\frac{\left\langle n_i^d, n_i^p \right\rangle}{\left\| n_i^d \right\|_2 \left\| n_i^p \right\|_2}$ indicates the cosine similarity of the surface normal vector between the estimated and true depth map. The closer the value is to 1, the higher the similarity.

Finally, we assign different weights to the L_d, L_g and L_n. The weights are obtained by experiments and are 0.6, 0.2 and 0.2 respectively. The loss function we adopt is,

$$Loss = 0.6 * L_d + 0.2 * L_g + 0.2 * L_n \tag{6}$$

IV. EXPERIMENTATION

This paper conducts experiments on the indoor dataset−NYU Depth v2, which is usually used to evaluate the performance of depth estimation models, to verify the effectiveness of proposed method. The detailed experimental process and results will be described in the following summary.

A. Dataset: NYU Depth v2

We use the NYU Depth v2 dataset, which is commonly used in depth prediction, to train and test our model. The raw dataset was acquired by the Microsoft Kinect camera and contained 464 scenes. We use the official split, with 249/215 training and test scenes. We expand the original rough dataset of NYU Depth v2 as the training set of the model. There are about 40,000 training images in the original rough dataset. This paper uses data enhancement operations such as rotation, flips, scale, to expand the training images to about 1.2 million as the training set. The test set is the official 164 test images.

The size of the raw image are 640×480, we downsample the raw image to half, and then center clipped to 300×228 as the input to network. Finally, we train our model with a batch size of 8 for approximately 20 hours. The initial learning rate is 0.0001. The momentum is 0.9.

The visual effect of the experiment is represented in Fig.4 below.

(a) (b) (c)

Fig.4 The visual effect of our method on the test set. (a) is the original input, (b) is the ground-truth depth map, and (c) is the output predicted by this method.
The darker the color represents the smaller the depth value and the closer the object is to the lens; the lighter the color (the yellow area in the figure) represents the greater the depth value and the farther the object is from the lens.

B. Data processing and experimental details

The depth map output by our model has a lower resolution than the ground truth depth map. When comparing, we use bilinear interpolation to expand the resolution of the prediction result (114×150) to be consistent with the ground truth (480×640). The evaluation criteria used in this paper are the same as classic papers, and the evaluation criteria used are as follows:

Threshld: percentage of predicted pixels.

$$\max\left(\frac{y_i^*}{y_i}, \frac{y_i}{y_i^*}\right) = \delta < threshold$$

Mean relative error (REL): $\frac{1}{n}\sum_{i=1}^{n}\frac{\left|y_i^* - y_i\right|}{y_i^*}$

Mean Log_{10} error (Log_{10}): $\frac{1}{n}\sum_{i=1}^{n}\left|\log_{10} y_i^* - \log_{10} y_i\right|$

Root mean squared error (RMSE): $\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i^* - y_i\right)^2}$

where n is the total number of effective pixels, i represents the index. y_i and y_i^* is the i pixel value of the predicted and ground truth depth map, separately. The *threshold* is a constant: 1.25, 1.25^2, 1.25^3.

In order to verity the capability of the model on quantitative indicators, we conducted a comparative

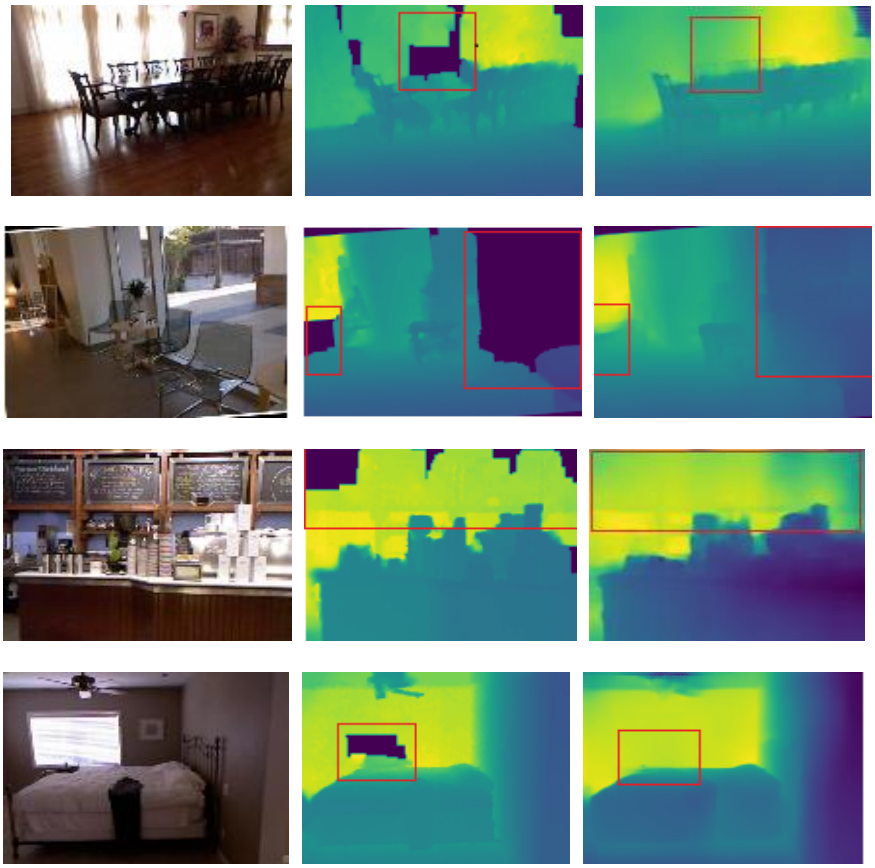
experiment on the test set of the indoor dataset to prove that our model is better than other methods. Figure 1 shows the quantitative indicators of our method compared with other methods.

Table 1. The experimental results of the depth estimation on NYU Depth v2 dataset.

Method	Higher is better			Lower is better		
	$\delta<1.25$	$\delta<1.25^2$	$\delta<1.25^3$	REL	Log_{10}	RMSE
Saxena et al.[16]	0.447	0.745	0.897	0.349	--	1.214
Ladicky et al.[17]	0.542	0.829	0.940	--	--	--
Karsch et al.[18]	--	--	--	0.350	--	1.2
Liu et al.[22]	0.614	0.883	0.971	0.230	0.095	0.824
Eigen et al.[20]	0.611	0.887	0.971	0.215	--	0.907
Eigen et al.[21]	0.769	0.950	0.988	0.158	--	0.641
Laina et al. (L2)[25]	0.785	0.952	0.987	0.138	0.060	0.592
Ours	0.796	0.962	0.990	0.134	0.058	0.583

C. Depth Completion

Our method can also fill in missing depth values. The depth map obtained by depth sensors such as Microsoft Kinect has the problem of loss of depth information. For instance, if the surface of an object is too smooth, bright or high reflectivity, 10% to 50% of the depth information will be lost. The method only needs to input the RGB image to perform the depth completion on the corresponding depth map in the NYU Depth v2 datasets. The result of completing the depth map is shown in Fig.5



(a) (b) (c)

Fig.5 Deep completion cases. (a) is the raw input, (b) is the ground-truth depth map, and (c) is the output complemented by proposed model. The figure shows that this method can complement the depth of objects with high reflectivity in the ground-truth depth map, such as windows, glass, and objects directly illuminated by lights.

V. CONCLUSION

Obtaining depth information from monocular images is the meaning mission. The paper put forward the novel network structure to estimate the depth information. There are two main innovations in this paper, as follows. First of all, the article adopts an auto-encoder network based on pyramid structure to enhance the flow of feature information. Second, the loss function is composed of depth, gradient and surface normal loss with different weight values. The gradient loss makes the edges of the prediction results clearer, and the gradient changes in the edge area of the object are more obvious. The surface normal loss is used to refine the details of the object. Besides, the novel up-sampling structure is proposed to expand the size of the output. Experiments conducted on the indoor dataset NYU Depth v2 verify that the method proposed is superior to other methods. In the future, we will merge the output predicted by proposed model with the original RGB image to complete the semantic segmentation of complex scenes.

References

[1] X. Ren, L. Bo, and D. Fox, "RGB-D scene labeling: Features and algorithms," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (IEEE, 2012), pp. 2759–2766.

[2] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, "Real-time human pose recognition in parts from single depth images," Commun. ACM. 56(1), 116–124 (2013).

[3] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in European Conference on Computer Vision (ECCV, 2012), pp. 746–760.

[4] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon, "The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation." in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (IEEE, 2012), pp. 103–110.

[5] Chen, Y, Yang, D, Liao, W. Efficient multi-view 3D video multicast with depth image-based rendering in LTE networks. In Proceedings of the IEEE Global Communications Conference (GLOBECOM), Atlanta, GA, USA, 9–13 December 2013; pp. 4427–4433.

[6] Cao, Y.; Xu, B.; Ye, Z.; Yang, J.; Cao, Y.; Tisse, C.; Li, X. Depth and thermal sensor fusion to enhance 3D thermographic reconstruction. Opt. Express 2018, 26, 8179–8193.

[7] Ragaglia, M.; Zanchettin, A.M.; Rocco, P. Trajectory generation algorithm for safe human-robot collaboration based on multiple depth sensor measurements. Mechatronics 2018, 55, 267–281.

[8] Wang, S.; Zuo, X.; Wang, R.; Cheng, F.; Yang, R. A generative human-robot motion retargeting approach using a single depth sensor. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Marina Bay Sands, Singapore, 29 May–3 June 2017; pp. 5369–5376.

[9] W. Lee, N. Park, and W. Woo. Depth-assisted real-time 3d object detection for augmented reality. ICAT, 2:126–132, 2011.

[10] R. Memisevic and C. Conrad. Stereopsis via deep learning. In NIPS Workshop on Deep Learning, volume 1, 2011.

[11] F. H. Sinz, J. Q. Candela, G. H. Bakır, C. E. Rasmussen, and M. O. Franz. Learning depth from stereo. In Pattern Recognition, pages 245–252. Springer, 2004.

[12] R. Szeliski. Structure from motion. In Computer Vision, Texts in Computer Science, pages 303–334. Springer London, 2011.

[13] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah. Shape-from-shading: a survey. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 21(8):690–706, 1999.

[14] S. Suwajanakorn and C. Hernandez. Depth from focus with your mobile phone. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[15] D. Hoiem, A. A. Efros, and M. Hebert. Automatic photo pop-up. In ACM SIGGRAPH, pages 577–584, 2005.

[16] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In NIPS, 2005.

[17] L. Ladicky, J. Shi, and M. Pollefeys. Pulling things out of perspective. In CVPR, 2014.

[18] K. Karsch, C. Liu, S. B. Kang, and N. England. Depth extraction from video using non-parametric sampling. In TPAMI, 2014.

[19] J. Konrad, M. Wang, and P. Ishwar, "2d-to-3d image conversion by learning depth from examples," Computer Vision and Pattern Recognition Workshops (CVPR, 2012), pp. 16-22.

[20] D. Eigen, C. Puhrsch, and R. Fergus. Prediction from a single image using a multi-scale deep network. In Proc. Conf. Neural Information Processing Systems (NIPS), 2014.

[21] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In Proc. Int. Conf. Computer Vision (ICCV), 2015.

[22] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In Proc. Conf. Computer Vision and Pattern Recognition (CVPR), pages 5162–5170, 2015.

[23] D. Xu, E. Ricci, W. Ouyang, X. Wang, and N. Sebe, "Multi-Scale continuous crfs as sequential deep networks for monocular depth estimation," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (IEEE, 2017), pp. 5354–5362.

[24] J. Li, R. Klein, and A. Yao, "A Two-Streamed Network for Estimating Fine-Scaled Depth Maps from Single RGB Images," IEEE International Conference on Computer Vision (ICCV, 2017), pp. 3392-3400.

[25] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper Depth Prediction with Fully Convolutional Residual Networks," Fourth International Conference on 3D Vision (3DV, 2016), pp. 239-248.
[26] Y. Kuznietsov, J. Stückler, and B. Leibe, Semi-Supervised Deep Learning for Monocular Depth Map Prediction, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR, 2017), pp. 6647-6655.
[27] A. Pilzer, D. Xu, M. M. Puscas, et al. Unsupervised Adversarial Depth Estimation using Cycled Generative Networks. International Conference on 3D Vision(3DV),2018,pp.587-595
[28] S. Xie, R. Girshick, P. Dollar, Z. Tu, K. He. Aggregated Residual Transformations for Deep Neural Networks. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp.5987-5995.
[29] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-assisted Intervention, MICCAI, Munich, Gemany, 5-9, 2015, pp. 234-241