Research Article
An Extremely Effective Spatial Pyramid and Pixel Shuffle Upsampling Decoder for Multiscale Monocular Depth Estimation

Huilan Luo, Yuan Chen, and Yifeng Zhou
School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou 341000, China
Correspondence should be addressed to Huilan Luo; luohuilan@jxust.edu.cn
Received 13 June 2022; Accepted 12 July 2022; Published 1 August 2022

Academic Editor: Dalin Zhang
Copyright © 2022 Huilan Luo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To estimate the accurate depth from a single image, we proposed a novel and effective depth estimation architecture to solve the problem of missing and blurred contours of small objects in the depth map. The architecture consists of Extremely Effective Spatial Pyramid modules (EESP) and Pixel Shuffle upsampling Decoders (PSD). The results of this study show that multilevel information and the upsampling method in the decoders are essential for recovering the accurate depth map. Through the model we proposed, competitive performance compared with state-of-the-art methods in terms of reconstruction of object boundaries and the detection rate of small objects has been demonstrated. Our approach has wide applications in higher-level visual tasks, including 3D reconstruction and autonomous driving.

1. Introduction

Monocular depth estimation is a long-standing task, which aims to predict the continuous depth value of each pixel from a single RGB image. This task has a wide range of application in various fields, such as scene understanding [1], scene segmentation [2], 3D reconstruction [3], and simultaneous localization and mapping (SLAM) [4]. Traditional depth estimation methods of image-based depth estimation are usually based on a binocular camera, which calculates the disparity of two 2D images (taken by a binocular camera) through stereo matching and triangulation to obtain a depth map. However, the binocular depth estimation method requires at least two fixed cameras, and it is difficult to capture enough features in the image to match when the scene has less or no texture. Therefore, researchers turn their attention to monocular depth estimation. Monocular depth estimation uses only one camera to obtain an image, which does not require additional complicated equipment and professional techniques. Thus, there has been an increasing demand for monocular depth estimation in recent years.

With the great success of deep learning, researchers have studied a number of various monocular depth estimation methods based on convolutional neural networks (CNN). Eigen et al. [5] first proposed a multiscale deep network to regress dense depth maps from coarse to fine. Laina et al. [6] proposed a fully convolutional residual network, which explores a new upsampling method to obtain more accurate depth predictions. Liu et al. [7] proposed a deep convolution neural field, which combines the depth convolution neural network with the continuous conditional random field to extract the structural information of features. In recent years, there have been new advances in monocular depth estimation. Wofk et al. [8] proposed an efficient and lightweight encoder-decoder network architecture and applied network pruning to further reduce computational complexity and latency for real-time monocular depth estimation. In literature [9], an attention mechanism and a multiscale feature fusion dense pyramid were used to further improve estimated depth maps of distant small-scale objects. In literature [10], an adversarial loss was introduced into the training stage of self-supervised depth estimation to optimize the depth network with high-level information.

Although monocular depth estimation research has made great progress, there are still some problems such as missing small objects and distortions in object shapes. Some works address the above issues by introducing auxiliary...
information such as geometric constraints and semantic information [11]. Others use feature fusion methods to improve depth estimation of small objects [9]. The CNN-based methods use a convolutional neural network to extract image features and then regress the features into image depth values, so the quality of extracted features directly affects the effect of monocular depth estimation. Inspired by this perception, we propose a novel U-shape network based on EESP skip connection modules and an upsampling method based on PSD modules. These modules can result in fewer parameters, clearer object contours, less object distortion, and fewer small objects missing. The main contributions can be summarized as follows:

(i) We design the Extremely Effective Spatial Pyramid (EESP) connection module as the skip connection of the U-shaped network. The EESP connection module transfers the multiscale features of different layers of the backbone network into the PSD module to supplement multilevel comprehensive information so as to obtain a better depth estimation effect. It is extremely effective in terms of performance improvement.

(ii) We introduce the Pixel Shuffle upsampling Decoders (PSD) module for the decoder to improve the resolution of feature maps and fully fuse feature information of different receptive fields. Compared with other upsampling methods, our method does not introduce extra parameters and can better alleviate the edge blur and artifacts caused by information loss.

2. Related Works

2.1. U-Shaped Networks. Since the FCRN [6] network was proposed, the encoder-decoder structure has been adopted in most monocular depth estimation methods. The encoder downsamples the input image several times to extract the global features, and the decoder upsamples the feature map to get the depth map. However, only relying on the final high-level features to estimate the depth of each pixel, the performance is not desirable because it lacks local detail information. Therefore, more recently, the U-shaped network, for example, U-Net, is commonly used, in which the decoder utilizes features from all layers by adding shortcut connections between the corresponding decoder layers and the encoder layers. In work [8], the multilevel features extracted from the encoder are directly added to the decoder. Their experimental results demonstrate that the enhancement of local features improves the accuracy of local depth in the prediction maps. Different from them, in this paper, the spatial pyramid EESP module is added to the skip connections of the U-Net network structure to fuse different scale-level features from different encoder layers.

2.2. Upsampling Methods. Depth estimation models usually use a backbone network such as ResNet [12] to extract the features of the input image, and the resolution of the output features is usually small, which will limit the resolution of the depth prediction map. Therefore, it is necessary to use encoder-decoder structure with upsampling operations at the decoder part to improve the resolution of the features. Commonly used upsampling methods are interpolation, deconvolution, and pixel shuffle. Bilinear interpolation is the most commonly used interpolation method [13, 14], but it will dilute the feature information, thus blurring the image edges, and ultimately affect the effect of depth estimation. Deconvolution is also one of the popular upsampling methods. Its advantage is that it can improve the upsampling effect through training, but multiple deconvolutions may produce artifacts [15]. Therefore, some works [6, 8] add convolution layers after deconvolution to alleviate the artifacts caused by continuous deconvolution to a certain extent.

Pixel shuffle, also known as subpixel convolution, is widely used in the field of image super-resolution processing [16, 17]. This method improves the resolution of the feature maps by reducing the number of channels of features, which also results in fewer parameters for subsequent convolution operations. Pixel shuffle retains all the feature information, which can better alleviate the edge blur and artifacts caused by information loss. Due to the advantages of pixel shuffle, some researchers have applied it to other computer vision areas, such as image reconstruction [18] and semantic segmentation [19]. In order to improve the resolution of output feature maps and further learn feature fusion, this paper uses pixel shuffle operation in the decoder. By rearranging the input features, pixel shuffle realizes the feature fusion between different channels, so it can also play the role of feature fusion.

2.3. Depthwise Separable Convolution. The depthwise separable convolution [20] consists of two parts: a depthwise convolution and a pointwise convolution. The depthwise convolution is to convolute each channel independently, while the pointwise convolution is to fuse features across channels. The depthwise separable convolution is widely used in lightweight networks such as MobileNet [21] and FastDepth [8] due to its fewer parameters. Moreover, compared with the standard convolution, the depthwise separable convolution pays more attention to the fusion of features in a single channel. Since the pixel shuffle operation improves the resolution by reshaping multiple channels into one channel, this paper uses the depthwise separable convolution on the output of pixel shuffle layer not only to reduce the number of network parameters but also to improve the feature fusion effect of pixel shuffle. In addition, the dilated convolution [22] can increase the receptive field of the convolution kernel without introducing extra parameters. Therefore, in this paper, depthwise dilated separable convolutions are used at the decoder part to obtain a larger receptive field and multiscale fusion feature while keeping as few parameters as possible.

3. Method

3.1. The Overview of Our Method. In recent years, the U-shaped network [8] has been commonly used in most monocular depth estimation methods. On this basis, we
propose the pixel shuffle encoder-decoder convolution neural network (PSDNet). As shown in Figure 1, PSDNet uses ResNet [12] as the backbone network to extract features from the input image. Three EESP connection modules are used to transfer information from the three residual blocks of the feature extraction backbone to the upsampling modules of the decoder network. The decoder network contains four upsampling modules, that is, pixel shuffle decoder (PSD). Besides, the residual connections between every two adjacent PSD modules, which consist of bilinear interpolation and a $5 \times 5$ convolution layer, are designed to enhance information sharing and alleviate the gradient vanishing problem. Finally, the depth map is obtained by using a $3 \times 3$ convolution layer on the output of the last PSD module, and then the resolution of the prediction map is increased to the size of the input image using bilinear interpolation.

3.2. EESP Connection Module. Motivated by the feature fusion method of the spatial pyramid in lightweight network [23], we design different EESP skip connection modules to bring more comprehensive information to the decoder. The dilated convolution [22] increases the receptive field while avoiding a surge in the number of parameters. The depthwise separable convolution [20, 21] can also reduce the number of parameters by separating it into a depthwise convolution and a pointwise convolution. The combination of the two methods not only increases the receptive fields but also reduces the number of parameters. Therefore, it is much lighter and more efficient than other feature fusion methods. The EESP module extracts features using depthwise separable convolutions with different dilation rates and fuses the extracted features using the hierarchical feature fusion method HFF [23]. In HFF, feature maps from the branch with the lowest dilation rate are combined with the feature maps from the branch with the next highest dilation rate. S__hen, the features of all branches are concatenated as the output of HFF.

3.3. Pixel Shuffle Decoder. Pixel shuffling is widely used in the field of image super-resolution processing [17]. By rearranging the input features, pixel shuffling can not only play the role of upsampling but also reduce the number of channels. The reduction of the number of channels can also greatly reduce the parameters of the subsequent convolution layer. In addition, compared with other upsampling methods (such as deconvolution [15]), it has no parameters. Therefore, we design the PSD module based on pixel shuffling.

The structure of the PSD modules (denoted as PSD2, PSD3, and PSD4 in Figure 1) is shown in Figure 3. The PSD module first adds the features from the previous PSD module and the output features of the corresponding EESP connection module. For the PSD1 (see Figure 1), there is only one input, which is the deepest level features output by the last residual block, and the other structures are the same as in Figure 3. The added features double the number of channels using a $1 \times 1$ convolution layer, and then the pixel shuffle unit changes the feature map to $2H \times 2W \times C/2$; that is, the length and width of the input features are doubled, while the number of channels is reduced to one-fourth of the original. Pixel shuffle can improve the resolution and reduce the number of channels. At the same time, it can achieve the effect of feature fusion by disrupting the feature values. However, pixel shuffling also destroys the connections between feature values in each channel, so this paper uses depthwise separable convolution to reconstruct the connections between feature values. In the proposed PSD module, after the pixel shuffle layer, a $5 \times 5$ depthwise separable convolution and a $3 \times 3$ depthwise separable convolution with a dilation rate of 2 are designed in parallel to establish new connections in each channel.
The feature maps of these two convolution branches are summed and then further fused using a $3 \times 3$ convolution to get the final output of the PSD module. The process is shown as follows:

$$Z = \varphi_{ps} (\text{Conv}_{1 \times 1} (Y + x)),$$

$$\text{PSD} = \text{Conv}_{3 \times 3} (\text{DDConv}_{5 \times 5, 1} (Z) + \text{DDConv}_{3 \times 3, 2} (Z)),$$

where $x$ denotes the input feature maps from the previous PSD module and its residual connection, $Y$ denotes the input feature from the EESP module, and $\varphi_{ps}$ denotes pixel shuffle operation.

### 3.4. Loss Function

In order to improve the sharpness of the object edges in the predicted depth map, the loss function $L_{\text{total}}$ is composed of three parts: the $L_1$ loss function, the gradient loss function, and the $L_2$ regularization term, as shown in equation (5). $\alpha$ is set to 0.5 and $\beta$ to 0.0001 in our experiments.

$$L_{\text{total}} = L_1 + \alpha L_{\text{gradient}} + \beta L_{2-\text{normal}},$$  

where $L_1$ loss function calculates the absolute error between the predicted value $\tilde{y}_i$ and the true value $y_i$, as shown in equation (2), where $i$ is the pixel index and $N$ denotes the number of pixels in the depth map. $L_1$ loss function measures the overall error between the prediction and the
The gradient loss function is shown in equation (5), where \( \nabla_h y_i \) and \( \nabla_v y_i \) denote the horizontal gradient and vertical gradient of the predicted depth map, respectively. The gradient of the depth map reflects the change rate of depth values. This loss function is designed to make the depth change of the predicted map more real.

\[
L_{\text{gradient}} = \frac{1}{2N} \sum_{i=1}^{2N} \left( |\nabla_h y_i - \nabla_h \tilde{y}_i| + |\nabla_v y_i - \nabla_v \tilde{y}_i| \right). 
\]

4. Experimental Results and Analysis

4.1. Experimental Settings

4.1.1. Dataset. We evaluate our method on the NYU depth V2 [24] indoor scene dataset, which is commonly used for monocular depth estimation. The NYU depth V2 contains 464 indoor scenes with 249 scenes for training and 215 scenes for testing.

4.1.2. Implementation Details. To improve the accuracy of our proposed network and prevent overfitting, an online data augmentation method is adopted. The specific operations of data augmentation are randomly rotating the RGB-D image with a rotation angle range of \([-5, 5]\); randomly adjusting the brightness, contrast, and saturation of the RGB image in the range of \([0.6, 1.4]\); and randomly flipping the RGB image and the depth truth map horizontally with 50% probability. We train on a single NVIDIA GeForce RTX 2080Ti with 11 GB of GPU memory. The experiments are conducted on the TensorFlow framework. The Adam optimizer is used for training, with an initial learning rate of 0.0001 and a decaying learning rate of 0.1 per 5 iterations. The batch size is set to 8, and the maximum number of iterations is 30.

4.1.3. Evaluation Metrics. The following evaluation metrics are used to measure the performance of monocular depth estimation methods: absolute relative error (rel, lower is better), root mean squared error (rms, lower is better), log mean error (log 10, lower is better), and threshold accuracy (\(\delta_1\), \(\delta_2\), and \(\delta_3\), higher is better). The functional expressions of the evaluation metrics are shown as follows.

\[
\begin{align*}
\text{rel} &= \frac{1}{N} \sum_{i=1}^{N} \left| \frac{y_i - \tilde{y}_i}{y_i} \right|, \\
\text{rms} &= \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left( y_i - \tilde{y}_i \right)^2}, \\
\log 10 &= \frac{1}{N} \sum_{i=1}^{N} \left| \log y_i - \log \tilde{y}_i \right|, \\
\delta &= \max \left( \frac{y_i}{\tilde{y}_i}, \frac{\tilde{y}_i}{y_i} \right) < T, \ T \in \{1.25, 1.25^2, 1.25^3\}. 
\end{align*}
\]

4.1.4. Compared with Other Advanced Methods. The depth estimation test results on the NYU depth V2 dataset of our method and other state-of-the-art monocular depth estimation methods based on deep learning in recent years are reported in Table 1. Other methods involved in the comparisons are the full convolutional network method of Laina et al. [6], the conditional random field optimization of superpixel depth proposed by Liu et al. [7], the cascaded conditional random field depth optimization method of Xu et al. [25], the efficient and lightweight encoder-decoder network architecture proposed by Wofk et al. [8], the gradient optimization proposed by Li et al. [26], the
multiscale feature fusion method of Xu et al. [27], the augment ordinal depth relationship methods of Cao et al. [28], the method based on the geometric cues and scene parsing of He et al. [11], the successive encoder-decoder style subnetworks proposed by Dong et al. [29], and the attention mechanism and multiscale feature fusion method proposed by Xu et al. [9]. Different from the feature fusion methods [9], our multiscale feature fusion method transfers the features of different receptive fields to the decoder network, constructing the connections between the encoder and the decoder. Among them, the methods of Laina et al. [6], Wofk et al. [8], Xu et al. [9], Cao et al. [28], Dong et al. [29], and this paper do not perform other additional refinement steps, while Liu et al. [7], Xu et al. [25], and Xu et al. [27] used conditional random fields to do postprocessing. Li et al. [26] used image gradients for depth map optimization, and He et al. [11] used the geometric constraints and semantic information of the scene to alleviate the ambiguity in monocural depth estimation. As can be seen from the results in Table 1, the method in this paper achieves competitive results in all indicators.

Figure 4 illustrates some depth estimation results of our method and Laina et al. [6], where column (a) indicates the input RGB image, column (b) is the predicted depth map of Laina et al. [6], column (c) is the depth prediction map of this paper, and column (d) indicates the depth truth map. The brighter the color of the pixel points, the smaller the depth value, and the darker color, the larger the depth value. Observing the prediction maps in Figure 4, we can find that the performance of our method is better than that of Laina et al. [6] in terms of local depth values. For example, the depth estimation results marked by the blue rectangular box in column (c) indicate that our method can better predict the depth of small objects such as chair legs and table lamps. And the results within the green box in column (c) shows that the prediction map of our method has clearer edges than that of Laina et al. [6]. Specifically, the outline of the rocking chair in the second row can be clearly seen in our predicted depth map.

5. Ablation Experiments

In this section, we conduct experiments to illustrate the effectiveness of each component of our PSDNet.

### 5.1. Comparison of Different EESP Modules

In this experiment, we compare the depth estimation effects of several other structure options of the three EESP modules (see Figure 1), and the experimental results are shown in Table 2. In Table 2, “3-EESP” indicates that the three EESP connection modules adopt the same design as in Figure 2; that is, they all use four depthwise separable convolutions with dilation rates of 1, 2, 4, and 8, respectively. “EESP-4” indicates one depthwise separable convolution with the dilation rate of 1 for EESP1, two depthwise separable convolutions with dilation rates of 1 and 2, respectively, for EESP2, and three depthwise separable convolutions with dilation rates of 1, 2, and 4, respectively, for EESP3. "EESP-HFF" indicates that the HFF fusion is removed from the EESP modules, and others remain the same as in our proposed design, in which EESP1 has two depthwise separable convolutions with dilation rates of 1 and 2, respectively, for EESP2, and three depthwise separable convolutions with dilation rates of 1, 2, and 4, respectively, for EESP3. From the comparison results with “3-EESP” and “EESP-4,” it can be seen that our proposed design schema obtains the best performance due to features of different receptive fields fused in a complementary fashion. Compared with the experimental results of “EESP-HFF,” the rms is improved after using HFF fusion, while the rel remains unchanged. Since the rel is more sensitive to the smaller depth area than the rms, the improvement of rms can prove that HFF fusion can slightly improve the prediction accuracy of the larger depth area.

### 5.2. Comparison of Different Feature Fusion Methods in PSD Modules

In order to study how different fusion methods after pixel shuffling in PSD modules affect the performance of the model, we compare different design options. The results are compared in Table 3, where “A” indicates that only the standard 5 × 5 convolution is used for feature fusion after pixel shuffling; “B” indicates that only the depthwise separable convolution with a kernel size of 5 × 5 and a dilation rate of 1 is used for feature fusion after pixel shuffling; “C” means that, after pixel shuffling, the features are input in parallel into a depthwise separable convolution of 3 × 3 with a dilation rate of 2 and a depthwise separable convolution of 5 × 5.
5 × 5 with a dilation rate of 1, and then simply add the outputs of these two branches as the result features; “D” is similar to the method proposed in Figure 3, except that the dilation rate of the 3 × 3 depthwise dilated separable convolution is reduced to 1. From the results in Table 3, it can be found that the “proposed” structure in Figure 3 has the best performance. It demonstrates that the proposed method can further integrate the features of different receptive fields and reconstruct the relationship between different channels.

5.3. Ablation Experiments on Decoder Network. The decoder mainly contains three parts: the EESP connection modules, the PSD modules, and the residual connections. This section verifies the effects of the three modules on the depth estimation performance by ablation experiments. The experimental results are shown in Figure 5. Compared with the predicted depth maps in column (b) of Figure 5, the sharpness of object edges in the predicted depth maps in column (c) of Figure 5 is significantly improved, and the performance of predicting small objects depth values, such as table lamps and chair legs is enhanced, which proves that the EESP connection module can effectively improve the depth prediction performance of the network for object edges and small objects. Comparing the areas of the blue box in column (c) of the last row in Figure 5 with the corresponding areas in columns (b) and (e), it can be seen that the EESP connection modules disrupt the continuity of depth values in the depth prediction maps, while the combination of the EESP modules and the

---

**Table 2: Comparison of different EESP modules.**

| Method      | Errors | Accuracy (%) |  |  |  |
|-------------|--------|--------------|---|---|---|
|             | rel    | rms | δ₁ | δ₂ (%) | δ₃ (%) |
| 3-EESP      | 0.142  | 0.505 | 81.2 | 95.9 | 98.8 |
| EESP-4      | 0.144  | 0.505 | 80.8 | 95.8 | 98.9 |
| EESP-HFF    | 0.141  | 0.512 | 81.3 | 95.6 | 98.8 |
| Proposed    | 0.141  | 0.503 | 81.3 | 96.0 | 99.0 |

**Table 3: Comparison of different feature fusion methods in PSD.**

| Method | Errors | Accuracy (%) |  |  |  |
|--------|--------|--------------|---|---|---|
|        | rel    | rms | δ₁ | δ₂ (%) | δ₃ (%) |
| A      | 0.151  | 0.522 | 79.3 | 95.5 | 98.9 |
| B      | 0.147  | 0.513 | 80.1 | 95.8 | 98.9 |
| C      | 0.178  | 0.607 | 73.4 | 93.0 | 98.0 |
| D      | 0.147  | 0.515 | 80.1 | 95.6 | 98.9 |
| Proposed | 0.141  | 0.503 | 81.3 | 96.0 | 99.0 |

**Figure 4:** Qualitative results on the NYU depth V2 dataset. (a) RGB inputs; (b) results of Laina et al. [6]; (c) results of our method; (d) ground truth depth maps. The brighter the color of the pixel points, the smaller the depth value, and the darker the color, the larger the depth value.
residual connections can effectively improve the continuity of predicted depth values. Compared with the predicted depth maps in columns (b), (c), and (d) in Figure 5, the areas marked in red and green boxes in column (e) reflect that the predicted depth maps obtained by using these modules jointly not only have clearer object contours but also greatly improve the problem of missing small objects.

6. Conclusion

A depth estimation encoder-decoder architecture based on spatial pyramid EESP and pixel shuffle is proposed in this paper to address the problems of object distortion and missing small objects existing in monocular depth estimation. The spatial pyramid EESP modules are used to fully utilize the features of different scales. The proposed pixel shuffle decoder upsamples the features extracted from the backbone network and generates the depth prediction map by fusing the features of different scales step by step. Compared with other state-of-the-art methods, the depth map estimated by our method has clearer object contours, less object distortion, and fewer small objects missing. The experimental result demonstrates the role of the EESP connection module and residual connection in feature fusion and verifies the reliability of our method in solving the problem of missing and blurred contours of small objects. However, the depth estimation performance of our method is not desirable in areas of very small depth values and objects refracting light such as mirrors, so the next step will be to try masking and other methods to improve the accuracy of depth estimation.

Data Availability

The data used to support the findings of this study are available at https://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was funded in part by the National Natural Science Foundation of China under Grant nos. 61462035 and 61862031 and Training Plan for Academic and Technical Leaders of Major Disciplines in Jiangxi Province (no. 20213BC22004).

References

[1] S. Zia, B. Yuksel, D. Yuret, and Y. Yemez, “Rgbd object recognition using deep convolutional neural networks,” in Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 896–903, IEEE, Venice, Italy, October 2017.
[2] Y. Lu, J. Zhou, J. Wang et al., “Curve-structure segmentation from depth maps: A cnn-based approach and its application to exploring cultural heritage objects,” in Proceedings of the AAAI Conference on Artificial Intelligence, February 2018.
[3] T. Yan, P. Wu, Y. Qian, Z. Hu, and F. Liu, “Multiscale fusion and aggregation pccn for 3d shape recovery,” Information Sciences, vol. 536, pp. 277–297, 2020.
[4] I. Bouazaoui, S. A. R. Florez, and A. El Ouardi, “Enhancing rgb-d slam performances considering sensor specifications for indoor localization,” IEEE Sensors Journal, vol. 22, no. 6, pp. 4970–4977, 2022.
[5] D. Eigen, C. Puhrsch, and R. Fergus, Depth Map Prediction from a Single Image Using a Multi-Scale Deep Network, MIT Press, Cambridge, MA, USA, 2014.
[6] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, “Deeper depth prediction with fully convolutional residual networks,” in Proceedings of the Fourth international
[7] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258, IEEE, Honolulu, HI, USA, July 2017.

[8] A. G. Howard, M. Zhu, B. Chen et al., “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” 2017, http://www.w3.org/2001/XMLSchema:anyURI.

[9] D. Eigen and R. 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, pp. 2650–2658, IEEE, Santiago, Chile, December 2015.

[10] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, “Indoor segmentation and support inference from rgbd images,” in Proceedings of the European conference on computer vision, pp. 746–760, Springer, New York, NY, USA, 2012.

[11] K. Li, Z. Fu, H. Wang, Z. Chen, and Y. Guo, “Adv-depth: Self-supervised monocular depth estimation with an adversarial loss,” IEEE Signal Processing Letters, vol. 28, pp. 678–682, 2021.

[12] D. Eigen and R. 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, pp. 2650–2658, IEEE, Santiago, Chile, December 2015.

[13] J. Hu, M. Ozay, Y. Zhang, and T. Okatani, “Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries,” in Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1043–1051, IEEE, Waikoloa, HI, USA, Jan. 2019.

[14] A. Odena, V. Dumoulin, and C. Olah, “Deconvolution and checkerboard artifacts,” Distill, vol. 1, no. 10, p. e3, 2016.

[15] X. Feng, X. Li, and J. Li, “Multi-scale fractal residual network for image super-resolution,” Applied Intelligence, vol. 51, no. 4, pp. 1845–1856, 2021.

[16] X. Zhao, Y. Zhang, T. Zhang, and X. Zou, “Channel splitting network for single mr image super-resolution,” IEEE Transactions on Image Processing, vol. 28, no. 11, pp. 5649–5662, 2019.

[17] Y. Sun, J. Chen, Q. Liu, and G. Liu, “Learning image compressed sensing with sub-pixel convolutional generative adversarial network,” Pattern Recognition, vol. 98, no. 12, Article ID 107051, 2020.

[18] Z. Tian, T. He, C. Shen, and Y. Yan, “Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 2020.

[19] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258, IEEE, Honolulu, HI, USA, July 2017.

[20] A. G. Howard, M. Zhu, B. Chen et al., “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” 2017, http://www.w3.org/2001/XMLSchema:anyURI.

[21] F. Yu and V. Koltun, “Multi-scale context aggregation by dilated convolutions,” 2015, http://www.w3.org/2001/XMLSchema:anyURI.