VR Image Depth Estimation Method Based on ResNeXt and Spatial Pyramid

Hua Li
Department of Animation Art, Zibo Polytechnic University, Zibo 255300, China
E-mail: lh401281894@126.com

Keywords: depth estimation, virtual reality, ResNeXt, spatial pyramid pooling, mixed attention mechanism

Received: May 22, 2025

The rapid development of virtual reality technology has made depth estimation the key to enhancing immersion, but existing methods still suffer from insufficient multi-scale feature fusion and detail loss in complex scenes, leading to a decrease in depth map accuracy. To this end, a deep estimation network model based on Residual Networks with Next Generation (ResNeXt) and spatial pyramid modules is proposed. This model combines the efficient feature extraction of ResNeXt with the multi-scale fusion capability of Extremely Efficient Spatial Pyramid (EESP) module, combined with a hybrid attention mechanism, to improve depth estimation accuracy while optimizing computational efficiency. The experimental results show that the parameter count and floating-point operations of the proposed model in the training set are 19.2M and 33.2G, respectively, with an inference speed of 46FPS, demonstrating its robustness in indoor, outdoor, and low light environments. In addition, the model exhibits excellent performance in indoor, outdoor, and low light environments. In indoor scenes, the mean square error of the model is 0.045, the peak signal-to-noise ratio is 38.5, and the structural similarity is 0.92, which is 8% higher in accuracy than the baseline model. The results indicate that the method proposed by the research provides an effective and efficient solution for high-precision depth estimation in virtual reality applications.

Povzetek: Prispevek uvaja mrežo EESP-ResNeXt, ki združuje ResNeXt, prostorsko piramido in hibridni pozornostni mehanizem za učinkovito oceno globine VR slik v zahtevnih okoljih.

1 Introduction

Virtual Reality (VR) technology is rapidly developing, especially in fields such as entertainment, healthcare, and education, demonstrating enormous potential [1-2]. Depth maps play a crucial role in VR, autonomous driving, robot navigation, and other fields by recording the spatial separation between each point within the scene and the camera, providing basic 3D information for tasks such as 3D reconstruction, object recognition, and path planning. Although traditional depth estimation methods relying on stereo vision, structured light, or LiDAR have high accuracy, they have limitations such as high hardware costs and poor environmental adaptability. Therefore, how to enhance the precision and computational efficiency of depth estimation through software algorithms has emerged as a pressing issue that demands immediate resolution within the realm of computer vision. Pintore G et al. proposed a geometric information extraction and rendering method based on a single spherical panorama, significantly enhancing the 3D immersive experience of VR applications. This method adopted an end-to-end deep learning framework to synchronously predict scene depth and room structure, and optimized network performance through pre training with synthesized data. Its lightweight design enabled real-time interactive panoramic view generation and supported perspective transformation synchronized with head movements. The outcomes of the experiments revealed that the proposed approach surpassed the currently available methods with respect to latency and accuracy, and performed well on mainstream indoor panoramic datasets [3]. Cai Y et al. introduced a real-time 6DoF video processing system that integrated three core technologies: unsupervised multi-view depth estimation, real-time virtual view rendering, and 6DoF video encoding. Experimental data showed that the system achieved significant algorithm acceleration, increasing depth estimation speed by 34 times and improving the efficiency of Depth Image-Based Rendering (DIBR) algorithm by 168 times [4]. Liu et al. developed an unsupervised ship depth estimation method based on monocular drone images. This method first utilized realistic rendering techniques to construct a specialized training dataset with uniform lighting and a clean background. Subsequently, a lightweight knowledge distillation network based on a differentiable rendering framework was designed to achieve accurate ship depth estimation through unsupervised learning. The outcomes of the experiments revealed that the proposed approach surpassed the currently available methods with respect to accuracy while maintaining a more compact model volume [5].

In recent years, deep learning-based depth estimation methods have gradually become mainstream, making significant progress by learning to directly predict depth maps from monocular or multi-view images [6]. The

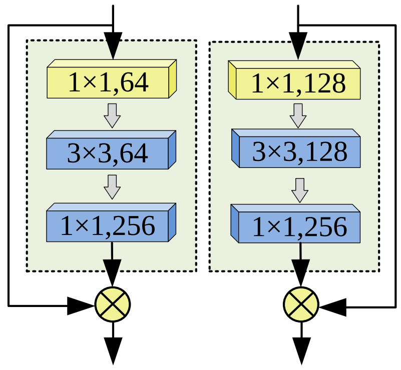
Residual Networks with Next Generation (ResNeXt) network, as an improved residual network, has strong feature extraction capabilities and can effectively avoid the computational bottleneck caused by traditional deep networks as depth increases [7]. Khan et al. proposed an improved ResNeXt deep convolutional network for Bengali handwritten composite character recognition. This model embedded a squeeze excitation module in the traditional ResNeXt architecture, effectively integrating spatial information within the local receptive field and inter channel dependencies by dynamically adjusting channel feature weights. The experiment outcomes indicated that the model achieved an average recognition accuracy of 99.82%, which was better than the current optimal method [8]. Hu et al. proposed an improved ResNeXt 3x1D deep residual network specifically designed for detecting abnormal behavior in aquaculture. The network was optimized based on the R(2+1)D convolutional architecture, and experiment outcomes indicated that the proposed ResNeXt 3x1D performed well in identifying abnormal behaviors in the field of aquaculture, with a recognition accuracy of 95.3% [9]. The Extremely Efficient Spatial Pyramid (EESP) module utilizes Spatial Pyramid Pooling (SPP) technology to enhance the model's ability to process multi-scale information by capturing image features at multiple scales [10]. Xiong et al. proposed an improved You Only Look Once version 3 (YOLOv3) traffic sign detection method, which enhanced detection performance by integrating SPP module and Adaptive Spatial Feature Fusion (ASFF) mechanism. This method introduced the SPP module in the feature extraction stage to fuse multi-scale contextual information, and used the ASFF module to optimize the gradient propagation of the feature pyramid in the detection stage. Experimental results showed that this SPP-ASFF-YOLOv3 architecture significantly improved the detection accuracy of the original YOLOv3 network [11]. Zhao et al. proposed a dual branch network based on multi-scale dilated fusion for real-time semantic segmentation. Its core consisted of three innovative modules: The semantic guided spatial detail module improved boundary accuracy and fine-grained classification, the multi-scale dilated pyramid module integrated multiple dilation rate features, and the bilateral fusion module optimized feature fusion through cross weighting. The experiment showed that the network achieved a good balance between accuracy and speed [12].

In summary, although existing depth estimation models perform well in some static scenarios, they still face challenges in complex environments. For example, although Pintore G et al. [3] proposed a method of jointly estimating depth and room layout using a single spherical panoramic image, it significantly enhanced indoor VR immersion. However, this model was tailored for omnidirectional indoor views and lacked robust multi-scale feature fusion, which limited its generalization ability to complex outdoor scenes. Xiong S et al. [11] and Zhao S et al. [12] integrated SPP or expansion fusion modules into detection or segmentation tasks to capture multi-scale context, but did not include explicit attention mechanisms. To further improve the accuracy and robustness of VR image depth estimation, a new depth estimation network structure is proposed by innovatively combining ResNeXt and EESP modules. Compared with the limited applicability of the 6DoF real-time video system based on multi-view depth prediction proposed by Cai Y et al. [4] for monocular depth estimation, the proposed method solves the monocular constraint problem by using ResNeXt's grouped convolution for efficient feature extraction, and introduces spatial pyramid fusion to maintain cross scale scene structure. This module is capable of capturing rich spatial features at multiple scales, especially in complex scenes and low lighting conditions, demonstrating high depth estimation accuracy and computational efficiency.

2 Methods and materials

2.1 ResNeXt combined with spatial pyramid network construction

ResNeXt is a Convolutional Neural Network (CNN) architecture based on deep residual learning, aimed at improving the performance and efficiency of the network. By introducing the idea of grouped convolution, the standard convolution operation is divided into multiple smaller groups to lower the parameter number and improve computation speed. Compared with the traditional ResNet architecture, ResNeXt has higher flexibility and better performance, and the structural comparison between the two is shown in Figure 1 [13].



(a) ResNet (b) ResNeXt

Figure 1: Structural comparison between ResNet and ResNeXt.

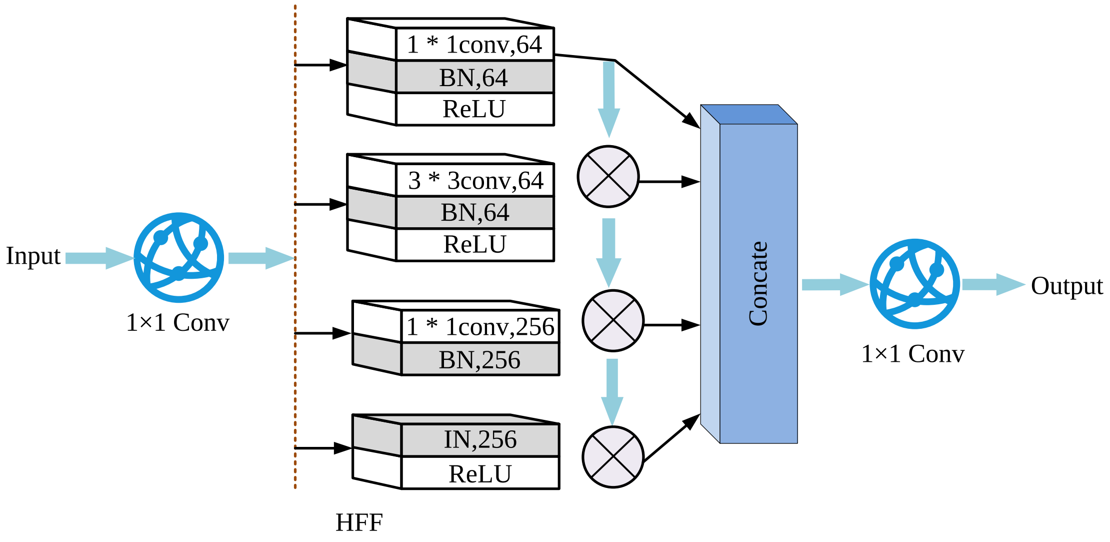


Figure 2: EESP module structure.

Figures 1 (a) and 1 (b) show the residual structures of ResNet and ResNeXt, respectively. Both architectures employ similar designs. They include a 1×1 convolutional layer, a 3×3 convolutional layer, and the final 1×1 convolutional layer. However, ResNeXt introduces the concept of Cardinality. Cardinality refers to the number of groups in grouped convolutions. When compared to simply increasing the network depth, this grouped convolution method has dual advantages. It not only reduces the computational complexity but also improves the model efficiency. Additionally, it avoids the training difficulty and hardware burden that increasing depth could cause. This method enables ResNeXt to enhance the model's capability to capture different feature representations without a substantial increase in the number of parameters. The ResNeXt expression is shown in equation (1) [14].

$$y = x + \sum_{i=1}^{C}(f_i(x)) \qquad (1)$$

In equation (1), y represents the output feature map (FM), x represents the input FM, C represents the number of groups, $f_i(x)$ represents the FM processed through a group convolution operation. To reduce computational costs, the EESP module is introduced to further improve the performance of the network. ResNeXt enhances the network's expressive power through grouped convolution and C, enabling the network to learn rich feature representations more efficiently. The EESP module uses Depthwise Dilated Separable Convolution (DDSC) with multiple dilation rates to capture multi-scale features, but the interval sampling characteristics of dilated convolutions may result in some pixels being missed, leading to grid artifacts. The EESP module structure is shown in Figure 2 [15].

In Figure 2, R means the expansion rate. This module adopts the Hierarchical Feature Fusion (HFF) mechanism to eliminate grid artifacts. Its core idea is to stack convolution results with different dilation rates step by step, and finally concatenate the fused features and compress them through 1×1 convolution. After convolution operations, batch normalization (BN) and nonlinear activation operations are performed. BN is employed to expedite convergence and enhance the stability of the training process. Meanwhile, the activation function imparts nonlinear properties to the model, preventing it from degenerating into mere linear transformations. This, in turn, safeguards the network's expressive capabilities and bolsters its classification performance. Introducing EESP module in skip connections can effectively fuse multi-scale features. The size of the FM in the encoder stage decreases while the number of channels increases: Shallow features have higher resolution, fewer channels, and retain fine-grained information such as texture. Deep level features have low resolution and multiple channels, containing abstract semantic information. The EESP module dynamically adjusts the dilation rate of depthwise separable convolutions to achieve collaborative optimization of global semantics and local details. The formula for calculating the size of the dilated convolution filter is shown in equation (2).

$$S = R \cdot (size - 1) + 1 \tag{2}$$

In equation (2), S is the size of the convolutional filter and $size$ is the size of the input FM.

2.2 EESP-ResNeXt structure integrating hybrid attention mechanism

On the basis of ResNeXt combined with spatial pyramid network, in order to further optimize the effectiveness of multi-scale feature fusion, a hybrid attention mechanism (AM) fusion EESP-ResNeXt structure is proposed. The quality of feature fusion directly determines the generation effect of depth maps, and the multi-scale features extracted by the EESP module are enhanced in the decoding stage by combining them with deep features. To achieve this objective, a hybrid mechanism that integrates both spatial and channel attention is implemented within the decoder. This mechanism not only adeptly captures the interdependencies among features but also dynamically enhances the weights of crucial features. As a result, it enables more precise extraction of the most pertinent feature information for depth estimation tasks. The structure is shown in Figure 3.

In Figure 3, X_{SD} represents the input of the spatial attention module and W_D the width of the deep FM. The hybrid AM adopts a serial processing approach, first analyzing the spatial distribution relationship of the scene through the EESP module, focusing on enhancing the weight distribution of key regions, and guiding the model to focus on salient regions. The input of this module is composed of shallow and deep features, which are convolved by 1×1 to reduce the dimensionality to 1/2 channel and then concatenated to form the input features of the spatial attention layer. The flowchart of spatial AM and channel AM is shown in Figure 4 [16-17].

Figure 4 (a) shows the spatial AM, and Figure 4 (b) shows the channel AM. Firstly, the input features are reduced to half of the original number of channels through a 1×1 convolution. Then, average pooling and max pooling are performed on the channel dimension to obtain two types of spatial description information. After stitching them together, the spatial attention map is generated through convolution and Sigmoid activation function to highlight key areas in the scene. The weighted features are input into the channel attention module, global average pooling is first input, and the average activation value of each channel is obtained. Then the channel weight map is calculated through 1×1 convolution and Sigmoid function. Finally, the feature maps are weighted by channel wise multiplication to enhance semantically significant channel responses. This two-stage mechanism enables the model to simultaneously focus on key regions in space and semantic features in channels, thereby effectively improving the feature expression ability in depth estimation tasks. In the spatial AM, input feature X_{SD} is compressed after average pooling and max pooling, and then concatenated to form a new feature representation, whose mathematical expression is shown in equation (3) [18].

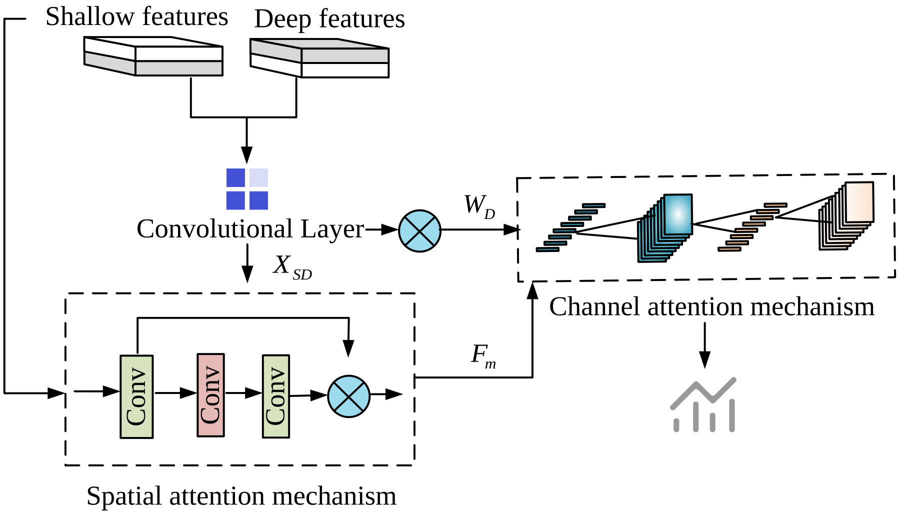
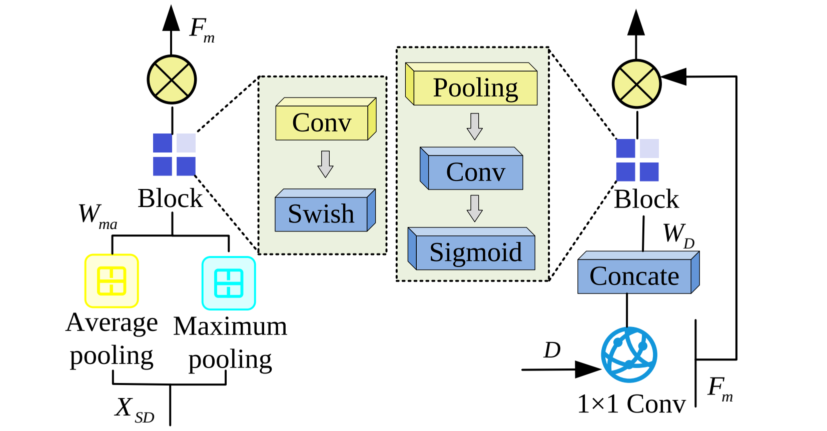


Figure 3: Module structure of hybrid AM.



(a) Spatial attention mechanism (b) Channel attention mechanism

Figure 4: Spatial AM and channel AM.

$$W_{ma} = [\max(X_{SD}), avg(X_{SD})] \qquad (3)$$

In equation (3), W_{ma} represents the new features obtained after max pooling and average pooling. Then, using convolution operation combined with Sigmoid nonlinear transformation, the expression is shown in equation (4).

$$W_{Space} = \sigma(Conv(W_{ma})) \qquad (4)$$

In equation (4), W_{Space} represents the spatial feature weight map and σ represents the activation function, $Conv$ represents convolution operation on W_{ma}. The input feature X_{SD} is point multiplied with the spatial weight map W_{Space}, and the output FM is shown in equation (5).

$$F_m = X_{SD} \otimes W_{Space} \qquad (5)$$

In equation (5), F_m represents an FM of size $H \times W \times CS$, and $\otimes$ represents dot multiplication operation. In CNN, each channel corresponds to a specific image feature extracted, but a single channel of shallow features can only capture limited geometric information. The channel AM dynamically adjusts the weights of each channel by analyzing cross channel feature relationships, achieving channel level calibration of FMs. Similar to the spatial attention module, the first step is to reduce the number of channels of feature D to CS through 1×1 convolution, and then concatenate them with F_m to generate features as shown in equation (6).

$$W_D = [D, F_m] \qquad (6)$$

According to equation (6), global average pooling is used followed by 1×1 convolution and Sigmoid activation, and the channel weight map of output $1 \times 1 \times CS$ is shown in equation (7).

$$W_T = \sigma(Conv(avg(W_J))) \qquad (7)$$

In equation (7), W_T represents the channel feature weight map, W_J represents the output of a certain layer in a convolutional network. To enable the model to dynamically select effective features, dot multiplication operation between W_T and F_m is required, as shown in equation (8).

$$Y_i = W_T \otimes F_m, i \in [1,3] \qquad (8)$$

In equation (8), Y_i represents the FM with a size of $H \times W \times CS$. The D1, D2, and D3 nodes of the decoder network all integrate a hybrid AM. Among them, D1 outputs FM X1 as D2 input, and the processing flow of each node is the same. This design effectively enhances the network's attention to key features by integrating shallow and deep features. Based on this, the EESP-ResNeXt network model was proposed for VR image depth estimation, as shown in Figure 5.

In Figure 5, the network architecture consists of three core components: feature extraction module, ESPP module, and hybrid attention module. The encoder uses ResNeXt as the base network, and the residual module adopts a grouped convolution design. During downsampling, the FM resolution is reduced by 50% while the number of channels is doubled. In addition, the network embeds ESPP modules at cross layer connections to achieve multi-scale feature fusion and optimize the decoding process. This network adopts an unsupervised training method and can select left/right views as inputs. Left and right disparity maps are generated separately through convolutional networks, and these disparity maps are used to reconstruct corresponding views. By comparing the differences between the real view and the reconstructed view, the LOSS value is calculated, and the network parameters are continuously optimized based on the backpropagation algorithm, ultimately achieving end-to-end training of the model. The overall loss function is defined as shown in equation (9).

$$L_{sum} = \sum_{s=1}^{4} L_s \qquad (9)$$

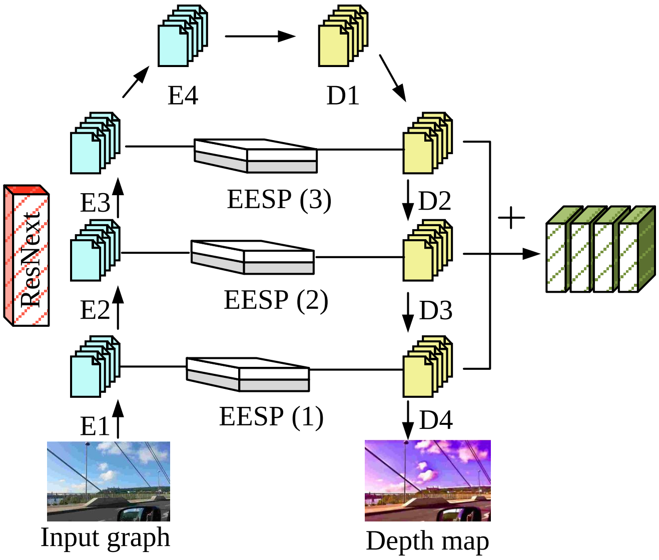


Figure 5: EESP-ResNeXt network model (Source from : https://www.1001freedownloads.com/free-photo/perspective-w-rzburg-russian-fortress-bridge).

Table 1: Training configuration and implementation details.

Parameter	Value
Batch size	16
Epochs	40
Input resolution	256 × 512
Weight initialization	He normal initialization
Framework	PyTorch 1.13
GPU	NVIDIA RTX 3090

Table 2: Results of ablation experiment.

Test group	RMSE	REL	δ 1
Group 1	0.852	0.152	0.872
Group 2	0.817	0.143	0.891
Group 3	0.793	0.138	0.902
Group 4	0.768	0.13	0.915
Group 5	0.744	0.125	0.928
Group 6	0.716	0.118	0.942

In equation (9), L_{sum} represents the total loss function, L_s integrates appearance similarity loss, disparity smoothing constraint, and left-right disparity matching loss. The specific expression is shown in equation (10).

$$L_s = \omega_a (L_a^l + L_a^r) + \omega_p (L_p^l + L_p^r) + \omega_{lr} (L_{lr}^l + L_{lr}^r) \tag{10}$$

In equation (10), L_a represents appearance matching loss, L_p represents disparity smoothness loss, and L_{lr} represents left-right disparity consistency loss. ω_a, ω_p, and ω_{lr} are the weight coefficients corresponding to the respective losses. The process of converting a disparity map into a depth map is shown in equation (11).

$$e' = \frac{df}{e} \tag{11}$$

In equation (11), e represents the disparity map of the given scene, e' represents the converted depth map, d represents the disparity map of the given scene, and f represents the focal length.

3 Results

3.1 EESP-ResNeXt network model performance testing

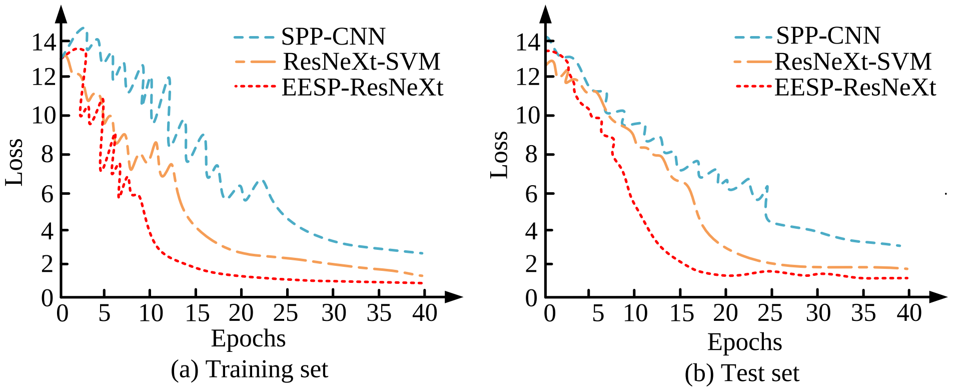
To verify the performance advantages of the EESP-ResNeXt network model proposed by the research, the KITTI dataset was selected for ablation experiments. The experiment was conducted in a Python environment, using TensorFlow and PyTorch frameworks to train and test the model. The hardware platform was a workstation equipped with NVIDIA RTX 3090 GPU, ensuring efficient training and inference speed. The hyperparameter configuration for training is shown in Table 1.

Table 1 shows the key hyperparameter settings, data preprocessing methods, optimizer selection, loss function composition, and hardware and software environment configuration during the training process. Based on this, ablation experiments were conducted for testing. The experiment used the standard ResNet-50 as the baseline model, which was the first group. Firstly, ResNeXt's grouped convolution structure was introduced in the second group to verify its improvement effect on feature expression ability. Subsequently, the third group added an EESP module without HFF mechanism based on ResNeXt to analyze the role of multi-scale feature extraction and the potential grid artifact problems it may cause. The fourth group further integrated HFF mechanism in the EESP module to specifically evaluate its effectiveness in eliminating grid artifacts. The fifth group extended the EESP module to skip connections and studies the contribution of multi-scale features in cross layer fusion. The final complete model of Group 6 integrated all core components, including grouped convolution, EESP module combined with HFF, hybrid AM, and skip connection optimization, to verify the collaborative performance improvement of the overall architecture. The root mean square error (RMSE), relative absolute error (REL), and threshold accuracy δ 1 were used as indicators for testing, δ1 represents the proportion of pixels with an error within the range of 1.25 times. The results are shown in Table 2.

According to the results in Table 2, from the baseline model ResNet-50 to the complete model, RMSE decreased from 0.852 to 0.716, REL improved from 0.152 to 0.118, and δ 1 increased from 0.872 to 0.942, indicating the cumulative contribution of each component to the model performance. After introducing grouped convolution in the second group, all three indicators showed significant improvement, indicating that the structure effectively enhanced the feature expression ability. The performance of the third group continued to improve after adding the basic EESP module, and the fourth group further optimized the results by integrating the HFF mechanism, confirming the important role of hierarchical feature fusion in eliminating grid artifacts. After introducing the EESP module in the skip connection, the model performed better in multi-scale feature fusion in Group 5. In the end, the complete model integrated all optimized components and achieved the best level in all evaluation metrics, with an accuracy of 0.942, an increase of 8 percentage points from the baseline, fully verifying the effectiveness of the overall architecture design. To further validate the performance of the proposed model, EESP-ResNeXt was compared with Spatial Pyramid Pooling Convolutional Neural Network (SPP-CNN) [19] and ResNeXt Support Vector Machine (ResNeXt-SVM)-based methods [20]. The KITTI dataset was divided into a training set and a testing set in a 7:3 ratio, and the variation of the loss function with the number of iterations is shown in Figure 6.

Figures 6 (a) and 6 (b) show the loss function variation curves of the three models on the training and testing sets, respectively. In Figure 6 (a), the EESP-ResNeXt model had the fastest loss reduction rate, indicating that its convergence speed during training was faster than that of the other two models. The loss reduction of the ResNeXt-SVM model was slower, while the loss reduction of the SPP-CNN model was the slowest, indicating its poor convergence effect during the training process. In Figure 6 (b), EESP-ResNeXt still exhibited a fast convergence speed, and in the later stages of training, the loss value tended to stabilize. The performance of the ResNeXt-SVM model on the test set was also relatively stable, but its loss value was slightly higher than that of EESP-ResNeXt. The SPP - CNN model had a high loss value on the test set and converged slowly, indicating poor generalization ability of the model. The study compared the introduction of Resnet 50 with Spatial Pyramid Pooling (Resnet50-SPP) [21] and the MobileViT-based depth (MViTDepth) model [22], using parameters, Floating Point Operations (FLOPs), and inference time as indicators. The computational efficiency results are shown in Table 3.

In Table 3, the MViTDepth model had the highest number of parameters of 25.3M and the highest floating-point operation of 43.5G, but its inference speed on the training set was only 39 FPS, indicating that it consumed a large amount of computing resources and had average speed performance. The parameters of the ResNet50 SPP model were 25.1M, FLOPs were 41.6G, and inference speed was 36 FPS. The overall performance was at a relatively low level among all models. In contrast, the SPP-CNN model had parameters of 24.6M, FLOPs of 40.3G, and inference speed of 38 FPS, slightly better than ResNet50 SPP. ResNeXt SVM performed better on the training set with parameters of 22.5M, FLOPs of 39.8G, and inference speed of 41 FPS. The EESP ResNeXt model had the fewest parameters, only 19.2M, FLOPs of 33.2G, and inference speed of 46 FPS, making it the fastest inference model on the training set. On the test set, EESP ResNeXt still maintained a leading position, with an inference speed of 57 FPS, parameters of 19.4M, and FLOPs of 33.7G, fully demonstrating its superior balance between model lightweighting and high inference efficiency. The inference speed of the ResNeXt SVM model on the test set was 52 FPS, with parameters of 21.5M and FLOPs of 37.8G, and its performance was also quite excellent. The SPP-CNN model had parameters of 23.7M, FLOPs of 41.9G, and inference speed of 45 FPS, showing average performance. MViTDepth and ResNet50 SPP both had 44 FPS on the test set, with FLOPs of 42.6G for MViTDepth and 40.2G for ResNet50 SPP. The parameters were 25.1M and 24.8M, respectively, both showing high resource consumption. The throughput of the three models on the training and testing sets is shown in Figure 7.



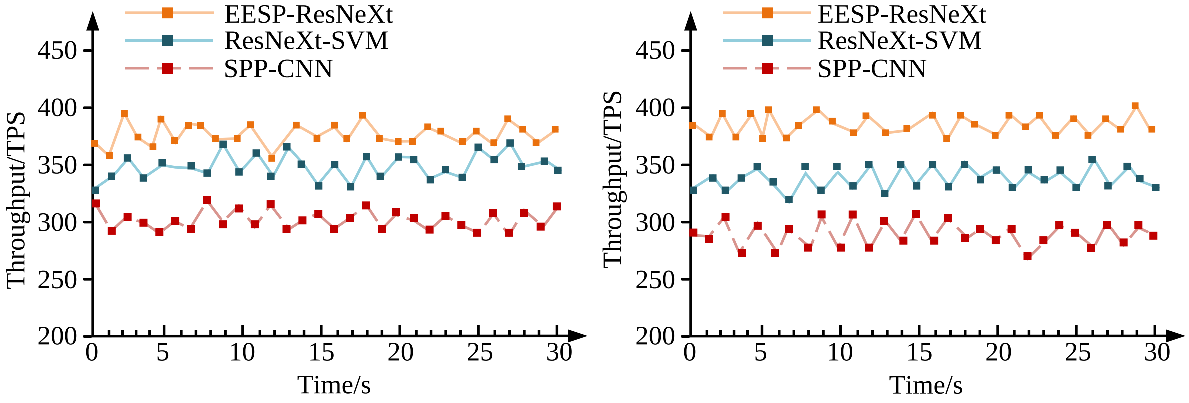
(a) Training set

(b) Test set

Figure 6: Change result of loss function.

Table 3: Comparison of computational efficiency results.

Data set	Model	Params (M)	FLOPs (G)	Reasoning speed (FPS)
Training set	MViTDepth	25.3	43.5	39
	Resnet50-SPP	25.1	41.6	36
	SPP-CNN	24.6	40.3	38
	ResNeXt-SVM	22.5	39.8	41
	EESP-ResNeXt	19.2	33.2	46
Test set	MViTDepth	25.1	42.6	44
	Resnet50-SPP	24.8	40.2	44
	SPP-CNN	23.7	41.8	45
	ResNeXt-SVM	21.5	37.8	52
	EESP-ResNeXt	19.4	33.7	57



(a) Training set

(b) Test set

Figure 7: Throughput changes of different models.

Table 4: Error result analysis.

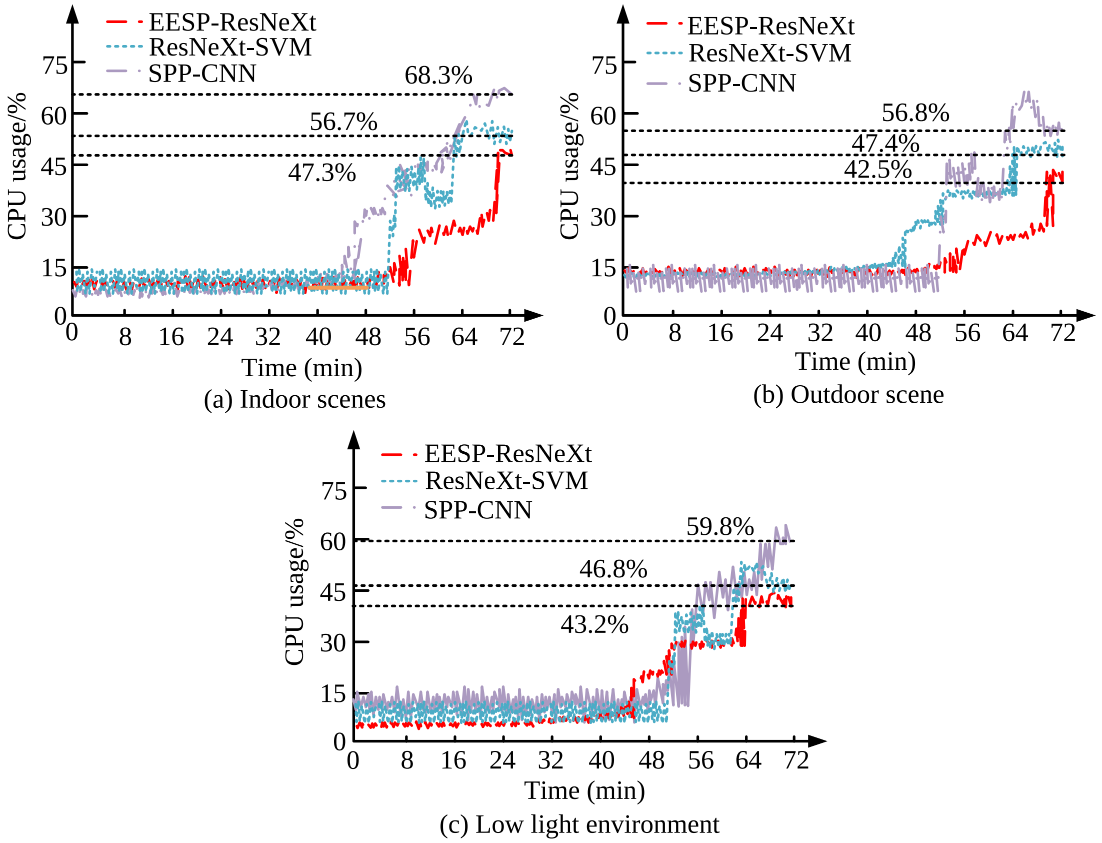
Scene	Model	MSE	MAE	PSNR	SSIM
Indoor	EESP-ResNeXt	0.045	0.035	38.5	0.92
	ResNeXt-SVM	0.055	0.042	35.6	0.89
	SPP-CNN	0.072	0.068	33.2	0.85
Outdoor	EESP-ResNeXt	0.054	0.046	37.9	0.93
	ResNeXt-SVM	0.063	0.045	35.8	0.88
	SPP-CNN	0.084	0.065	32.3	0.83
Low light environment	EESP-ResNeXt	0.038	0.028	39.2	0.93
	ResNeXt-SVM	0.048	0.035	36.4	0.94
	SPP-CNN	0.065	0.055	34.2	0.86

Figures 7 (a) and 7 (b) show the throughput changes of the three models on the training and testing sets, respectively. In Figure 7 (a), the throughput of EESP-ResNeXt consistently remained at a high level, significantly higher than that of ResNeXt-SVM and SPP-CNN. From the changes in the curve, EESP-ResNeXt exhibited a relatively stable throughput during the training process, indicating its high computational efficiency and ability to maintain good training performance. In contrast, ResNeXt-SVM had lower throughput and greater fluctuations, indicating that it might face bottlenecks in certain training stages. The throughput of SPP-CNN was the lowest and fluctuated the most, indicating that its computational efficiency during training was low and there might have been a high computational burden. In Figure 7 (b), the throughput of EESP-ResNeXt remained at the highest level, with overall small fluctuations, demonstrating good stability and efficiency. The throughput of ResNeXt-SVM was slightly lower than that of EESP-ResNeXt, but still significantly higher than that of SPP-CNN. The throughput of SPP-CNN on the test set was still the lowest, further confirming its high computational burden in the inference stage, which affected its performance.

3.2 Application effect analysis of EESP-ResNeXt network model

To verify the application effect of EESP-ResNeXt in VR image depth estimation, three scenarios were simulated: indoor scenes, dynamic scenes, and low light environments, and the application effects of different models were tested in different scenarios. Each test scenario used the same dataset and environment settings, with Mean Squared Error (MSE), Mean Absolute Error (MAE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index (SSIM) as metrics. The test results are shown in Table 4.

In Table 4, the EESP-ResNeXt model performed the best in indoor scenes, with the lowest MSE and MAE of 0.045 and 0.035, respectively. The PSNR was 38.5 and the SSIM was 0.92, demonstrating the accuracy and stability of the model in indoor environments. In contrast, SPP - CNN performed the worst in terms of MSE and MAE, and had the lowest PSNR and SSIM, indicating that the model had significant shortcomings in accuracy and structural similarity. In outdoor scenes, EESP-ResNeXt also performed well, with an MSE of 0.054, an MAE of 0.046, a PSNR of 37.9, and an SSIM of 0.93. The performance of ResNeXt-SVM and SPP-CNN was slightly inferior. Especially, SPP-CNN had poor performance in PSNR and SSIM, which were 32.3 and 0.83 respectively, indicating its poor adaptability to depth changes in outdoor scenes. In low-light environments, EESP-ResNeXt still maintained its lead, with an MSE of 0.038, an MAE of 0.028, a PSNR of 39.2, and an SSIM of 0.93. The PSNR and SSIM of ResNeXt-SVM and SPP-CNN were significantly lower than those of EESP-ResNeXt, especially in terms of MSE and MAE. SPP-CNN performed the worst, indicating its lower depth-estimation accuracy in processing multi-view information. The CPU utilization of different models in different scenarios is shown in Figure 8.



(a) Indoor scenes

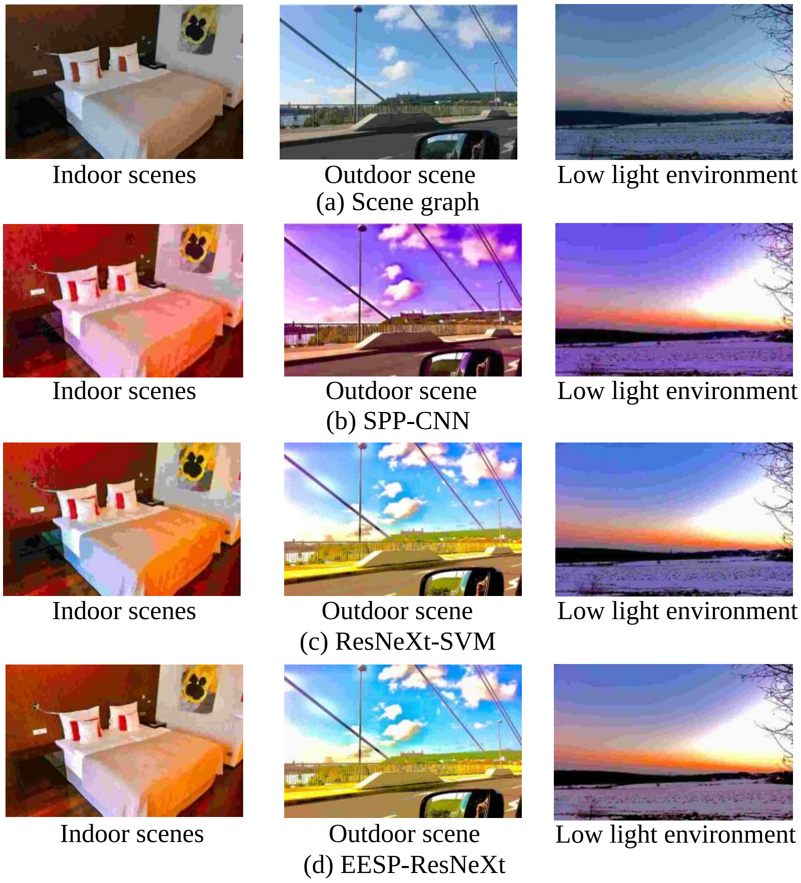
(b) Outdoor scene

(c) Low light environment

Figure 8: CPU utilization in different scenarios.

Figures 8 (a), 8 (b), and 8 (c) show the CPU usage of the three models in three simulation scenarios, respectively. In Figure 8 (a), EESP-ResNeXt had the lowest CPU utilization rate at 47.3%, while SPP-CNN had a CPU utilization rate of 56.7%, and ResNeXt-SVM had the highest CPU utilization rate at 68.3%. This indicated that EESP-ResNeXt could effectively reduce computational overhead and maintain low resource consumption when processing indoor scenes, while still achieving good performance. In Figure 8 (b), the CPU utilization of EESP-ResNeXt was 42.5%, which was the lowest among the three. The CPU utilization rates of SPP-CNN and ResNeXt-SVM were 56.8% and 47.4%, respectively. This indicated that EESP-ResNeXt had lower computational requirements in outdoor scenarios compared to the other two models, and had high computational efficiency, which could efficiently handle depth estimation tasks in outdoor environments. In Figure 8 (c), EESP-ResNeXt had the lowest CPU utilization at 43.2%, while SPP-CNN had a CPU utilization of 59.8% and ResNeXt-SVM had a CPU utilization of 46.8%. This indicated that EESP-ResNeXt maintained low computational resource requirements under low-light conditions and could provide efficient depth estimation in resource-limited environments. To visually compare the depth maps generated by the three models, three images were selected for estimation in indoor, outdoor, and low light environments. The visualization results are shown in Figure 9.

Figure 9 (a) shows three scene diagrams, while Figures 9 (b), 9 (c), and 9 (d) show the depth maps generated by different models, respectively. In Figure 9 (b), SPP-CNN could capture the details of objects well. For outdoor scenes, the depth map of SPP-CNN performed well with strong depth hierarchy, but the clarity was poor. In low light environments, the depth map generated by SPP-CNN maintained relatively clear depth information, but there was relatively less depth information in some shaded areas. In Figure 9 (c), the depth map of ResNeXt-SVM was more accurate than SPP-CNN, and could more accurately represent the depth information of objects. For outdoor scenes, the depth map of ResNeXt-SVM showed a strong sense of hierarchy, with clear object depth differentiation and better performance than SPP-CNN. In low light environments, ResNeXt-SVM performed more stably in generating depth maps, but there were still deviations in some shaded areas. In Figure 9 (d), the depth map generated by EESP-ResNeXt could better capture subtle depth differences in indoor environments, especially in the details and edges of objects. In outdoor scenes, the depth map of EESP-ResNeXt displayed extremely detailed depth variations, which could effectively distinguish between near and far objects. In low light environments, EESP-ResNeXt could maintain high accuracy. Although there might be slight changes in areas with insufficient lighting, it could still maintain the depth relationship between objects well, especially in shadow areas where depth estimation was more accurate.



(a) Scene graph

(b) SPP-CNN

(c) ResNeXt-SVM

(d) EESP-ResNeXt

Figure 9: Visualization results (Picture "Indoor scenes" source from: https://www.1001freedownloads.com/free-photo/bed-hotel-pillow-bedroom; Picture "Outdoor scenes" source from: https://www.1001freedownloads.com/free-photo/perspective-w-rzburg-russian-fortress-bridge; Picture "Low light environment" source from: https://www.1001freedownloads.com/free-photo/oulu-finland-sunset-buildings-river-water).

4 Conclusion

A EESP-ResNeXt network model based on ResNeXt combined with EESP module was proposed to meet the requirements of depth estimation accuracy and efficiency in VR environment. This model effectively improved the depth estimation accuracy of the model in complex environments by introducing multi-scale feature fusion and hybrid AMs. The experimental results showed that EESP-ResNeXt outperformed existing ResNeXt-SVM and SPP-CNN models in indoor, outdoor, and low light environments. The MSE of EESP-ResNeXt in indoor scenes was 0.045, PSNR was 38.5, and SSIM was 0.92, demonstrating its high accuracy and robustness in complex scenes. In low light environments, EESP ResNeXt could still maintain low MSE and high PSNR, with values of 0.050 and 37.9, respectively. This fully validated its robustness under varying lighting conditions. In addition, EESP ResNeXt had a parameter of 19.4M and an inference speed of 57 FPS, demonstrating comparable or higher depth accuracy while maintaining hardware efficiency, making it more suitable for VR real-time applications. Although EESP ResNeXt performed well in multiple scenarios, performance degradation might still occur in extreme lighting, strong reflection, dynamic occlusion, or adverse weather conditions. This type of environment usually had problems such as sparse texture, blurry imaging, and geometric distortion, and the current model was mainly trained in sunny days and conventional indoor and outdoor scenes, with certain limitations on generalization ability. Future research can introduce multi-modal sensor data, such as infrared or event cameras, to enhance the perceptual robustness of the model under harsh conditions. By combining style transfer and domain adaptation techniques, an enhanced training set across weather scenarios is constructed, and an uncertainty modeling mechanism is introduced to further improve the accuracy of depth estimation.

References

[1] Dong Zhao, Jia Li, Hongyu Li, and Long Xu. Stripe sensitive convolution for omnidirectional image dehazing. IEEE Transactions on Visualization and Computer Graphics, 30(7):3516-3531, 2023. https://doi.org/10.1109/TVCG.2022.3233900

[2] Fei Liu, Yunlong Wang, Qing Yang, Shubo Zhou, and Kunbo Zhang. A comprehensive research on light field imaging: Theory and application. IET Computer Vision, 18(8):1269-1284, 2024. https://doi.org/10.1049/cvi2.12321

[3] Giovanni Pintore, Fabio Bettio, Marco Agus, and Enrico Gobbetti. Deep scene synthesis of Atlanta-world interiors from a single omnidirectional image. IEEE Transactions on Visualization and Computer Graphics, 29(11):4708-4718, 2023. https://doi.org/10.1109/TVCG.2023.3320219

[4] Yangang Cai, Xuesong Gao, Weiqiang Chen, and Ronggang Wang. Towards 6DoF live video streaming system for immersive media. Multimedia Tools and Applications, 81(25), 35875-35898, 2022. https://doi.org/10.1007/s11042-021-11589-2

[5] Tao Liu, Zi Jia, Zhengling Lei, Xiaocai Zhang, and Yuchi Huo. Unsupervised depth estimation for ship target based on single view UAV image. International Journal of Remote Sensing, 43(9):3216-3235, 2022. https://doi.org/10.1080/01431161.2022.2088260

[6] Hamam Mokayed, Tee Zhen Quan, Lama Alkhaled, and V. Sivakumar. Real-time human detection and counting system using deep learning computer vision techniques. Artificial Intelligence and Applications, 1(4):221-229, 2023. https://doi.org/10.47852/bonviewAIA2202391

[7] Hongyong Leng, Cheng Chen, Rumeng Si, Chen Chen, Hanwen Qu, and Xiaoyi Lv. Accurate screening of early-stage lung cancer based on improved ResNeXt model combined with serum Raman spectroscopy. Journal of Raman Spectroscopy, 53(7):1302-1311, 2022. https://doi.org/10.1002/jrs.6365

[8] Mohammad Meraj Khan, Mohammad Shorif Uddin, Mohammad Zavid Parvez, and Lutfur Nahar. A squeeze and excitation ResNeXt-based deep learning model for Bangla handwritten compound character recognition. Journal of King Saud University-Computer and Information Sciences, 34(6):3356-3364, 2022. https://doi.org/10.1016/j.jksuci.2021.01.021

[9] Wu-Chih Hu, Liang-Bi Chen, and Hong-Ming Lin. A method for abnormal behavior recognition in aquaculture fields using deep learning une méthode de reconnaissance des comportements anormaux dans l'aquaculture à l'aide de l'apprentissage profond. IEEE Canadian Journal of Electrical and Computer Engineering, 47(3):118-126, 2024. https://doi.org/10.1109/ICJECE.2024.3398653

[10] Yufang Yang, Dashe Li, and Siwei Zhao. A novel approach for underwater fish segmentation in complex scenes based on multi-levels triangular atrous convolution. Aquaculture International, 32(4):5215-5240, 2024. https://doi.org/10.1007/s10499-024-01424-4

[11] Shimin Xiong, Bin Li, Shiao Zhu, Dongfei Cui, and Xiaonan Song. Spatial pyramid pooling and adaptively feature fusion based yolov3 for traffic sign detection. The International Arab Journal of Information Technology, 20(4):592-599, 2023. https://doi.org/10.34028/iajit/20/4/5

[12] Shan Zhao, Yunlei Wang, Xuan Wu, and Fukai Zhang. MAFNet: dual-branch fusion network with multiscale atrous pyramid pooling aggregate contextual features for real-time semantic segmentation. Complex & Intelligent Systems, 10(4):5107-5126, 2024. https://doi.org/10.1007/s40747-024-01428-w

[13] P. Aruna Sri, and V. Santhi. RETRACTED: The reptile optimized deep learning model for land cover classification of the uppal earth region in telangana state using satellite image fusion. Journal of

Intelligent & Fuzzy Systems, 46(2):3209-3229, 2024. https://doi.org/10.3233/JIFS-232891

[14] Hao Dong, Yinlai Du, Dong Feng, Qingyuan Hu, Mingzhu Zhou, Jun Xing, Long Zhang, Shu Wang, and Yong Liu. CSegNet: a hybrid transformer-CNN network for road crack image segmentation. Insight-Non-Destructive Testing and Condition Monitoring, 66(12):737-746, 2024. https://doi.org/10.1784/insi.2024.66.12.737

[15] Saleh Saeed, Sungjun Lee, Yongju Cho, and Unsang Park. ASPPMVSNet: A high-receptive-field multiview stereo network for dense three-dimensional reconstruction. ETRI Journal, 44(6):1034-1046, 2022. https://doi.org/10.4218/etrij.2021-0305

[16] Xin Song, and Baoyun Wang. The segmentation of debris-flow fans based on local features and spatial attention mechanism. Journal of Geographical Sciences, 34(12):2534-2550, 2024. https://doi.org/10.1007/s11442-024-2303-2

[17] Yu Zhang, Zilong Wang, and Yongjian Zhu. 3D point cloud classification method based on multiple attention mechanism and dynamic graph convolution. Information Technology and Control, 52(3):605-616, 2023. https://doi.org/10.5755/j01.itc.52.3.33035

[18] Ning Ma, and Songwen Jin. CCUNet: UNet based on an improved coordinate channel attention mechanism and its applications. Traitement du Signal, 42(1):119-128, 2025. https://doi.org/10.18280/ts.420111

[19] Chengpei Wu, Yang Lou, Lin Wang, Junli Li, Xiang Li, and Guanrong Chen. SPP-CNN: An efficient framework for network robustness prediction. IEEE Transactions on Circuits and Systems I: Regular Papers, 70(10):4067-4079, 2023. https://doi.org/10.1109/TCSI.2023.3296602

[20] Guohui Wang, Hao Zheng, and Xuchen Li. ResNeXt-SVM: a novel strawberry appearance quality identification method based on ResNeXt network and support vector machine. Journal of Food Measurement and Characterization, 17(5):4345-4356, 2023. https://doi.org/10.1007/s11694-023-01959-9

[21] Christine Dewi, and Rung-Ching Chen. Combination of resnet and spatial pyramid pooling for musical instrument identification. Cybernetics and Information Technologies, 22(1):104-116, 2022. https://doi.org/10.2478/cait-2022-0007

[22] Wei Gao, Di Rao, Yang Yang, and Jie Chen. Edge devices friendly self-supervised monocular depth estimation via knowledge distillation. IEEE Robotics and Automation Letters, 8(12):8470-8477, 2023. https://doi.org/10.1109/LRA.2023.3330054