Creativity and Innovation
2025, VOL. 9, NO. 6, 156-160
DOI: 10.47297/wspciWSP2516-252735.20250906

A Hierarchical Pruning Method for Video Super-Resolution Models Based on L1 Norm

Xuejian Liu

Keyi College Of Zhejiang Sci-Tech University, Shaoxing, Zhejiang, 312369, China

ABSTRACT

In recent years, deep learning-based Video Super-Resolution (VSR) models have achieved remarkable success in reconstructing high-resolution videos. However, these models typically suffer from extremely high computational complexity and a massive number of parameters, which severely hinders their practical deployment on resource-constrained devices. To address this issue, this paper proposes a hierarchical pruning method for VSR models[3] based on the L1 norm. By analyzing the structural characteristics of VSR models, the method designs a hierarchical differentiated pruning strategy, applying different pruning ratios to modules such as motion compensation, feature extraction, temporal fusion, and reconstruction. Specifically, conservative pruning is applied to critical motion alignment modules, while progressive pruning is implemented on the highly redundant reconstruction modules. Structural integrity is maintained in core components like the PCD alignment and TSA fusion modules. Experiments based on the EDVR [2]model and validated on the REDS dataset show that the proposed algorithm achieves model parameter compression while keeping the loss in Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) within an acceptable range.

KEYWORDS

Video super-resolution; L1 norm; Model pruning; Hierarchical pruning; Parameter compression; Computational complexity

1 Introduction

Among the numerous VSR models, EDVR stands out for its well-designed PCD (Pyramid, Cascading and Deformable) alignment module and TSA (Temporal and Spatial Attention) fusion module, demonstrating exceptional reconstruction capabilities when handling challenging scenarios like large motions, complex occlusions, and deformations. However, like most advanced deep learning models, EDVR's outstanding performance comes at the cost of exceptionally high computational demands. Its vast number of parameters and complex computational workflows lead to significant memory usage and energy consumption, making direct deployment on mobile or embedded devices with limited computational power, storage space, and battery capacity extremely difficult.

It is noteworthy that current research in the VSR field primarily focuses on improving model reconstruction accuracy and visual quality, while exploration into the critical issue of model compression and acceleration remains relatively preliminary. Compared to the mature model compression techniques in computer vision tasks like image classification and object detection, VSR models pose unique challenges due to their inherent temporal dependencies, complex motion alignment requirements, and pixel-level reconstruction needs. Specifically, key components in VSR models, such as inter-frame alignment modules and temporal fusion networks, possess unique structural characteristics. Directly applying existing pruning or quantization schemes often severely disrupts the model's ability to maintain temporal consistency, leading to a significant drop in reconstruction quality. Therefore, developing compression algorithms specifically tailored for the characteristics of VSR models, which can substantially reduce computational overhead while preserving reconstruction performance, has become a key bottleneck for enabling the practical application of this technology.

This research precisely addresses this significant gap. Taking the representative VSR model EDVR as a starting point, it explores model compression schemes suitable for video super-resolution tasks. Through in-depth analysis of the characteristics and redundancy of various components in VSR models, we propose a structure-aware hierarchical pruning strategy. This strategy aims to accurately identify and remove redundant parameters that do not affect reconstruction quality, thereby achieving an optimal balance between model complexity and performance, and providing a feasible technical path for deploying advanced VSR algorithms in resource-constrained environments.

2 Literature Review

2.1 Development of Video Super-resolution Technology

The core of VSR technology lies in effectively utilizing the rich temporal redundant information between adjacent frames in a video sequence. Early deep learning-based VSR methods, such as VESPCN, were the first to introduce motion estimation and compensation modules into end-to-end network training. Subsequently, TOFlow further integrated motion compensation with video processing tasks. The EDVR model synthesized and developed previous ideas,

proposing a more powerful multi-scale, deformable convolutional PCD alignment module capable of achieving precise alignment in scenes with large motions. Simultaneously, its TSA fusion module employs a spatio-temporal attention mechanism to adaptively emphasize information-rich regions in each frame, achieving high-quality feature fusion and reconstruction, pushing the performance of VSR to new heights.

2.2 Model Compression and Pruning Techniques

Model compression is a key step for the practical deployment of deep learning. In network pruning, methods can be broadly categorized into unstructured pruning and structured pruning. Unstructured pruning removes individual weights or connections in a fine-grained manner, achieving high theoretical compression rates, but the resulting sparse matrix patterns are difficult to accelerate effectively on general-purpose hardware. Structured pruning, on the other hand, removes entire filters, channels, or layers in a coarse-grained manner, directly reducing the size of the computational graph, thus enabling significant acceleration on general-purpose hardware, albeit with lower flexibility and typically lower compression rates compared to unstructured pruning.

Common pruning criteria include those based on weight magnitude (e. g., L1, L2 norm), gradient information, and second-order methods based on the Hessian matrix [8]. Among them, weight pruning based on the L1 norm (i. e., Magnitude-based Pruning) has become one of the most commonly used and effective criteria due to its low computational overhead, simple implementation, and stable performance. Its basic assumption is that the absolute value of a weight is positively correlated with its importance; therefore, weights with absolute values below a certain threshold are deemed "unimportant" and set to zero. However, most existing pruning research focuses on image classification tasks. Research on more complex video restoration tasks, especially for generative models with intricate structures like EDVR, is still insufficient, and a universal pruning method specifically designed for video super-resolution models has not yet emerged.

3 Research Methodology

This chapter elaborates on the proposed hierarchical pruning method, including the analysis of the EDVR model structure, pruning strategy design, and algorithm implementation workflow.

3.1 EDVR Model Structure Analysis

The EDVR model is a classic architecture in the field of video super-resolution. Its core lies in achieving high-quality video reconstruction through the collaboration of multiple modules. The model first extracts features from the input frames through a preprocessing module, then proceeds to the key deformable convolutional alignment module. This module employs a pyramid structure to compute motion offsets in multi-scale feature space, achieving precise non-rigid motion compensation. The quality of this step directly affects the final reconstruction result.

After motion alignment, the Temporal-Spatial Attention (TSA) fusion module comes into play. This module consists of two components: temporal attention and spatial attention, which filter important information from the inter-frame and intra-frame dimensions, respectively. Temporal attention evaluates the relevance of each frame to the reference frame, while spatial attention focuses on texture-rich areas within each frame. This dual attention mechanism ensures the quality of the fused features.

Finally, the deep reconstruction module is responsible for converting the fused features into a high-resolution image. This module consists of dozens of residual blocks, gradually refining features through complex nonlinear transformations, and finally upsampling the resolution through an upsampling layer. This part has the largest number of parameters. While providing powerful representational capacity, it also contains a high degree of redundancy. The structure of the EDVR model is shown in Figure 1.

3.2 Hierarchical Pruning Strategy Design

Based on in-depth analysis of the model structure, we propose a hierarchical differentiated pruning strategy. This strategy allocates different pruning intensities according to the functional importance and parameter redundancy of each module.

For the input and output layers, we adopt a conservative strategy, setting the pruning ratio to 10%. These layers are critical interfaces for model-data interaction and need to maintain high integrity to ensure the quality of feature extraction and final output.

The motion alignment module employs a more detailed protection strategy. The deformable convolution kernels

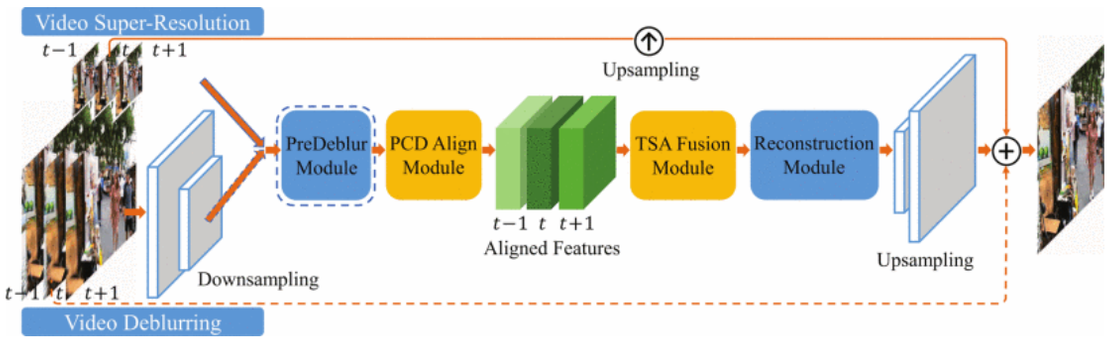

Figure 1 EDVR Model Structure

within it are core components, with the pruning ratio set to the lowest 10%. The auxiliary convolution layers that compute motion offsets are of secondary importance, with a pruning ratio of 15%. This differentiated protection ensures that key alignment functions remain unaffected.

The feature extraction and spatio-temporal attention modules adopt a balanced strategy, with pruning ratios set between 25% and 30%. Although these modules are important, experiments show they can withstand a certain degree of sparsification.

The deep reconstruction module adopts a progressive pruning scheme. The first twenty residual blocks have a pruning ratio of 30%, the middle ten blocks 25%, and the last ten blocks 20%. This rising-then-falling strategy is based on the consideration that deeper features are more closely related to the output quality, requiring more caution closer to the output.

The upsampling layers adopt a conservative strategy, with pruning ratios controlled between 15% and 20%. These layers are responsible for transforming the feature space into the pixel space; excessive pruning may cause artifacts or loss of detail. Additionally, special structural layers like pixel shuffle are completely excluded from pruning operations.

3.3 Algorithm Implementation Workflow

The algorithm is implemented based on the PyTorch framework, using an unstructured pruning method [4]. First, the pre-trained model is loaded, a hierarchical pruning ratio dictionary is initialized, and a mapping relationship between layer names and pruning ratios is established.

In the pruning loop phase, all parameters of the model are traversed to identify weight tensors that need processing. For each convolutional layer, the pruning ratio is determined through name matching. The minimum values are pruned, based on the assumption that parameters with smaller magnitudes have a negligible impact on the output. The most widely used standard is the L1 norm value [1]. The number of weights that need to be set to zero is calculated based on the pruning ratio, and the Quickselect algorithm is used to find the corresponding threshold. Using the Quickselect algorithm efficiently finds the corresponding threshold. This algorithm, based on the partitioning idea of Quicksort, can find the k-th largest element in average O(n) time complexity, significantly reducing computational overhead compared to the O(nlog n) method of full sorting, making it particularly suitable for processing large-scale weight tensors.

A binary mask with the same shape as the original weight is generated, and weights smaller than the threshold are set to zero through element-wise multiplication. This process ensures that each layer is sparsified according to the predetermined ratio while preserving important connections. The algorithm also integrates statistical functions, recording the sparsification status of each layer in real time, including the number of pruned parameters and the actual pruning ratio.

After pruning all layers, the overall compression effect is summarized, and the pruned model state is saved. The entire algorithm is efficient and precise, completing the pruning of all layers through a single forward pass with relatively small computational overhead. The introduction of the Quickselect algorithm further optimizes the threshold calculation step, ensuring that the pruning process remains efficient even in deep networks with massive numbers of parameters. The L1 norm-based pruning criterion accurately identifies weights with smaller contributions, and the hierarchical strategy ensures that the model's key functions are not compromised.

The primary goal of this unstructured pruning method is to reduce parameter storage. Achieving actual inference acceleration requires specialized sparse computation support, which is also an optimization direction for future work. Through this algorithm workflow, we transform the hierarchical pruning strategy into a practical compression tool, providing a technical solution for optimizing the efficiency of video super-resolution models.

4 Experiments and Analysis

4.1 Experimental Setup

Base Models: The experiments employed the EDVR-L and EDVR-M architectures. EDVR-L's reconstruction module contains 40 residual blocks, while EDVR-M's contains 10 residual blocks.

Dataset: The REDS dataset's training set was used for model analysis, and its validation set was used for performance evaluation. REDS contains videos with high dynamic range and large motions, making it an authoritative benchmark for evaluating VSR models.

Evaluation Metrics:

Compression Efficiency: Parameter compression rate, i.e., the ratio of the number of parameters set to zero to the total number of parameters.

Model Performance: Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) calculated on the Y channel to objectively measure the quality of the reconstructed video.

Comparison Methods:

Baseline: The original unpruned EDVR-L and EDVR-M models.

Comparison Method: Global uniform ratio pruning, comparing the PSNR and SSIM values of the pruned models with those of the original unpruned models.

4.2 Results and Discussion

After executing the hierarchical pruning algorithm, the statistical data obtained is shown in Table 1. Table 1 presents the quantitative results of the hierarchical pruning algorithm on the EDVR-L and EDVR-M models, including the pruning rate, PSNR, and SSIM metrics:

Table 1 Comparison of Model Parameters Before and After Pruning

Model	Pruning Rate	PSNR(dB)	SSIM
EDVR-L	0	31.09	0.8800
EDVR-L(prune)	25.57%	30.7538	0.8756
EDVR-M	0	30.05	0.8699
EDVR-M(prune)	20.81%	30.0089	0.8661

Based on the experimental results, we derive the following analysis:

(1) Pruning Effectiveness Analysis: The proposed hierarchical pruning method achieved 25.57% parameter compression on the EDVR-L model, with PSNR dropping only 0.3362 dB and SSIM dropping 0.0044. On the EDVR-M model, it achieved 20.81% parameter compression, with PSNR dropping only 0.0411 dB and SSIM dropping 0.0038. This level of performance loss is typically imperceptible within the range of visual perception, proving that the method effectively reduces model complexity while maintaining model performance.

(2) Visual Result Analysis: Figure 2 shows the pruning statistics for the EDVR-L model, indicating that a total of 132 network layers were processed, 5,270,563 parameters were removed, the final model parameter count was 20,426,256, achieving an overall pruning rate of 25.57%. The visual comparison results in Figure 3 indicate that the reconstructed images after pruning maintain detail clarity and edge sharpness without obvious artifacts or distortion, further verifying the practicality of the method.

(3) Strategy Superiority Analysis: Compared to the global pruning method achieving the same compression rate, our hierarchical strategy achieved higher PSNR and SSIM. This strongly proves the effectiveness of the differentiated pruning strategy. Global pruning, due to its failure to distinguish module importance, caused greater performance loss from over-pruning critical layers. The content-aware versions of PSNR and SSIM can be derived from the variance of the damaged image/video from PSNR without further reference to the original content [5], therefore a comparison of super-resolution results before and after pruning will not be presented here.

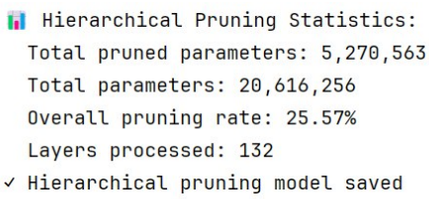

Figure 2 EDVR-L Parameter Pruning Results

```
  state_dict = torch.load('edvr_L_multi_ratio_pruned.pth', map_location='cpu')
2025-11-13 12:41:21,162 INFO: Pruned model weights loaded successfully!
2025-11-13 12:41:21,162 INFO: Testing REDS4...
2025-11-13 12:41:21,162 WARNING: nondist_validation is not implemented. Run dist_validation.
Test 020: 100/100: 100%|████████████████████████████████████████████████████| 400/400 [08:02<00:00,  1.21s/frame]
2025-11-13 12:49:23,472 INFO: Validation REDS4
         # psnr: 30.7358         # 000: 27.7962  # 011: 31.7391  # 015: 33.6360  # 020: 29.7720
            Best: 30.7358 @ EDVR_L_x4_REDS_SR_official iter
         # ssim: 0.8756  # 000: 0.8211   # 011: 0.8816   # 015: 0.9162   # 020: 0.8834
            Best: 0.8756 @ EDVR_L_x4_REDS_SR_official iter
```

Figure 3 EDVR-L Performance Pruning Results

5 Conclusion and Outlook

This paper addressed the significant computational cost of the advanced video super-resolution model EDVR by proposing and implementing an L1 norm-based hierarchical pruning compression method. Through in-depth analysis of the model architecture, we designed a refined pruning ratio configuration scheme, implementing differentiated pruning intensities for different functional modules. Experimental results show that this method can achieve 25.57% parameter compression for the EDVR-L model and 20.81% for the EDVR-M model, while controlling the performance loss to a very low level, and its effectiveness outperforms traditional global pruning methods. This work confirms the necessity and effectiveness of fine-grained pruning tailored to the structural characteristics of the model, providing a practical technical solution for deploying large VSR models in resource-constrained environments.

References

[1] T. Pragnesh and B. R. Mohan, "Comparing Different Sequences of Pruning Algorithms for Hybrid Pruning," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-7.

[2] X. Wang, K. C. K. Chan, K. Yu, C. Dong and C. C. Loy, "EDVR: Video Restoration With Enhanced Deformable Convolutional Networks," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Long Beach, CA, USA, 2019, pp. 1954-1963.

[3] H. Zhao, L. Yu, R. Guan, L. Jia, J. Zhang and Y. Yue, "PLSR: Unstructured Pruning with Layer-Wise Sparsity Ratio," 2023 International Conference on Machine Learning and Applications (ICMLA), Jacksonville, FL, USA, 2023, pp. 1-8.

[4] Wang Xianbao, Liu Pengfei, Xiang Sheng, et al. Unstructured Pruning Method Based on Neural Architecture Search[J]. Pattern Recognition and Artificial Intelligence, 2023, 36(05): 448-458.

[5] M. Martini, "A Simple Relationship Between SSIM and PSNR for DCT-Based Compressed Images and Video: SSIM as Content-Aware PSNR," 2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP), Poitiers, France, 2023, pp. 1-5.