This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

MVQA: Mamba with Unified Sampling for Efficient Video Quality Assessment

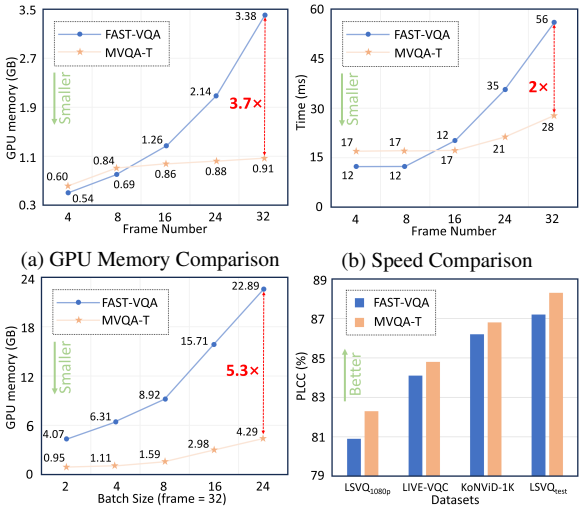
Yachun Mi[1] Yu Li[1] Weicheng Meng[1] Chaofeng Chen
Chen Hui[1,2*] Shaohui Liu[1*]
[1] Harbin Institute of Technology
[2] Nanjing University of Information Science and Technology

Abstract

The rapid growth of long-duration, high-definition videos has made efficient video quality assessment (VQA) a critical challenge. Existing research typically tackles this problem through two main strategies: reducing model parameters and resampling inputs. However, light-weight Convolution Neural Networks (CNN) and Transformers often struggle to balance efficiency with high performance due to the requirement of long-range modeling capabilities. Recently, the state space model, particularly Mamba, has emerged as a promising alternative, offering linear complexity with respect to sequence length. Meanwhile, efficient VQA heavily depends on resampling long sequences to minimize computational costs, yet current resampling methods are often weak in preserving essential semantic information. In this work, we present MVQA, a Mamba-based model designed for efficient VQA along with a novel Unified Semantic and Distortion Sampling (USDS) approach. USDS combines semantic patch sampling from low-resolution videos and distortion patch sampling from original-resolution videos. The former captures semantically dense regions, while the latter retains critical distortion details. To prevent computation increase from dual inputs, we propose a fusion mechanism using pre-defined masks, enabling a unified sampling strategy that captures both semantic and quality information without additional computational burden. Experiments show that the proposed MVQA, equipped with USDS, achieve comparable performance to state-of-the-art methods while being $2\times$ as fast and requiring only $1/5$ GPU memory. Code is available at https://github.com/xiao-mi-d/MVQA

1. Introduction

With the widespread adoption of high-definition portable recording devices and the ongoing advancements in video compression [9, 10, 25, 26] and network technologies, ordinary users can now effortlessly capture long-duration, high-resolution videos (e.g., 2K, 4K) and upload them to online platforms. The increasing length and size of video present substantial challenges for video quality assessment (VQA) algorithms. Thus, developing efficient methods to quickly evaluate massive video data is a pressing challenge.



(a) GPU Memory Comparison (b) Speed Comparison
(c) GPU Memory Comparison (d) Accuracy Comparison

Figure 1. Performance and efficiency comparisons between FAST-VQA[74] and our MVQA-tiny. Our method achieves comparable results while running up to $2\times$ faster than the current most efficient FAST-VQA, and reducing GPU memory usage by $5.3\times$. This advantage further scales with increased video frames and batch size.

Traditional VQA methods [1, 5, 7, 34, 42, 55, 60, 68] primarily rely on handcrafted features. While these approaches are computationally efficient and have low complexity, they struggle to effectively capture the intricate, high-dimensional characteristics of videos. As a result, their performance tends to be limited when processing high-resolution videos with rich details and diverse content. In contrast, deep learning-based VQA methods [2, 37, 38, 44, 64, 74–78, 84, 87, 90, 92, 93], particularly Convolutional Neural Networks (CNNs) [18, 19, 21, 22, 61, 65–67] and Transformers [6, 47, 48], have demonstrated substantial advantages in handling complex video tasks. However,

*Corresponding author.

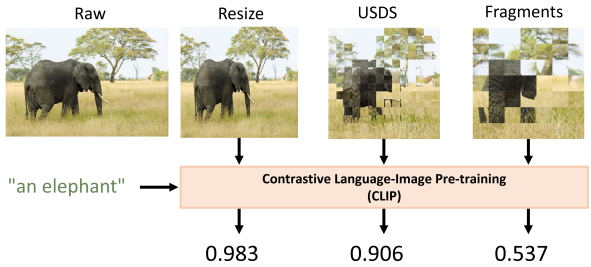


Figure 2. Comparison of semantic information retained by USDS with different samples (resize, Fragments [74]).

deep learning-based approaches typically incur high computational costs, including significant memory consumption and lengthy processing times, which become particularly challenging when dealing with large-scale, high-resolution video data. Specifically, although CNNs excel at extracting local features, they face limitations in modeling long-range dependencies and processing large-scale datasets. On the other hand, while Transformer models excel at capturing global dependencies, their computational complexity grows quadratically with video length and resolution, posing considerable challenges when applied to large video datasets. Consequently, relying on either CNNs or Transformers for VQA often fails to strike an effective balance between performance and computational efficiency.

Mamba [14], a novel sequence modeling architecture, has gained attention for addressing computational bottlenecks in processing long sequences. As a State Space Model (SSM) [8, 15, 16, 50, 63, 70], Mamba models long-range dependencies with state transfer equations and linear complexity relative to input length. Beyond temporal modeling, Mamba also excels in spatio-temporal tasks [11, 17, 39, 58, 89, 94], making it particularly advantageous for VQA. Traditional VQA models often struggle with redundant video data and high computational demands, but Mamba's efficiency allows it to handle large-scale video data accurately and with reduced processing time, especially for long and high-resolution videos. Thus, Mamba-based VQA offers a practical solution for efficient and precise video quality assessment.

Moreover, the computational complexity and accuracy of VQA models are influenced not only by the model architecture but also by the sampling strategy employed. Traditional sampling methods like cropping [29, 30] and resizing [32] reduce computational load but often distort video quality. To better retain quality information, Wu et al. [74] introduced Grid Mini-patch Sampling (GMS), which samples small patches across uniform grids to preserve quality and temporal features. However, the fragmentation from grid sampling limits semantic information capture, which is essential for VQA accuracy [13, 38, 51, 52, 72, 76]. Thus, an effective sampling strategy should maintain both video quality and semantic information.

Based on the above analysis, we propose **U**nified **S**emantic and **D**istortion **S**ampling **(USDS)**, which not only effectively samples video distortions but also preserves the semantic information of the video. As shown in Fig. 2, we use CLIP [59], with its strong zero-shot prediction capabilities, to verify that USDS effectively preserves semantic information. Specifically, USDS first obtains a downsampled version to capture the semantic information of the video, then performs grid sampling to capture the distortion information. These two sampled maps are then combined at a specified ratio through masking. Additionally, we introduce MVQA, a VQA model fully based on state space models, and experiments demonstrate that our model achieves a balance between performance and efficiency. As shown in Fig. 1, Compared to the current most efficient model, FastVQA, our MVQA-tiny achieves comparable performance while offering faster computation speed and lower memory usage.

Our contributions can be summarized as follows:

- This is the first work to successfully apply state space models in the VQA domain, which effectively balances computational efficiency and performance.
- We propose a novel sampling method, USDS, which unifies semantic and distortion patch sampling through masked fusion. This approach addresses the limitation of previous sampling methods, which failed to simultaneously account for both semantic content and distortions.
- Extensive experiments show MVQA outperforms transformer- and CNN-based methods, with efficient computation and reduced memory usage.

2. Related Work

2.1. VQA Methods

Current VQA methods are mainly divided into two categories: traditional and deep learning-based methods.

Traditional VQA methods [34, 55, 60, 68, 83] tend to be knowledge-driven, which evaluate the video quality by extracting hand-crafted features from spatial and temporal domains. Earlier VQA [60, 83] often extracts spatial features of video frames with the help of IQA algorithms [20, 36, 53, 54, 85, 86] and then predicts the video quality by feature aggregation. In addition, VQA through natural video statistics (NVS) is also a classic class of VQA methods [34, 55, 60, 68]. However, the performance is limited because hand-crafted features often fail to extract complex features from videos.

Deep learning-based VQA methods are primarily data-driven. Recently, numerous subjective experiments have produced large, high-quality VQA datasets [23, 41, 57, 62, 71, 87], advancing deep VQA performance [2, 37, 38, 44, 64, 74–78, 84, 87, 90, 92, 93]. Current VQA methods use either CNNs [18, 19, 21, 22, 61, 65–67] or Transformer ar-

chitectures [6, 47, 48]. For example, GST-VQA [2] and VSFA [38] employ CNNs like VGG [61] and ResNet-50 [21] with GRU [4] for temporal modeling, while other studies [44, 64, 72, 87, 88, 90] use 3D-CNNs for spatiotemporal features. With the success of ViTs [6], Transformer-based VQA [74–76] has gained prominence. Methods like FAST-VQA [74] and FasterVQA [75] sample spatial-temporal grids and utilize modified Video Swin Transformers [48]. However, these fragment sampling strategies often neglect semantic content. Recent studies [52, 79, 80, 92] address this by adding branches for semantic extraction, such as the aesthetic branch in DOVER [80], a CNN branch in Zoom-VQA [92], and a CLIP-based branch in CLiF-VQA [52] for aligning semantic features with linguistic prompts.

2.2. State Space Models

In recent years, state space models (SSMs) [15, 16, 63] have emerged as one of the primary contenders to CNN and Transformer architectures within the field of deep learning. Originally derived from classical control theory [28], SSMs have attracted a great deal of interest from researchers due to their great potential for remote dependency modeling and the linear growth of model complexity with respect to sequence length. Structured state space models (S4) [15] introduce a diagonal structure to SSMs and combine it with the diagonal plus low rank method, which improves the computational efficiency of SSMs. Building on this, a series of studies have enhanced the S4 model to further improve its effectiveness and efficiency. For example, S5 [63] further introduces MIMO SSM and efficient parallel scanning in S4. [8] presents a new SSM layer, H3, that greatly improves the effectiveness of SSM for language modeling. GSS [50] builds a gated state space layer by introducing gating mechanisms in S4 to enhance its modeling capabilities. Recently, Mamba [14] has refined the S4 model, a data-dependent State Space Model (SSM) that incorporates efficient hardware design and a selectivity mechanism. This enhancement allows it to outperform Transformer-based models in natural language processing tasks, while also scaling linearly in complexity with input length. Inspired by the success of mamba, a large amount of work has successfully applied mamba to a variety of vision tasks, including image understanding [24, 46, 94], video understanding [3, 39, 58], segmentation [43, 49], restoration [12, 17, 81], and others [27, 40, 56, 91, 95]. All the studies show that mamba has better performance and higher computational efficiency than CNN and Transformer models in vision tasks.

3. Approach

3.1. Preliminaries

State Space Models. State Space Models (SSMs) are inspired by the continuous linear time-invariant (LTI) systems, which maps an input sequence $x(t) \in \mathbb{R}^L$ to a latent spatial representation $h(t) \in \mathbb{R}^N$ and then predicts an output sequence $y(t) \in \mathbb{R}^L$ based on that representation. Formally, SSMs can be formulated as a linear ordinary differential equation (ODE) as follows:

$$h^{'}(t) = Ah(t) + Bx(t) \tag{1}$$
$$y(t) = Ch(t) + Dx(t) \tag{2}$$

where N is the dimension of the hidden state, $h(t) \in \mathbb{R}^N$ is a hidden state, $A \in \mathbb{R}^{N\times N}$ is the evolution parameter, $B \in \mathbb{R}^N$ and $C \in \mathbb{R}^N$ are the projection parameters, $D \in \mathbb{R}^1$ represents the skip connection.

Discretization. To process discrete inputs, the Zero-Order Holding (ZOH) rule is often utilized, which uses the sample timescale parameters $\triangle$ to convert the continuous parameters A and B in the ODE to discrete parameters $\overline{A}$ and $\overline{B}$:

$$\overline{A} = exp(\triangle A) \tag{3}$$
$$\overline{B} = (\triangle A)^{-1}(exp(A) - I) \cdot \triangle B \tag{4}$$

where $\overline{A} \in \mathbb{R}^{N\times N}$, $\overline{B} \in \mathbb{R}^N$. This results in the following discretized OED formula:

$$h_k = \overline{A}h_{k-1} + \overline{B}x_k \tag{5}$$
$$y_k = Ch_k + Dx_k \tag{6}$$

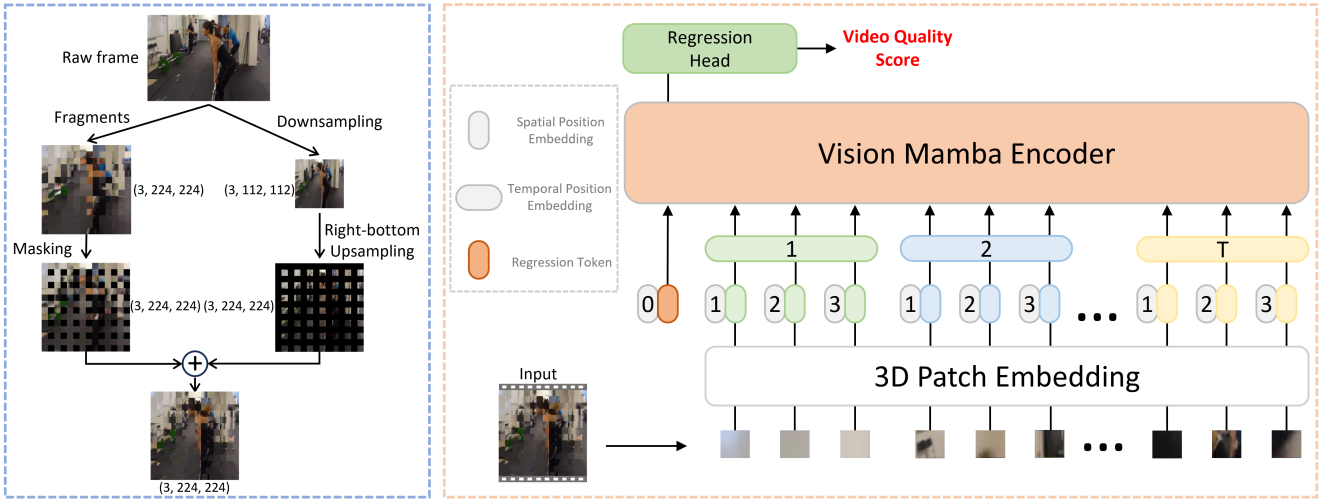
Then in order to increase the computational speed, global convolutional operations with the advantage of parallel computing are utilized to accelerate the above computational process:

$$\overline{K} \triangleq (C\overline{B}, C\overline{AB}, \cdots, C\overline{A}^{L-1}\overline{B}) \tag{7}$$
$$y = x \circledast \overline{K} \tag{8}$$

where L is the length of the input sequence, $\circledast$ denotes convolution operation, and $\overline{K} \in \mathbb{R}^L$ is a structured convolutional kernel.

3.2. Overall Architecture

The architecture proposed in this paper consists of two main components: the unified semantic and distortion sampling (USDS) and the MVQA model design (Fig. 3). Previous studies [33, 45, 74–76] have shown that effective sampling not only reduces model complexity but also improves performance. However, prior methods often lose critical semantic information due to fragmentation in sampled segments, affecting quality prediction. To solve this, we introduce USDS (Fig. 3a), which samples video at its original resolution for distortion and downsamples for semantic information. These are fused through mask fusion to form the final sampled video. The MVQA model then uses 3D embedding on USDS results (Fig. 3b), adding a regression



(a) Unified Semantic and Distortion Sampling.

(b) The network architecture of our MVQA.

Figure 3. The framework of our proposed method. (a) The USDS consists of three distinct but interrelated stages: distortion details extraction, semantic information retention, and fusion of all resolutions. (b) MVQA transforms the input image blocks into one-dimensional vectors by 3D embedding and adds spatial location embedding and temporal location embedding to them, then extracts the features of the video using a bidirectional vision mamba encoder [94], and finally predicts the quality scores of the video by regressing the header.

token for quality prediction. Spatial and temporal embeddings are successively added, and the vision Mamba encoder extracts features for final quality scoring based on the regression token.

3.3. Unified Semantic and Distortion Sampling

Considering the significant computational burden associated with model learning, we propose an approach that integrates high-resolution visual details with low-resolution semantic information to enhance the representation of video frames, which is called **U**nified **S**emantic and **D**istortion **S**ampling **(USDS)**, as shown in Fig. 3a. And the detailed USDS process is shown in Algorithm 1.

Distortion Details and Semantic Information Extraction. In the first stage, input video frames $V \in \mathbb{R}^{C\times T\times H\times W}$ are partitioned into high-resolution spatial fragments. The target resolution, denoted as $S_h = fragments_h \times fsize_h$ and $S_w = fragments_w \times fsize_w$ (e.g., $14 \times 16 = 224$ for both), is achieved by dividing the frame into a grid of $fragments_h \times fragments_w$ patches, each of size $fsize_h \times fsize_w$. These fragments are sampled with random spatial offsets within each grid cell, preserving local details such as textures and edges crucial for high-fidelity reconstruction. The grid starting points are determined by evenly dividing the height and width, adjusted to avoid exceeding the frame boundaries.

For semantic information retention, each frame is downsampled to a lower resolution, typically $S_h/2 \times S_w/2$ (e.g., 112×112 when $S_h = S_w = 224$), using bilinear interpolation:

$$V_{low}[:,t,:,:] = Resize(V[:,t,:,:], (S_h/2, S_w/2)) \quad (9)$$

This process captures global contextual features, such as object boundaries and scene structure, essential for understanding the broader video context.

Fusion of Resolutions. In the second stage, the high-resolution fragments and low-resolution semantic content are fused within a multi-resolution framework. Each frame is divided into blocks of size $2 \cdot fsize_h \times 2 \cdot fsize_w$ (e.g., 32×32 when $fsize_h = fsize_w = 16$), forming a grid of $fragments_h/2 \times fragments_w/2$ blocks. The high-resolution fragments are arranged to fill an intermediate frame $\tilde{V}[:,t,:,:]$ according to their grid positions.

The low-resolution content V_{low} is expanded into a frame $E[:,t,:,:] \in \mathbb{R}^{C\times S_h\times S_w}$ by placing each $fsize_h \times fsize_w$ patch into the bottom-right quadrant of the corresponding $2 \cdot fsize_h \times 2 \cdot fsize_w$ block:

$$E[:,t,R_{k,l}] = V_{low}[:,t,k' : k' + fsize_h, l' : l' + fsize_w] \quad (10)$$

where $k' = 0, 1, \ldots, fragments_h/2 - 1$, and $l' = 0, 1, \ldots, fragments_w/2 - 1$, and $R_{k,l}$ is the bottom-right $fsize_h \times fsize_w$ region of the block with top-left corner at $(k' \cdot 2 \cdot fsize_h, l' \cdot 2 \cdot fsize_w)$.

A binary mask $M \in \mathbb{R}^{1\times 1\times S_h\times S_w}$ is defined to select the bottom-right $fsize_h \times fsize_w$ region of each block. The fusion combines E and $\tilde{V}$ for each frame t:

$$\hat{V}[:,t,:,:] = E[:,t,:,:] \cdot M + \tilde{V}[:,t,:,:] \cdot (1 - M) \quad (11)$$

where $\hat{V} \in \mathbb{R}^{C\times T\times S_h\times S_w}$ is the final output, blending high-resolution details in three quadrants with low-resolution semantic information in the bottom-right quadrant of each block.

The fused fragments are reassembled into complete video frames, producing the output $\hat{V}$. Fig. 4 illustrates

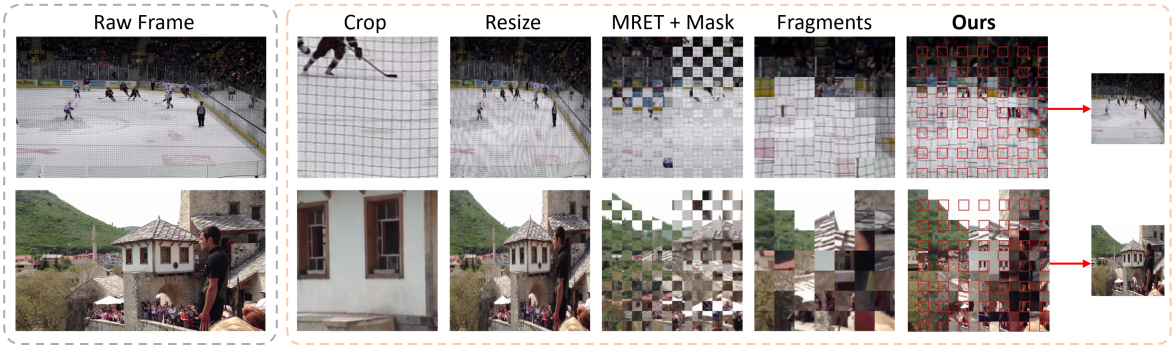


Figure 4. Comparison of USDS with crop, resize, MRET [33], Fragments [74] sampling.

Algorithm 1 Unified Semantic and Distortion Sampling

Require: Video tensor $V \in \mathbb{R}^{C\times T\times H\times W}$, fragments (f_h, f_w), size (s_h, s_w), alignment a

Compute $H' = f_h s_h$, $W' = f_w s_w$

$\tilde{V} = \text{Fragments}(V, f_h, f_w, s_h, s_w, a)$

Create mask $M \in \{0,1\}^{1\times 1\times H'\times W'}$:

$$M_{x,y} = \begin{cases} 1, & (x \bmod 2s_h) \geq s_h \wedge (y \bmod 2s_w) \geq s_w \\ 0, & \text{otherwise} \end{cases}$$

for each frame $t \in [0, T)$ **do**

  Initialize canvas: $E \leftarrow \mathbf{0}^{C\times H'\times W'}$

  $V_{\text{low}} \leftarrow \text{Resize}(V[:,t], \frac{H'}{2} \times \frac{W'}{2})$

  for each (i,j) in $\left(\frac{H'}{2s_h}\right) \times \left(\frac{W'}{2s_w}\right)$ grid **do**

    Copy $E[:, 2is_h{+}s_h : 2(i{+}1)s_h, 2js_w{+}s_w : 2(j{+}1)s_w] \leftarrow V_{\text{low}}[:, is_h : (i+1)s_h, js_w : (j+1)s_w]$

  end for

  $\hat{V}[:,t] \leftarrow E \odot M + \tilde{V}[:,t] \odot (1 - M)$

end for

return $\hat{V}$

a comparison between USDS and other sampling methods. By preserving fine-grained textures in high-resolution patches while integrating global semantic context from low-resolution data, USDS achieves a balanced representation that enhances both visual quality and semantic consistency.

3.4. MVQA Model

Fig. 3b shows the structure of the proposed MVQA. Since the mask fusion in our sampling operation is performed in terms of sample block size, we chunk the input strictly according to the sample size in order to better preserve the quality information and semantic information of the original video carried in the sampled video. Since chunking the input destroys the spatio-temporal information of the video, in order to better model the intra-frame and inter-frame information of the input video, we follow [39] to add spatial position embedding for sequences within the input frames and temporal position embedding for sequences between frames. Specifically, each input block of samples is processed using 3D convolution, where the size of the convolution kernel is $1 \times 16 \times 16$. For an input video of size $X^I \in \mathbb{R}^{3\times T\times H\times W}$, features of size $X^o \in \mathbb{R}^{L\times C}$ are obtained after convolution, where $L = T \times \frac{H}{16} \times \frac{W}{16}$. Then a regression token for final quality regression and positional embedding are added to the token sequence :

$$X = [X_{reg}, X] + p_s + p_t \tag{12}$$

where X_{reg} is a learnable token that is prepended to the start of the sequence, $p_s \in \mathbb{R}^{(\frac{H}{16}\times\frac{W}{16}+1)\times C}$ is a learnable spatial position embedding, $p_t \in \mathbb{R}^{t\times C}$ is a learnable temporal position embedding.

Then, we feed the sequence of tokens X into the vision mamba encoder to pass:

$$T_l = Vim(T_{l-1} + T_{l-1}) \tag{13}$$

where Vim represents the vision mamba block [94] and l represents the current number of layers.

Finally, we normalize the out put regression token X_L^0 and feed it to the multi-layer perceptron (MLP) head to obtain the final prediction of the video quality score Q as follows:

$$f = Norm(X_L^0), Q = MLP(f) \tag{14}$$

where L denotes the number of layers of vision mamba block.

Vision Mamba [94] introduces a bidirectional scanning mechanism that processes tokens forward and backward, enhancing Mamba's spatial modeling and yielding strong results on image tasks. However, video data requires both spatial and temporal modeling. Li et al. [39] found that a spatial-first, temporal-next scanning approach is most effective for video tasks, so MVQA adopts this mechanism, as shown in Fig. 5.

3.5. Loss Function

Several studies have shown that both monotonicity-induced loss and linearity-induced loss play a very important role in the quality prediction task, so we combine both as our loss function. Given the predicted quality score $\hat{Q}$ =

Table 1. Experimental performance of the pre-trained MVQA model on the LSVQ dataset on four test sets (LSVQ_{test}, LSVQ_{1080p}, KoNViD-1k, LIVE-VQC). Existing best multi-branch in **green** and existing best single-branch in **blue**.

Testing Type			Intra-dataset Test Datasets				Cross-dataset Test Datasets			
Testing Datasets			**LSVQ_{test}**		**LSVQ_{1080p}**		**KoNViD-1k**		**LIVE-VQC**	
Type	Methods	Source	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Classical	BRISQUE [53]	*TIP, 2012*	0.569	0.576	0.497	0.531	0.646	0.647	0.524	0.536
	TLVQM [34]	*TIP, 2019*	0.772	0.774	0.589	0.616	0.732	0.724	0.670	0.691
	VIDEVAL [68]	*TIP, 2021*	0.794	0.783	0.545	0.554	0.751	0.741	0.630	0.640
Deep Multi-branch	$\text{PVQ}_{wo/patch}$ [87]	*CVPR, 2021*	0.814	0.816	0.686	0.708	0.781	0.781	0.747	0.776
	$\text{PVQ}_{w/patch}$ [87]	*CVPR, 2021*	0.827	0.828	0.711	0.739	0.791	0.795	0.770	0.807
	BVQA [37]	*TCSVT, 2022*	0.852	0.854	0.771	0.782	0.834	0.837	0.816	0.824
	DOVER [80]	*ICCV, 2023*	0.881	0.879	0.782	0.827	0.871	0.872	0.812	0.841
	Zoom-VQA [92]	*CVPR,2023*	0.886	0.879	0.799	0.819	0.877	0.875	0.814	0.833
	Q-Align [82]	*ICML, 2024*	0.883	0.882	0.797	0.830	0.865	0.877	NA	NA
	CLiF-VQA [52]	*ACMMM,2024*	0.886	0.887	0.790	0.832	0.877	0.874	**0.834**	**0.855**
	MBVQA [73]	*CVPR, 2024*	**0.895**	**0.895**	**0.809**	**0.844**	**0.878**	**0.884**	0.806	0.844
Deep Single-branch	VSFA [38]	*ACMMM, 2019*	0.801	0.796	0.675	0.704	0.784	0.794	0.734	0.772
	FAST-VQA-M [74]	*ECCV, 2022*	0.852	0.854	0.739	0.773	0.841	0.832	0.788	0.810
	FAST-VQA [74]	*ECCV, 2022*	0.872	**0.874**	0.770	0.809	**0.864**	0.862	**0.824**	**0.841**
	FasterVQA [75]	*TPAMI, 2023*	**0.873**	**0.874**	**0.772**	**0.811**	0.863	**0.863**	0.813	0.837
	MVQA-tiny	***ours***	**0.882**	**0.883**	**0.781**	**0.823**	**0.870**	**0.868**	**0.828**	**0.848**
	MVQA-middle	***ours***	**0.898**	**0.899**	**0.812**	**0.846**	**0.885**	**0.887**	**0.852**	**0.873**
improvement to existing best Single-branch			2.86%	2.86%	5.18%	4.32%	2.43%	2.78%	3.40%	3.80%
improvement to existing best Multi-branch			0.34%	0.45%	0.37%	0.24%	0.80%	0.34%	2.16%	2.11%

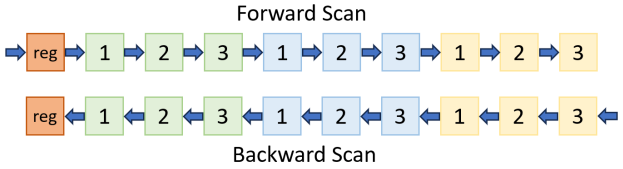


Figure 5. Scan method.

$\{\hat{q_1}, \hat{q_2}, ..., \hat{q_m}\}$ and ground-truth subjective quality score $Q = \{q_1, q_2, ..., q_m\}$.

The monotonicity-induced loss L_{mon} and the linearity-induced loss L_{lin} are defined as follows:

$$L_{mon} = \frac{1}{m^2} \sum_{i=1}^{m} \sum_{j=1}^{m} max(0, |q_i - q_j| - f(q_i, q_j) \cdot (\hat{q_i} - \hat{q_j})) \tag{15}$$

where $f(q_i, q_j) = 1$ if $q_i \geq q_j$, otherwise $f(q_i, q_j) = -1$.

$$L_{lin} = (1 - \frac{\sum_{i=1}^{m} (\hat{q}_i - \hat{a})(q_i - a)}{\sqrt{\sum_{i=1}^{m} (\hat{q}_i - \hat{a})^2 \sum_{i=1}^{m} (q_i - a)^2}}) / 2 \tag{16}$$

where $a = \frac{1}{m} \sum_{i=1}^{m} q_i$ and $\hat{a} = \frac{1}{m} \sum_{i=1}^{m} \hat{q}_i$.

The total loss function L is obtained by combining the two loss functions L_{mon} and L_{lin} above:

$$L = \alpha L_{mon} + \beta L_{lin} \tag{17}$$

where α and β represent the weights of monotonicity-induced loss and linearity-induced loss.

4. Experiments

4.1. Experimental Setups

Datasets. We test the model on four datasets including LSVQ [87], KoNViD-1k (1200 videos) [23], LIVE-VQC (585 videos) [62], and YouTube-UGC (1067 videos) [71]. Specifically, we pre-train MVQA on LSVQ_{train}, a subset of LSVQ containing 28,056 videos. Intra-dataset testing is performed on two subsets of LSVQ, LSVQ_{test} (7400 videos) and LSVQ_{1080p} (3600 videos). We perform cross-dataset testing on KoNViD-1k and LIVE-VQC. Further, we fine-tune the model on KoNViD-1k, LIVE-VQC, and YouTube-UGC. It should be noted that YouTube-UGC contains 1500 videos, but only 1067 videos are available to us.

Table 2. Details of MVQA with different sizes.

Models	Input Size	Depth	Dim	Param	Flops
MVQA-tiny	$32 \times 224 \times 224$	24	192	7M	34G
MVQA-middle	$32 \times 224 \times 224$	32	576	74M	403G

Evaluation Criteria. Spearman Rank Order Correlation Coefficient (SROCC) and Pearson Linear Correlation Coefficient (PLCC) are used as evaluation Metrics. Specifically, SRCC is used to measure the prediction monotonicity between predicted scores and true scores by ranking the values in both series and calculating the linear correlation between the two ranked series. In contrast, PLCC evaluates prediction accuracy by calculating the linear correlation between a series of predicted scores and true scores. And higher SROCC and PLCC scores indicate better performance.

Implementation Details. We employ PyTorch framework and an NVIDIA H100 card to train the model. We pre-train the MVQA backbone network on the Kinetics-400 [31] dataset. We set the initial learning rate to 0.0025, the optimizer to AdamW, and use a cosine annealing strategy to dynamically adjust the learning rate. The detailed information of our proposed MVQA with two different scales is shown in Tab. 2. For MVQA-tiny and MVQA-middle, we set the batch size to 32 and 8, and the weight decay during training to 0.1 and 0.05.

Table 3. The finetune results on LIVE-VQC, KoNViD and YouTube-UGC. Existing best multi-branch in **green** and existing best single-branch in **blue**.

Finetune Datasets			**LIVE-VQC**		**KoNViD-1k**		**YouTube-UGC**		***Average***	
Type	Methods	Source	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Classical	TLVQM [34]	*TIP, 2019*	0.799	0.803	0.773	0.768	0.669	0.659	0.732	0.726
	VIDEVAL [68]	*TIP, 2021*	0.752	0.751	0.783	0.780	0.779	0.773	0.772	0.772
	RAPIQUE [69]	*OJSP, 2021*	0.755	0.786	0.803	0.817	0.759	0.768	0.774	0.790
Classical + Deep	CNN+TLVQM [35]	*ACMMM, 2020*	0.825	0.834	0.816	0.818	0.809	0.802	0.815	0.814
	CNN+VIDEVAL [68]	*TIP, 2021*	0.785	0.810	0.815	0.817	0.808	0.803	0.806	0.810
Deep Multi-branch	PVQ [87]	*CVPR, 2021*	0.827	0.837	0.791	0.786	NA	NA	NA	NA
	BVQA [37]	*TCSVT, 2022*	0.831	0.842	0.834	0.836	0.831	0.819	0.832	0.832
	CoINVQ [72]	*TCSVT, 2021*	NA	NA	0.767	0.764	0.816	0.802	NA	NA
	DOVER [80]	*ICCV, 2023*	0.812	0.852	0.897	0.899	0.877	0.873	0.862	0.875
	MaxVQA [79]	*ACMMM,2023*	0.854	0.873	0.894	0.895	**0.894**	**0.890**	0.881	0.886
	CLiF-VQA [52]	*ACMMM,2024*	**0.866**	0.878	**0.903**	0.903	0.888	**0.890**	**0.886**	**0.890**
	MBVQA [73]	*CVPR, 2024*	0.860	**0.880**	0.901	**0.905**	0.876	0.877	0.879	0.887
Deep Single-branch	VSFA [38]	*ACMMM, 2019*	0.773	0.795	0.773	0.775	0.724	0.743	0.752	0.765
	GST-VQA [2]	*TCSVT, 2021*	NA	NA	0.814	0.825	NA	NA	NA	NA
	FAST-VQA-M [74]	*ECCV, 2022*	0.803	0.828	0.873	0.872	0.768	0.765	0.815	0.822
	FAST-VQA [74]	*ECCV, 2022*	**0.845**	0.852	0.890	0.889	0.857	0.853	0.864	0.865
	FasterVQA [75]	*TPAMI, 2023*	0.843	**0.858**	**0.895**	**0.898**	**0.863**	**0.859**	**0.867**	**0.872**
	MVQA-tiny	***ours***	**0.850**	**0.866**	**0.903**	**0.904**	**0.869**	**0.872**	**0.874**	**0.881**
	MVQA-middle	***ours***	**0.878**	**0.895**	**0.925**	**0.925**	**0.901**	**0.903**	**0.901**	**0.908**
improvement to existing best Single-branch			3.91%	4.31%	3.35%	3.01%	4.40%	5.12%	3.92%	4.13%
improvement to existing best Multi-branch			1.39%	1.71%	2.44%	2.21%	0.78%	1.46%	1.69%	2.02%

4.2. Experiment Results

Pre-training Results on LSVQ. We pre-train the proposed MVQA on LSVQ and conduct intra-dataset testing on LSVQ_{test} and LSVQ_{1080p}. Additionally, cross-dataset testing was performed on KoNViD-1k and LIVE-VQC. The results are shown in Tab. 1. First, for our tiny version of the MVQA model, compared to the current best-performing and fastest FasterVQA model, our model achieves comparable performance while offering faster processing speed. In addition, for the middle version of MVQA, compared with the classical methods (BRISQUE, TLVQM, VIDEVAL), our method shows superior performance on all test sets. Our model also achieves better results compared to deep learning-based methods. Specifically, compared to the current state-of-the-art single-branch models, our model achieve average improvements of 4.02% and 3.59% in SROCC and PLCC metrics for intra-dataset testing. In cross-dataset testing, the improvements are 2.92% and 3.29%, respectively. Since single-branch models often struggle to simultaneously capture both quality and semantic features of videos, recent multi-branch models, which better incorporate video semantic information, have shown outstanding performance. The performance of our single-branch based MVQA model is still ahead of all current multi-branch models. Compared to the current state-of-the-art results, our model achieves improvements of 0.92% and 0.79% in SROCC and PLCC, respectively, while offering higher computational efficiency.

Fine-tuning Results on Small Datasets. We fine-tune MVQA on three small datasets (LIVE-VQC, KoNViD-1k, YouTube-UGC), as shown in Tab. 3. We can see that MVQA-tiny achieves comparable performance compared to the current best single-branch model. Moreover, MVQA-middle achieves improvements of 3.92% and 4.13% in SROCC and PLCC, respectively, on three datasets compared to the best results from single-branch models. Additionally, compared to the best multi-branch models, it shows improvements of 1.69% and 2.02% in SROCC and PLCC, respectively.

Table 4. FLOPs and running time(average of 10 runs) on GPU (RTX 3090). FLOPs and time are in G and s, respectively.

Methods	**540p**		**720p**		**1080p**	
	FLOPs	Time	FLOPs	Time	FLOPs	Time
VSFA [38]	6440	1.506	11426	2.556	25712	5.291
PVQ [87]	9203	1.792	13842	2.968	36760	6.556
BVQA [37]	17705	3.145	31533	7.813	70714	14.34
FAST-VQA [74]	284	0.056	284	0.056	284	0.056
FasterVQA [75]	70	0.032	70	0.032	70	0.032
DOVER [80]	282	0.061	282	0.061	282	0.062
CLiF-VQA [52]	1432	0.522	1432	0.522	1432	0.522
MBVQA [73]	912	0.194	1232	0.267	2150	0.891
MVQA-tiny	**34**	**0.028**	**34**	**0.028**	**34**	**0.028**
MVQA-middle	**403**	**0.130**	**403**	**0.130**	**403**	**0.130**

4.3. Efficiency

To test the efficiency, we compare MVQA with the current mainstream deep learning-based models. Specifically, we compare the FLOPs and GPU runtimes for videos of different resolutions, where the length of the videos are 150 frames, as shown in Tab. 4. Compared to CNN-based models (VSFA, PVQ, BVQA), MVQA-tiny reduces FLOPs by up to $756\times$, $1081\times$, and $2079\times$, as well as reduces computation time by up to $189\times$, $234\times$, and $512\times$, respectively. Moreover, compared to the fastest FasterVQA, our model has comparable computational computation as well as a $2\times$ reduction in flops. In addition, MVQA-middle not only per-

Table 5. USDS performance in the Video Swin Transformer.

Datasets	LSVQ_{test}		KoNViD-1k		LIVE-VQC	
Sampling	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Fragments	0.872	0.874	0.864	0.862	0.824	0.841
USDS	0.877	0.879	0.871	0.869	0.832	0.851

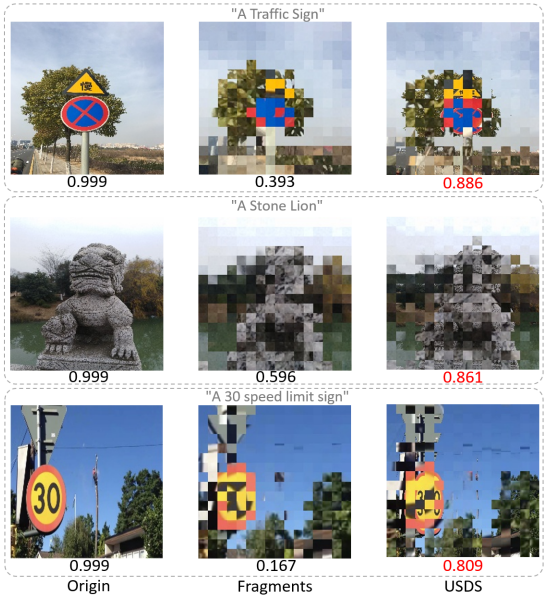


Figure 6. Results of semantic analysis experiments.

forms better but also reduce FLOPs by up to $3.55\times$ and $5.33\times$, and computation time by up to $4.02\times$ and $6.85\times$, respectively, compared to the current best models CLiF-VQA and MBVQA.

4.4. Evaluation on USDS

Semantic Analysis. We conduct experiments to verify that USDS can capture sufficient semantic information. Consistent with the testing method in Fig. 2, we use CLIP model with strong visual language capabilities to validate the semantic information in the sampled maps. As shown in Fig. 6, USDS can retain more semantics compared to Fragments. More qualitative results can be found in the Appendix.

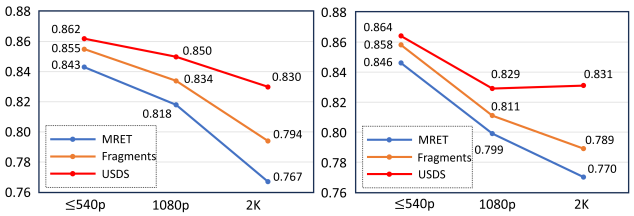
Cross-Architecture Validation. To demonstrate the advantages of USDS under different models. We validate USDS's effectiveness in the Video Swin Transformer, the dominant architecture in VQA, as shown in Tab. 5. The results show that USDS is not only effective in the proposed MVQA architecture, but also in the Video Swin Transformer model due to the traditional Fragments sampling method.

4.5. Ablation Studies

Ablation on Sampling. We compare the proposed USDS with four common sampling methods (resize, crop, MRET [33], fragments [74]) to validate its effectiveness. To ensure a fair comparison, we keep the model structure and the block size used for frame segmentation during input unchanged. As shown in Tab. 6. Specifically, USDS performs very well compared to traditional resize and crop sampling.

Table 6. Ablation study on different sampling methods in MVQA.

Datasets	LSVQ_{test}		KoNViD-1k		LIVE-VQC	
Sampling	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Resize	0.847	0.851	0.834	0.836	0.762	0.809
Crop	0.798	0.810	0.765	0.779	0.744	0.762
MRET [33]	0.851	0.854	0.841	0.839	0.785	0.820
Fragments [74]	0.864	0.865	0.850	0.853	0.791	0.829
USDS	0.882	0.883	0.870	0.868	0.828	0.848



(a) PLCC (b) SROCC

Figure 7. Ablation study on different resolutions.

Additionally, compared to the currently most effective sampling method, fragments , which can fully and evenly preserve the local texture of the video, USDS shows improvements of 2.99% and 2.00% in SROCC and PLCC, respectively. This demonstrates that USDS not only preserves the local texture of the video to the greatest extent but also retains more semantic information.

Ablation on Different Resolutions. To further validate the effectiveness of the semantic information extracted by USDS, we conducted ablation experiments at different video resolutions. Since both MRET [33] and Fragments [74] sample fixed-size blocks from the video, at higher video resolutions, the sampling results become highly fragmented, often leading to a complete loss of semantic information. As shown in Fig. 7. Since the MRET sampling strategy performs multi-scale sampling from four regions of the video, when the video resolution is too high, it can only sample from localized areas of the video. This results in the sampling not fully capturing the quality information of the video. Although the Fragments sampling method can sample uniformly from the video, at high resolutions, the features sampled using this method completely lose semantic information. The experimental results show that as the resolution increases, the advantages of USDS become more pronounced. This is because we have effectively embedded the semantic information during the sampling process.

5. Conclusion

In this paper, we are the first to explore the application of recent state-space models (Mamba) in VQA, aiming to balance computational efficiency with excellent performance. We first propose an effective sampling method, USDS, which efficiently integrates video semantic information, and introduce MVQA, a VQA model entirely based on state-space models (SSM). Extensive experiments across multiple datasets demonstrate that MVQA is a simple, effective, and highly efficient VQA model.

6. Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62272128, in part by the Startup Foundation for Introducing Talent of Nanjing University of Information Science and Technology under grant 2025r029. We also appreciate the support from China Mobile Group Heilongjiang Co., Ltd. @A001 on our research, the research is jointly completed by both parties.

References

[1] Mehdi Banitalebi-Dehkordi, Abbas Ebrahimi-Moghadam, Morteza Khademi, and Hadi Hadizadeh. No-reference video quality assessment based on visual memory modeling. *IEEE Trans. Broadcast.*, 66(3):676–689, 2019. 1

[2] Baoliang Chen, Lingyu Zhu, Guo Li, Fangbo Lu, Hongfei Fan, and Shiqi Wang. Learning generalized spatial-temporal deep feature representation for no-reference video quality assessment. *IEEE TCSVT*, 32(4):1903–1916, 2021. 1, 2, 3, 7

[3] Guo Chen, Yifei Huang, Jilan Xu, Baoqi Pei, Zhe Chen, Zhiqi Li, Jiahao Wang, Kunchang Li, Tong Lu, and Limin Wang. Video mamba suite: State space model as a versatile alternative for video understanding. *arXiv preprint arXiv:2403.09626*, 2024. 3

[4] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014. 3

[5] Sathya Veera Reddy Dendi and Sumohana S Channappayya. No-reference video quality assessment using natural spatiotemporal scene statistics. *IEEE TIP*, 29:5612–5624, 2020. 1

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 3

[7] Joshua Peter Ebenezer, Zaixi Shang, Yongjun Wu, Hai Wei, Sriram Sethuraman, and Alan C Bovik. Chipqa: No-reference video quality prediction via space-time chips. *IEEE TIP*, 30:8059–8074, 2021. 1

[8] Daniel Y Fu, Tri Dao, Khaled Kamal Saab, Armin W Thomas, Atri Rudra, and Christopher Re. Hungry hungry hippos: Towards language modeling with state space models. In *ICLR*. 2, 3

[9] Changsheng Gao, Dong Liu, Li Li, and Feng Wu. Towards task-generic image compression: A study of semantics-oriented metrics. *IEEE TMM*, 25:721–735, 2021. 1

[10] Changsheng Gao, Yiheng Jiang, Siqi Wu, Yifan Ma, Li Li, and Dong Liu. Imofc: identity-level metric optimized feature compression for identification tasks. *IEEE TCSVT*, 2024. 1

[11] Fei Gao, Yuhao Lin, Jiaqi Shi, Maoying Qiao, and Nannan Wang. Aesmamba: Universal image aesthetic assessment with state space models. In *ACM Multimedia 2024*, 2024. 2

[12] Hu Gao, Bowen Ma, Ying Zhang, Jingfan Yang, Jing Yang, and Depeng Dang. Learning enriched features via selective state spaces model for efficient image deblurring. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 710–718, 2024. 3

[13] Franz Götz-Hahn, Vlad Hosu, Hanhe Lin, and Dietmar Saupe. Konvid-150k: A dataset for no-reference video quality assessment of videos in-the-wild. *IEEE Access*, 9:72139–72160, 2021. 2

[14] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 2, 3

[15] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 2, 3

[16] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *NeurIPS*, 34:572–585, 2021. 2, 3

[17] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. In *ECCV*, pages 222–241. Springer, 2025. 2, 3

[18] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning spatio-temporal features with 3d residual networks for action recognition. In *ICCV Workshops*, pages 3154–3160, 2017. 1, 2

[19] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*, pages 6546–6555, 2018. 1, 2

[20] Rania Hassen, Zhou Wang, and Magdy MA Salama. Image sharpness assessment based on local phase coherence. *IEEE TIP*, 22(7):2798–2810, 2013. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 2, 3

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, pages 630–645. Springer, 2016. 1, 2

[23] Vlad Hosu, Franz Hahn, Mohsen Jenadeleh, Hanhe Lin, Hui Men, Tamás Szirányi, Shujun Li, and Dietmar Saupe. The konstanz natural video database (konvid-1k). In *QoMEX*, pages 1–6. IEEE, 2017. 2, 6

[24] Tao Huang, Xiaohuan Pei, Shan You, Fei Wang, Chen Qian, and Chang Xu. Localmamba: Visual state space model with windowed selective scan. *arXiv preprint arXiv:2403.09338*, 2024. 3

[25] Chen Hui, Shaohui Liu, Wuzhen Shi, Feng Jiang, and Debin Zhao. Spatio-temporal context based adaptive camcorder recording watermarking. *ACM TOMM*, 18(3s):1–25, 2022. 1

[26] Chen Hui, Shengping Zhang, Wenxue Cui, Shaohui Liu, Feng Jiang, and Debin Zhao. Rate-adaptive neural network for image compressive sensing. *IEEE TMM*, 2023. 1

[27] Md Mohaiminul Islam, Mahmudul Hasan, Kishan Shamsundar Athrey, Tony Braskich, and Gedas Bertasius. Efficient movie scene detection using state-space transformers. In *CVPR*, pages 18749–18758, 2023. 3

[28] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960. 3

[29] Le Kang, Peng Ye, Yi Li, and David Doermann. Convolutional neural networks for no-reference image quality assessment. In *CVPR*, pages 1733–1740, 2014. 2

[30] Le Kang, Peng Ye, Yi Li, and David Doermann. Simultaneous estimation of image quality and distortion via multi-task convolutional neural networks. In *ICIP*, pages 2791–2795. IEEE, 2015. 2

[31] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 6

[32] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *ICCV*, pages 5148–5157, 2021. 2

[33] Junjie Ke, Tianhao Zhang, Yilin Wang, Peyman Milanfar, and Feng Yang. Mret: Multi-resolution transformer for video quality assessment. *Frontiers in Signal Processing*, 3:1137006, 2023. 3, 5, 8

[34] Jari Korhonen. Two-level approach for no-reference consumer video quality assessment. *IEEE TIP*, 28(12):5923–5938, 2019. 1, 2, 6, 7

[35] Jari Korhonen, Yicheng Su, and Junyong You. Blind natural video quality prediction via statistical temporal features and deep spatial features. In *ACM MM*, pages 3311–3319, 2020. 7

[36] Debarati Kundu, Deepti Ghadiyaram, Alan C Bovik, and Brian L Evans. No-reference quality assessment of tone-mapped hdr pictures. *IEEE TIP*, 26(6):2957–2971, 2017. 2

[37] Bowen Li, Weixia Zhang, Meng Tian, Guangtao Zhai, and Xianpei Wang. Blindly assess quality of in-the-wild videos via quality-aware pre-training and motion perception. *IEEE TCSVT*, 32(9):5944–5958, 2022. 1, 2, 6, 7

[38] Dingquan Li, Tingting Jiang, and Ming Jiang. Quality assessment of in-the-wild videos. In *ACM MM*, pages 2351–2359, 2019. 1, 2, 3, 6, 7

[39] Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. In *ECCV*, pages 237–255. Springer, 2025. 2, 3, 5

[40] Shufan Li, Harkanwar Singh, and Aditya Grover. Mamba-nd: Selective state space modeling for multi-dimensional data. In *ECCV*, pages 75–92. Springer, 2025. 3

[41] Yang Li, Shengbin Meng, Xinfeng Zhang, Shiqi Wang, Yue Wang, and Siwei Ma. Ugc-video: perceptual quality assessment of user-generated videos. In *MIPR*, pages 35–38. IEEE, 2020. 2

[42] Liang Liao, Kangmin Xu, Haoning Wu, Chaofeng Chen, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring the effectiveness of video perceptual representation in blind video quality assessment. In *ACM MM*, pages 837–846, 2022. 1

[43] Jiarun Liu, Hao Yang, Hong-Yu Zhou, Yan Xi, Lequan Yu, Cheng Li, Yong Liang, Guangming Shi, Yizhou Yu, Shaoting Zhang, et al. Swin-umamba: Mamba-based unet with imagenet-based pretraining. In *MICCAI*, pages 615–625. Springer, 2024. 3

[44] Wentao Liu, Zhengfang Duanmu, and Zhou Wang. End-to-end blind quality assessment of compressed videos using deep neural networks. In *ACM MM*, pages 546–554, 2018. 1, 2, 3

[45] Yongxu Liu, Yinghui Quan, Guoyao Xiao, Aobo Li, and Jinjian Wu. Scaling and masking: A new paradigm of data sampling for image and video quality assessment. In *AAAI*, pages 3792–3801, 2024. 3

[46] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024. 3

[47] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pages 10012–10022, 2021. 1, 3

[48] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, pages 3202–3211, 2022. 1, 3

[49] Jun Ma, Feifei Li, and Bo Wang. U-mamba: Enhancing long-range dependency for biomedical image segmentation. *arXiv preprint arXiv:2401.04722*, 2024. 3

[50] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. In *ICLR*, 2023. 2, 3

[51] Yachun Mi, Yu Li, Yan Shu, and Shaohui Liu. ZE-FESG: A zero-shot feature extraction method based on semantic guidance for no-reference video quality assessment. In *ICASSP*, pages 3640–3644, 2024. 2

[52] Yachun Mi, Yan Shu, Yu Li, Chen Hui, Puchao Zhou, and Shaohui Liu. CLiF-VQA: Enhancing video quality assessment by incorporating high-level semantic information related to human feelings. In *ACM MM*, page 9989–9998, 2024. 2, 3, 6, 7

[53] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE TIP*, 21(12):4695–4708, 2012. 2, 6

[54] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE SPL*, 20(3):209–212, 2012. 2

[55] Anish Mittal, Michele A Saad, and Alan C Bovik. A completely blind video integrity oracle. *IEEE TIP*, 25(1):289–300, 2015. 1, 2

[56] Arnab Mondal, Stefano Alletto, and Denis Tome. Hummuss: Human motion understanding using state space models. In *CVPR*, pages 2318–2330, 2024. 3

[57] Mikko Nuutinen, Toni Virtanen, Mikko Vaahteranoksa, Tero Vuori, Pirkko Oittinen, and Jukka Häkkinen. CVD2014—A database for evaluating no-reference video quality assessment algorithms. *IEEE TIP*, 25(7):3073–3086, 2016. 2

[58] Jinyoung Park, Hee-Seon Kim, Kangwook Ko, Minbeom Kim, and Changick Kim. Videomamba: Spatio-temporal selective state space model. In *ECCV*, pages 1–18. Springer, 2025. 2, 3

[59] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 2

[60] Michele A Saad, Alan C Bovik, and Christophe Charrier. Blind prediction of natural video quality. *IEEE TIP*, 23(3): 1352–1365, 2014. 1, 2

[61] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1, 2, 3

[62] Zeina Sinno and Alan Conrad Bovik. Large-scale study of perceptual video quality. *IEEE TIP*, 28(2):612–627, 2018. 2, 6

[63] Jimmy TH Smith, Andrew Warrington, and Scott Linderman. Simplified state space layers for sequence modeling. In *ICLR*. 2, 3

[64] Wei Sun, Xiongkuo Min, Wei Lu, and Guangtao Zhai. A deep learning based no-reference quality assessment model for ugc videos. In *ACM MM*, pages 856–865, 2022. 1, 2, 3

[65] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, pages 6105–6114. PMLR, 2019. 1, 2

[66] Mingxing Tan and Quoc Le. Efficientnetv2: Smaller models and faster training. In *ICML*, pages 10096–10106. PMLR, 2021.

[67] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 1, 2

[68] Zhengzhong Tu, Yilin Wang, Neil Birkbeck, Balu Adsumilli, and Alan C Bovik. UGC-VQA: benchmarking blind video quality assessment for user generated content. *IEEE TIP*, 30: 4449–4464, 2021. 1, 2, 6, 7

[69] Zhengzhong Tu, Xiangxu Yu, Yilin Wang, Neil Birkbeck, Balu Adsumilli, and Alan C Bovik. Rapique: Rapid and accurate video quality prediction of user generated content. *IEEE OJSP*, 2:425–440, 2021. 7

[70] Jue Wang, Wentao Zhu, Pichao Wang, Xiang Yu, Linda Liu, Mohamed Omar, and Raffay Hamid. Selective structured state-spaces for long-form video understanding. In *CVPR*, pages 6387–6397, 2023. 2

[71] Yilin Wang, Sasi Inguva, and Balu Adsumilli. Youtube ugc dataset for video compression research. In *MMSP*, pages 1–5. IEEE, 2019. 2, 6

[72] Yilin Wang, Junjie Ke, Hossein Talebi, Joong Gon Yim, Neil Birkbeck, Balu Adsumilli, Peyman Milanfar, and Feng Yang. Rich features for perceptual quality assessment of ugc videos. In *CVPR*, pages 13435–13444, 2021. 2, 3, 7

[73] Wen Wen, Mu Li, Yabin Zhang, Yiting Liao, Junlin Li, Li Zhang, and Kede Ma. Modular blind video quality assessment. In *CVPR*, pages 2763–2772, 2024. 6, 7

[74] Haoning Wu, Chaofeng Chen, Jingwen Hou, Liang Liao, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Fast-vqa: Efficient end-to-end video quality assessment with fragment sampling. In *ECCV*, pages 538–554. Springer, 2022. 1, 2, 3, 5, 6, 7, 8

[75] Haoning Wu, Chaofeng Chen, Liang Liao, Jingwen Hou, Wenxiu Sun, Qiong Yan, Jinwei Gu, and Weisi Lin. Neighbourhood representative sampling for efficient end-to-end video quality assessment. *IEEE TPAMI*, 2023. 3, 6, 7

[76] Haoning Wu, Chaofeng Chen, Liang Liao, Jingwen Hou, Wenxiu Sun, Qiong Yan, and Weisi Lin. Discovqa: Temporal distortion-content transformers for video quality assessment. *IEEE TCSVT*, 2023. 2, 3

[77] Haoning Wu, Liang Liao, Jingwen Hou, Chaofeng Chen, Erli Zhang, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring opinion-unaware video quality assessment with semantic affinity criterion. *ICME*, 2023.

[78] Haoning Wu, Liang Liao, Annan Wang, Chaofeng Chen, Jingwen Hou, Wenxiu Sun, Qiong Yan, and Weisi Lin. Towards robust text-prompted semantic criterion for in-the-wild video quality assessment. *arXiv preprint arXiv:2304.14672*, 2023. 1, 2

[79] Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Towards explainable in-the-wild video quality assessment: A database and a language-prompted approach. In *ACM MM*, page 1045–1054, 2023. 3, 7

[80] Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *ICCV*, pages 20144–20154, 2023. 3, 6, 7

[81] Hongtao Wu, Yijun Yang, Huihui Xu, Weiming Wang, Jinni Zhou, and Lei Zhu. Rainmamba: Enhanced locality learning with state space models for video deraining. In *ACM MM*, pages 7881–7890, 2024. 3

[82] Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, et al. Q-align: teaching lmms for visual scoring via discrete text-defined levels. In *ICML*, pages 54015–54029, 2024. 6

[83] Jingtao Xu, Peng Ye, Yong Liu, and David Doermann. No-reference video quality assessment via feature learning. In *ICIP*, pages 491–495. IEEE, 2014. 2

[84] Jiahua Xu, Jing Li, Xingguang Zhou, Wei Zhou, Baichao Wang, and Zhibo Chen. Perceptual quality assessment of internet videos. In *ACM MM*, pages 1248–1257, 2021. 1, 2

[85] Wufeng Xue, Xuanqin Mou, Lei Zhang, Alan C Bovik, and Xiangchu Feng. Blind image quality assessment using joint statistics of gradient magnitude and laplacian features. *IEEE TIP*, 23(11):4850–4862, 2014. 2

[86] Peng Ye, Jayant Kumar, Le Kang, and David Doermann. Unsupervised feature learning framework for no-reference image quality assessment. In *CVPR*, pages 1098–1105. IEEE, 2012. 2

[87] Zhenqiang Ying, Maniratnam Mandal, Deepti Ghadiyaram, and Alan Bovik. Patch-vq:'patching up'the video quality problem. In *CVPR*, pages 14019–14029, 2021. 1, 2, 3, 6, 7

[88] Junyong You and Jari Korhonen. Deep neural networks for no-reference video quality assessment. In *ICIP*, pages 2349–2353. IEEE, 2019. 3

[89] Weihao Yu and Xinchao Wang. Mambaout: Do we really need mamba for vision? *arXiv preprint arXiv:2405.07992*, 2024. 2

[90] Zicheng Zhang, Wei Wu, Wei Sun, Danyang Tu, Wei Lu, Xiongkuo Min, Ying Chen, and Guangtao Zhai. Md-vqa: Multi-dimensional quality assessment for ugc live videos. In *CVPR*, pages 1746–1755, 2023. 1, 2, 3

[91] Zeyu Zhang, Akide Liu, Ian Reid, Richard Hartley, Bohan Zhuang, and Hao Tang. Motion mamba: Efficient and long sequence motion generation. In *ECCV*, pages 265–282. Springer, 2025. 3

[92] Kai Zhao, Kun Yuan, Ming Sun, and Xing Wen. Zoom-vqa: Patches, frames and clips integration for video quality assessment. In *CVPR*, pages 1302–1310, 2023. 1, 2, 3, 6

[93] Hanwei Zhu, Baoliang Chen, Lingyu Zhu, and Shiqi Wang. Learning spatiotemporal interactions for user-generated video quality assessment. *IEEE TCSVT*, 33(3): 1031–1042, 2022. 1, 2

[94] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024. 2, 3, 4, 5

[95] Nikola Zubic, Mathias Gehrig, and Davide Scaramuzza. State space models for event cameras. In *CVPR*, pages 5819–5828, 2024. 3