IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 August 2024, accepted 2 November 2024, date of publication 18 November 2024, date of current version 26 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3501152

RESEARCH ARTICLE

Robust Video List Decoding in Error-Prone Transmission Systems Using a Deep Learning Approach

YUJING ZHANG[1,2,3], STÉPHANE COULOMBE[1,3], (Senior Member, IEEE), FRANÇOIS-XAVIER COUDOUX[2], (Senior Member, IEEE), ALEXIS GUICHEMERRE[1,3], AND PATRICK CORLAY[2]

[1]Department of Software and IT Engineering, École de technologie supérieure, Université du Québec, Montréal, QC H3C 1K3, Canada
[2]CNRS, UMR 8520, Département d'Opto-Acousto-Électronique (DOAE), Institut d'Électronique de Microélectronique et de Nanotechnologie (IEMN), Université Polytechnique Hauts-de-France, 59313 Valenciennes, France
[3]International Laboratory on Learning Systems (ILLS), McGill-ÉTS-Mila-CNRS-Université Paris Saclay-CentraleSupelec, Montréal, QC, H3H 2T2, Canada

Corresponding author: Yujing Zhang (yujing.zhang.1@etsmtl.net)

This work was supported in part by the Conseil de Recherches en Sciences Naturelles et en Génie du Canada (CRSNG), and in part by the Université Polytechnique Hauts-de-France (UPHF).

ABSTRACT This paper introduces a novel deep-learning assisted video list decoding method for error-prone video transmission systems. Unlike traditional list decoding techniques, our proposed system uses a Transformer-based no-reference image quality assessment method to select the highest-scoring reconstructed video candidate after reception. Three new components are defined and used in the Transformer-assisted image quality evaluation metric: neighborhood-based patch fidelity aggregation, discriminant color texture transformation and ranking-constrained penalty loss function. We have also created our own database of non-uniformly distorted images, similar to those that might result from transmission errors, in a High Efficiency Video Coding (HEVC) context. In our specific testing context, our improved Transformer-assisted method has a decision accuracy of 100% for intra-coded image, while, for errors occurring in an inter image, it is 96%. Notably, in the few cases where a wrong choice is made, the selected candidate's quality remains similar to the intact frame. Code: https://github.com/Yujing0926/Robust-Video-List-Decoding-Using-a-Deep-Learning-Approach.

INDEX TERMS Video transmission, list decoding, non-uniform distortions, no-reference image quality assessment, vision transformer, convolutional neural network.

I. INTRODUCTION

Over the past few years, video devices, systems and applications have undergone extremely rapid development. This trend will only continue, as video currently accounts for almost 80% of all Internet traffic [22], [36]. Real-time video is increasingly popular and video content transmission is the most common type of data transmitted worldwide today. As the Internet of Things (IoT) technology continues to evolve, the number of applications that can greatly benefit from visual information to enhance understanding of the environment is increasing. These include remote surveillance and machine control systems, electronic health devices, virtual and augmented reality. Intelligent Transport Systems (ITS) are also directly involved; the video makes it possible to communicate information about the driving environment or the state of the transport network between the vehicles and the infrastructure.

Moreover, video quality experience has greatly improved in recent years, thanks to the advent of high-definition (HD) video and the emergence of ultra-high-definition (UHD) content. Consequently, video streams now tend to contain more data. To significantly reduce the size of these video streams, new video compression solutions have been developed [34], [35], [44].

The associate editor coordinating the review of this manuscript and approving it for publication was Long Xu.

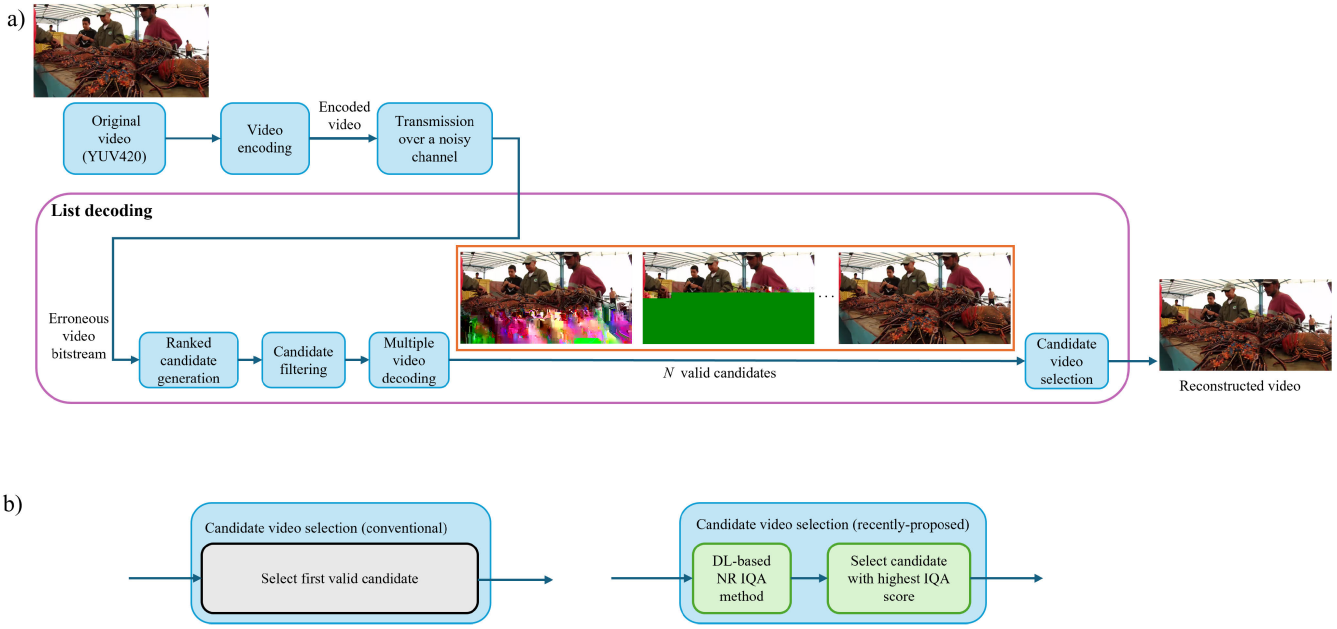


FIGURE 1. Video transmission using list decoding: a) general video transmission system using list decoding; b) different criteria of candidate video selection: left, criteria in conventional approaches; right, the recently-proposed deep-learning based approaches.

However, transmission errors on unreliable and error-prone networks, such as sensor networks and video internet objects (WiFi [2], BLE [11], etc.), can significantly degrade the user experience. These errors can cause problems like blur, geometric patterns, or green screens (see Figure 1(a) center part). However, repeatedly transmitting faulty data packets can also negatively impact network efficiency. In addition to being time-consuming and resource-intensive, such retransmissions may be incompatible with certain application domains, such as transportation and immersive video applications where video information must arrive reliably and in real time with very low latency. In such application domains, it is preferable to keep erroneous packets, even if they may lead to some visual artifacts, rather than discarding them and request new ones.

Various approaches have been proposed in the literature to find effective, low-complexity solutions for repairing video packets containing bit errors. Error concealment and error correction methods are the two main classes of approaches used to handle damaged packets. Error concealment [33], [43] is a technology applied on the decoder side to regenerate the lost information in the decoded video streams (i.e. attempt to reconstruct the parts that were damaged in the transport and subsequently discarded). Error concealment leverages the correlation of adjacent regions in the current frame (spatial concealment [19]) or previously received frames (temporal concealment [26]) or both (spatiotemporal concealment [21], [38]) to recover lost areas.

On the other hand, error correction involves identifying and correcting the erroneous bits in a packet using various strategies, such as error-correcting codes (e.g., Reed-Solomon in Digital Video Broadcasting [16]), leveraging the reliability information of each received bit (e.g. log-likelihood ratio (LLR) [3]) or utilizing newly proposed Cyclic Redundancy Check (CRC)-based error correction methods [6]. Unfortunately, the reliability information of each bit is usually not available to the video decoder. Furthermore, error-correcting codes add undesirable overhead to the communications. Therefore, since CRC is already widely used in IP communications (e.g. in UDP and TCP packets) and accessible at the application layer, CRC-based error correction is a promising approach. It uses readily available information and does not add overhead. However, in practice, CRC-based error correction can only correct a limited number of errors. Indeed, depending on the generator polynomial, packet size, and the maximum number of errors considered, the method may not lead to a unique corrected packet but, rather, a list of potentially corrected packets. This is where list decoding becomes relevant in combination with CRC-based error correction as well as with those leveraging received bit reliability.

Figure 1(a) shows a generic video transmission process using list decoding. A raw YUV video sequence is firstly encoded, complying to a video compression standard, such as H.264 [44] or H.265 (aka HEVC) [35], and then transmitted over a noisy channel. However, variable channel conditions, especially on a noisy wireless channel, may generate errors in the video bitstream. At this point, list decoding is employed to repair the corrupted video bitstream to enhance user experience. This method firstly generates an ordered list of candidates, each representing a plausible correction of the received video packet (see the *Ranked candidate generation*

part of the list decoding module in Figure 1(a)). Normally, the correct packet is included in this candidate list. When the list contains several candidates, as is usually the case, additional steps are necessary to determine which candidate to select as the corrected video packet. Traditional list decoding methods use a LLR-based formulation [3], which generates a ranked candidate list based on bit reliability. More recently, a CRC-based multi-error correction approach [6], [7] has been proposed, which generates the most probable CRC-compliant candidate video packets.

The second step in the list decoding module, illustrated in Figure 1(a), is to filter the possibly large list of candidates. This is done using additional information. For example, Checksum-Filtering [13] uses the receiver-side user datagram protocol checksum to eliminate packets whose checksum indicates an error. Similarly, CRC validation is used in LLR approaches [3], [8]. The third step of the list decoding module involves validating all candidate video packets through multiple video decoding operations. This eliminates any syntactically incorrect candidate. Subsequently, several videos may remain, leading to the final step of selecting the highest quality candidate video as the final reconstructed video.

Traditional list decoding have disadvantages in candidate video selection. As illustrated in Figure 1(b), the final candidate selection in these methods is determined by choosing the first valid candidate from the final ranked list, rather than considering the entire list for a more comprehensive evaluation. While this straightforward choice may seem appealing, it is not a rigorous process. The video sequence at the top of the list may not even have the best reconstruction quality.

Therefore, it is necessary to develop a new method for automatically selecting the highest quality video from the candidate list. We aim to achieve this by using a deep-learning (DL) system to determine the best candidate based on visual quality, as illustrated in the right part in Figure 1(b). Such DL-based approaches were recently proposed in [15] and [49]. Since errors will be handled per frame, we will focus on assessing image quality, not video quality. More specifically, each candidate will undergo evaluation by a DL-based No-Reference (NR) image quality assessment (IQA) method to obtain a score. The system will then select the candidate with the highest IQA score.

In previous research, the authors of [15] and [49] proposed a convolutional neural network (CNN)-assisted video list decoding system. This system functioned effectively on intra-coded images, but not on inter-coded images.

In this paper, we propose a novel Transformer-assisted video list decoding system that incorporates a visual quality evaluation framework using a Transformer-based metric to identify the best candidate from the list. It features a NR IQA metric based on a Vision Transformer to evaluate the quality of candidate videos, incorporating three new components: neighborhood-based patch fidelity aggregation (NPFA), discriminant color texture transformation (DCTT) and ranking-constrained penalty loss function (RCPL) to address the previous limitations.

This system assesses the quality of videos subjected to transmission errors without discarding lost packets or concealing lost regions. The distortions caused by transmission errors differ from those addressed by traditional visual quality metrics, which typically consider global, uniform image distortions. We will demonstrate that these metrics do not accurately distinguish between a correctly reconstructed version and various corrupted video versions.

This comprehensive approach, combining traditional, but revisited, list decoding techniques with a Transformer architecture to evaluate visual quality and select the best candidate, is unprecedented and delivers excellent results. Our paper makes the following key contributions:

- We propose the first Transformer-assisted video list decoding framework for error-prone video transmission systems. This framework selects the optimal candidate based on estimated visual quality from a set of options generated during the list decoding process, maximizing the final decoding's efficiency in terms of visual quality. The IQA method based on Transformer uses the structure proposed in [46] as the backbone, and we propose a neighborhood-based patch fidelity aggregation to better consider the local inhomogeneity at horizontal and vertical boundaries between adjacent patches.
- We improve the framework proposed in [15] and [49] by adding these new components:
  a) A Discriminant Color Texture Transformation is proposed to distinguish between a well-received uniform patch and an error patch initialized to be uniform by the decoder.
  b) A Ranking-constrained penalty loss function is proposed to further ensure that an intact video receives a higher quality score than a corrupted version, which is particularly important for inter images with subtle errors.

  The proposed advanced framework is designed to evaluate the quality of image subject to local distortions. It is sensitive the non-uniform distortions caused by transmission errors. With the addition of these two new components, our framework performs significantly better, both in intra-coded and inter-coded images.
- We created a database containing corrupted HEVC-encoded videos based on the original YUV video sequences collected from public datasets ([1], [27], [42]). Most existing datasets for image quality assessment focus on artificially synthesized losses or user-generated losses. However, there lacks datasets with various types of non-uniform distortions caused by transmission errors. Therefore, we created the scripts and instructions for regenerating the database, including the standard HEVC ([37]) encoding and addition of transmission errors to obtain non-uniform corrupted frames. Simple error patterns are applied

to the encoded video packets to mimic its passage through unreliable networks. Then, we collect the combination of $p \times p$ patches, which is called "super-patch" in the corrupted frames from these decoded video bitstreams, to train and test with the ground-truth neighborhood-based patch fidelity aggregation scores. We collect 90 original videos with 1920×1080 resolution from public datasets. All videos are encoded and decoded without error concealment to show the different distortions. Our scripts can be found on GitHub (https://github.com/Yujing0926/Robust-Video-List-Decoding-Using-a-Deep-Learning-Approach).

This paper is organized as follows. In Section II, we review the methods for no-reference image quality assessment from the literature. Section III presents the proposed deep-learning assisted video list decoding framework for error-prone video transmission systems, including the proposed Transformer-based image quality estimation. In Section IV, we present the experimental results and the ablation study. Finally, we conclude in Section V.

II. RELATED WORK

In this section, we review existing NR IQA methods, including both traditional and deep-learning-based approaches. By discussing these methods and highlighting their disadvantages, we underscore the necessity of the Transformer-assisted metric, which is well suited for our video list decoding system.

A. NO-REFERENCE IMAGE QUALITY ASSESSMENT

1) TRADITIONAL NR IQA

Several traditional NR IQA methods, such as BRISQUE [24], NIQE [25] and PIQE [41], use natural scene statistics or perception-based features from natural videos to evaluate the image quality without reference. These NR metrics perform well for evaluating the quality of images subject to uniform distortions. But our goal is to evaluate the image quality under non-uniformly distributed distortions caused by transmission errors such as those illustrated in the *N valid candidates* of Figure 1(a). Unfortunately, as we will demonstrate in the experimental results, such existing traditional metrics do not perform well in evaluating image quality when dealing with non-uniformly distributed distortions caused by bit errors.

2) CNN-BASED NR IQA

With the continuous development of deep learning technologies, more and more research focuses on applying deep learning to image quality assessment [40]. Relying on the ability of deep learning neural networks to process images, numerous image evaluation solutions based on deep learning technologies have emerged in recent years. Several studies have been proposed applying CNN in image quality assessment [4], [17], [20], [48]. These deep learning models perform better than traditional models in image quality assessment. For example, the authors of [17] proposed a patch-based approach where all patches in the image are assigned the same quality level as the entire image when learning. In [4], the authors proposed a data-driven approach based on CNN, where features and natural scene statistics are learned purely data-driven and combined with pooling and regression in one framework. Another deep bilinear model is presented in [48], which handles both synthetic and authentic distortions. These metrics allow using larger databases for simulation and more types of erroneous images can be evaluated without reference. However, these methods tend to assess the global quality of the entire image rather than the local quality, often overlooking local distortions in specific regions, which makes them unsuitable for our problem without, at least, a retraining on a visual database containing videos representative of those decoded after bit errors.

3) TRANSFOMER-BASED NR-IQA

With the large number of applications and the rapid development of the Transformer model [12], [39] in the field of image processing, numerous methods for evaluating image quality based on the Transformer model have appeared recently [9], [10], [14], [45], [46], [47]. The models based on Transformers always divide the whole image into several small patches first, then flatten them and enter them into the Transformer's encoder, so that it can learn the attention of these patches and assess image quality. By using an attention mechanism to rapidly calculate the importance and the relation between the patches, these methods improve the efficiency of processing large amounts of image data and assessing image quality. In [47], the authors proposed an architecture of using a shallow Transformer encoder on top of a feature map extracted by CNN. The authors of [45] proposed a local distortion extractor to obtain local distortion features from a pretrained CNN and a local distortion injector to inject the local distortion features into ViT [12]. The experiments presented in these papers also demonstrate that the results of evaluating this mixed objective model are more consistent with human visual perception.

B. NR-IQA MODELS FOR NON-UNIFORM VIDEO DISTORTIONS

Inspired by these works, the authors of [49] proposed a CNN-based image quality estimation metric applied to a deep-learning assisted video list decoding framework.

As shown in Figure 2, they chose the CNN architecture proposed in [17] as the backbone and improved it in several aspects to meet their objectives. The improvements include patch-based local normalization in a quality measurement approach to support non-uniform distortions in images. Firstly, the original CNN method poses a problem when applied to uniform patches: it cannot distinguish between a well-received uniform patch and an erroneous patch initialized to zero in YUV color space by the decoder. This leads to a green uniform patch. This issue is problematic when

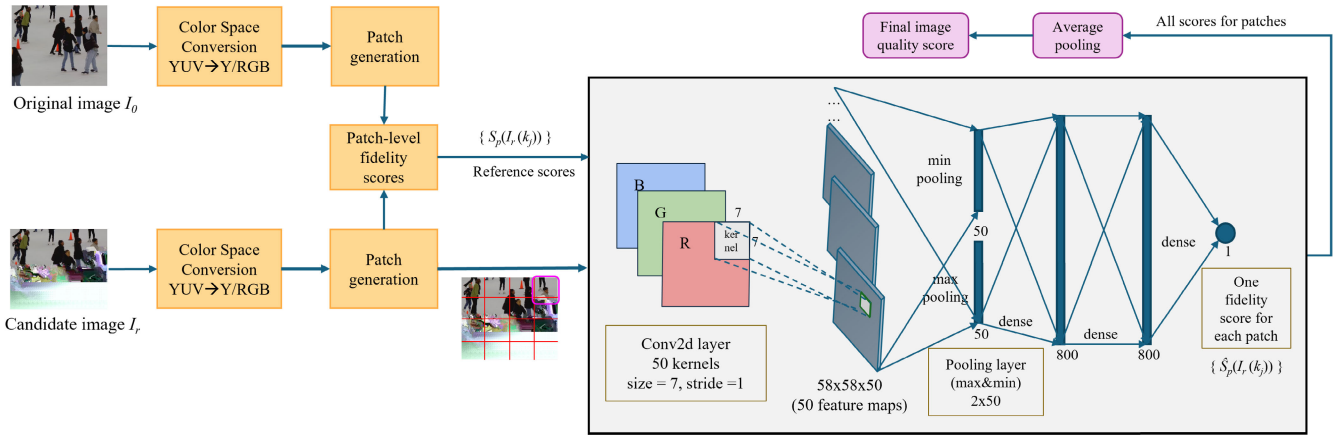


FIGURE 2. The previously proposed CNN-based IQA metric for ranking video candidates in list decoding [49]. The original image is only used during training.

it occurs in the training database, as it confuses the neural network during training. After normalization, a uniform patch and an error patch become identical and enter the layers of the CNN with different reference scores for learning. To address this, the authors of [49] improved local normalization by separating the two situations: a well-received uniform patch is normalized to a value of α, with $\alpha \neq 0$, while an erroneous uniform patch, initialized to 0 by the decoder, remains 0 after normalization.

The authors of [49] noticed that their method performed well on intra-coded images, at one point achieving 100% candidate selection accuracy. However, for inter-frame encoded images, they noted that the precision was around 80%, which is significantly lower than for intra-coded images and still leaves much room for improvement. We found that simply changing local normalization, as done in [49], has its limitations.

We aim to go further by better distinguishing between a well-received uniform patch and an erroneous patch that appears uniform, and by improving the performance of inter-coded images by ensuring that an intact patch receives a higher quality score than a corrupted version, which is particularly important for inter-coded images with small distortions.

III. PROPOSED METHOD

In this section, we firstly present the proposed transformer-assisted video list decoding framework. Then, we focus on the transformer-based image quality assessment process and explain all its new components: discriminant color texture transformation (DCTT), neighborhood-based patch fidelity aggregation (NPFA), and ranking-constrained penalty loss function (RCPL).

A. OVERALL PROPOSED TRANSFORMER-ASSISTED VIDEO LIST DECODING FRAMEWORK

We propose an enhanced and transformer-based version of the CNN-assisted video list decoding framework proposed in [15] and [49]. This framework uses a no-reference IQA metric based on the Transformer architecture [39] to evaluate candidate image quality.

As shown in Figure 3, the architecture of our metric consists of several blocks, including image pre-processing (e.g. color space conversion), network training and final image score calculation. For each original YUV video sequence, a list of candidate videos is generated by encoding the original sequence, injecting errors at various locations and decoding it. The first corrupted frames from each candidate sequence are extracted to enter into our network. The candidate frame is represented by I_r, and each candidate list includes an intact version I_i, which is received without error. We also prepare the corresponding original frame version I_o. Before entering the images into the training network, the image pre-processing is proposed with several steps: firstly, we apply the image color space conversion to change the image format from YUV420 to RGB, with a DCTT (see Section III-C1) component to distinguish well the uniform regions caused by errors and actual flat areas. Secondly, we generate the patches $I(k_j)$ for each version of the image and calculate the patch-level full-reference fidelity scores $S_p(I(k_j))$. Then we combine $p \times p$ normal patches to generate *super-patches* and use the proposed NPFA (see Section III-B1) to generate the reference scores for super-patches to better consider the local discrepancy at horizontal and vertical boundaries between adjacent patches. The size of the super-patches and the aggregation method are variable parameters which could be changed in the future.

Our method is based on MANIQA [46], aiming to estimate the IQA for our specific case. We use super-patches as the input data for the neural network and propose the RCPL (Ranking-constrained penalty loss function, see Section III-C2) to guarantee the predicted score of the corrupted super-patch to be lower than the corresponding intact version. The following subsection introduces in more details our proposed Transformer-assisted IQA metric with three new proposed components.

B. TRANSFORMER-BASED IMAGE QUALITY ESTIMATION METRIC

To address the limitations of the CNN-based approach, we propose using a more comprehensive deep learning architecture, the Vision Transformer [12]. Transformer models can learn to focus on the local patches and evaluate image quality. By employing an attention mechanism to swiftly compute the significance and interrelations among patches, such approach enhances the efficacy of handling extensive image datasets and evaluating image quality.

Our proposed Transformer-based image quality estimation metric is sensitive to local distortions in the image, which are non-uniformly distributed, based on a self-attention mechanism. Our proposed Transformer-assisted framework is capable of using a rigorous process to select the video candidate with the best visual quality.

In Figure 3, the gray blocks present four components existing in the original method: a feature extractor using ViT([12]), a transposed attention block, a scale swin Transformer block, and a two-branch structure for patch-weighted quality prediction. This method first extracts and connects four layers of features from ViT, and then computes the weights of different channels by the proposed transposed attention block (TAB). The authors apply the Self-Attention algorithm across channels rather than spatial dimensions to compute the mutual covariance across channels to generate the attention graphs in this module. To enhance the local interactions between image blocks, a Scale Swin Transformer Block (SSTB) is applied. Finally, a two-branch structure consisting of weighted and scored branches for the importance of each patch is applied and a quality prediction is presented to obtain the final score $\hat{S}_{sp}(I_r(k_j))$ of each super-patch. Finally, we collect all the predicted super-patch scores for each image and use an average pooling to obtain the image-level quality score.

The blocks in red dotted lines in Figure 3 are used only for training, and the blocks with solid lines are used in the inference process of the proposed Transformer-assisted NR-IQA system to select the best quality candidate, as discussed in Section I.

1) NEIGHBORHOOD-BASED PATCH FIDELITY AGGREGATION

After reproducing the simulations with the CNN IQA metric in [49], we found that this system still has many shortcomings, such as the inability to detect discontinuities between neighbouring coding tree unit (CTU) blocks when trained patch size is the same as the coding CTU size. Furthermore, replacing the CNN metric with a Transformer-based metric poses a challenge due to the small size of the individual patches used in the CNN model, which is not compatible with the basic Transformer architecture [46]. Therefore, we introduce super-patches instead of simple individual patches, where super-patches are the combination of $p \times p$ patches such that a complete image is divided into overlapping super-patches. The super-patches contain several neighbouring CTU blocks, enabling the model to more effectively analyze localized distortions resulting from error propagation in the neighbourhood of a given block. We demonstrate the necessity of super-patches for our research in the following.

We propose using neighborhood-based patch fidelity aggregation (NPFA) to analyze super-patch, which constitutes the first originality of our method. $\hat{S}_{sp}(I_r(k))$ presents the combination of the patch-level fidelity score $\hat{S}_p(I_r(k))$ associated with several adjacent patches, which is calculated in Eq. (1). $f(k, i)$ returns patch number i in the neighbourhood of patch k and COMB is a function to aggregate the score of multiple adjacent patches. The COMB function can be selected among various aggregation functions, such as average, minimum or power pooling to obtain the aggregation score for each super-patch. This allows us to give greater importance to local errors instead of computing simple averages. This is in accordance with the fact that quality assessment is not a global process but a local process based on several regions of interest that are more degraded.

$$\hat{S}_{sp}(I_r(k)) = \mathrm{COMB}(\hat{S}_p(I_r(f(k,1))), \hat{S}_p(I_r(f(k,2))), \ldots, \hat{S}_p(I_r(f(k,n))))$$

where

$$\mathrm{COMB}(\hat{S}_p(I_r(f(k,1))), \hat{S}_p(I_r(f(k,2))), \ldots, \hat{S}_p(I_r(f(k,n)))) = \begin{cases} \min\limits_{i\in[1,n]} \hat{S}_p(I_r(f(k,i))), & \text{if minimum} \\ 1 - \dfrac{1}{n}\sum\limits_{i=1}^{n}[1 - \hat{S}_p(I_r(f(k,i)))]^2, & \text{if squared error} \\ \dfrac{1}{n}\sum\limits_{i=1}^{n} \hat{S}_p(I_r(f(k,i))), & \text{if average} \end{cases} \tag{1}$$

Ideally, we want all patches of the intact frame to receive from the IQA system a score higher than or equal to the corresponding patches in any candidate frame. This applies whether we are dealing with individual patches or with super-patches. Formally, we would like:

$$d_{k,p} = \hat{S}_p(I_i(k)) - \hat{S}_p(I_r(k)) \geq 0 \tag{2}$$

$$d_{k,sp} = \hat{S}_{sp}(I_i(k)) - \hat{S}_{sp}(I_r(k)) \geq 0 \tag{3}$$

Once the system is trained to estimate the quality of each patch, we can evaluate the performance per patch and per super-patch with different NPFA methods, before establishing the performance for the whole image. Evaluating the performance at the super-patch level using various aggregation functions will indicate which to select. Table 1 shows the analysis results by reproducing the CNN predicted patch scores from intra-coded RGB images in [49], with patch size of 64×64, and CNN model trained with the improved local normalization algorithm. We used $p = 2$ to combine the super-patches from the individual patches in the analysis.

According to Table 1, the columns $d_k < 0$, $d_k > 0$ and $d_k = 0$ represent the percentage of each situation in all situations, which have negative, positive or zero d_k values,

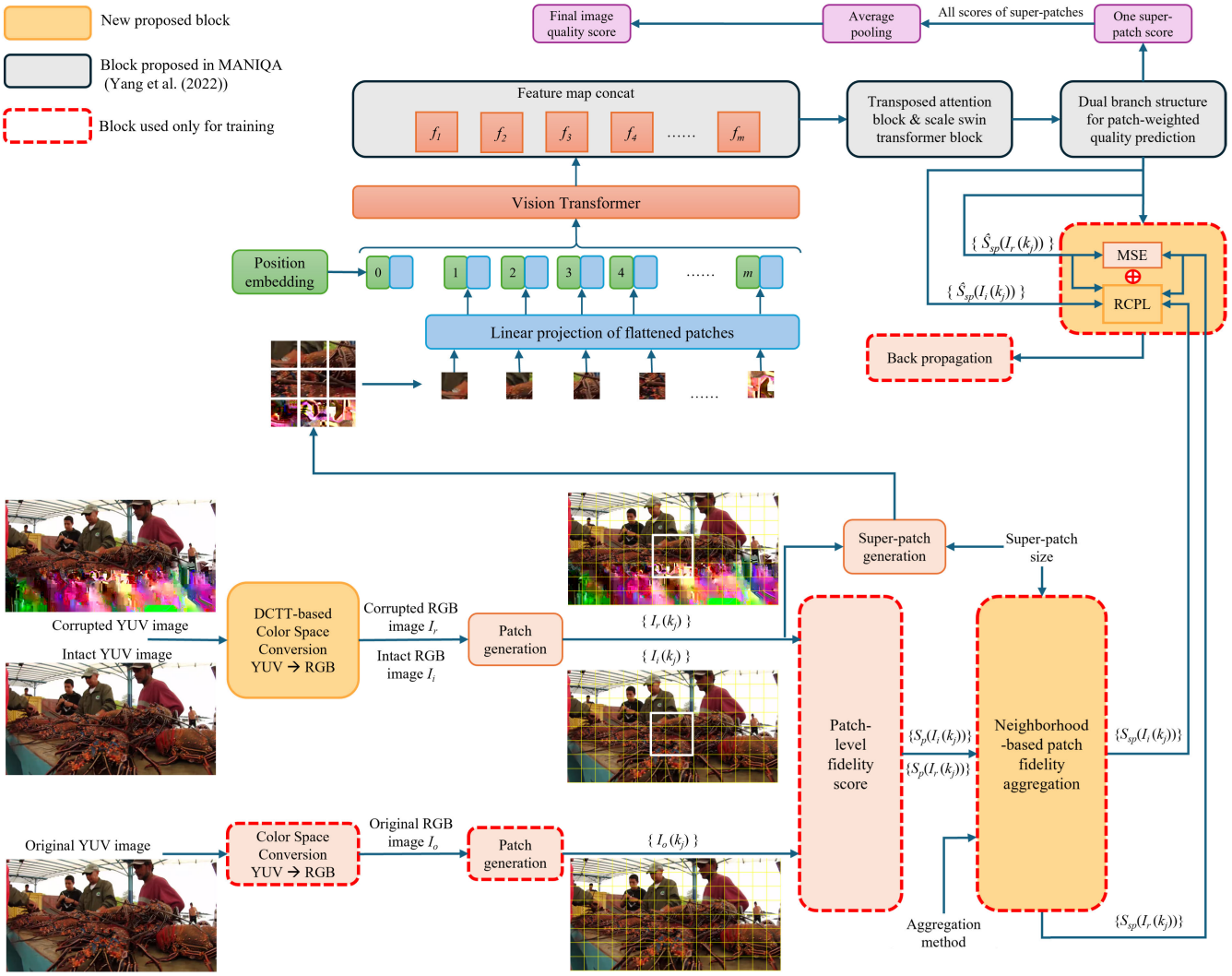


FIGURE 3. The proposed Transformer-assisted image quality estimation metric.

TABLE 1. Performance results using, $d_{k,p}$ (first line) and $d_{k,sp}$ (for average, minimum and squared error aggregation functions), of patch and super-patch quality estimation with CNN-predicted scores. The table shows that using super-patches with the *minimum* aggregation function leads to the best performance among tested options.

Combination type	$d_k < 0$	$d_k > 0$	$d_k = 0$
Individual patch	4.17%	45.62%	50.21%
Average	3.22%	50.69%	46.08%
Minimum	**1.84%**	49.58%	48.57%
Squared error	3.16%	50.75%	46.08%

respectively. We observe that super-patches can help further enhance performance compared to regular individual patches. The patch score predicted by CNN [49] shows that the individual patch score makes more errors than using the aggregation methods to calculate the super-patch score. Also, the minimum aggregation function performs best.

Furthermore, if we use the full image as input to the model, due to the high resolution of the image we are using (1920 × 1024), randomly cropping the image to a size of 224 × 224 from the training dataset, as proposed in the original MANIQA article ([46]) is not sufficient to guarantee that the model learns completely the local distortions in different regions of the image. Therefore, we propose to use overlapping super-patches instead of randomly cropping the image. This not only ensures that the model fully learns the different types of local distortions in different images, but also increases substantially the amount of training data and reduces the possibility of overfitting the model due to insufficient data.

We define our super-patches by combining the original patches, where each original patch size is 32 × 32 pixels. That is, one super-patch is composed of an integer number p of patches, both in horizontal and vertical directions. We set experimentally $p = 7$ to create super-patches of size 224 × 224 pixels. This choice ensures compatibility

with the input size expected by the original MANIQA model [46].

For testing purposes, we use the average of the predicted super-patch scores for each image to obtain the image-level quality score $\hat{S}(I_r)$:

$$\hat{S}(I_r) = \frac{1}{K_{sp}} \sum_{k=1}^{K_{sp}} \hat{S}_{sp}(I_r(k)) \tag{4}$$

where $\hat{S}_{sp}(I(k))$ denotes the quality score predicted for super-patch $I_r(k)$ by our Transformer metric, and K_{sp} is the total number of super-patches in the image.

C. IMPROVEMENTS TO THE FRAMEWORK APPLIED TO CNN AND TRANSFORMER

In this subsection, we introduce two other new components: Discriminant Color Texture Transformation (DCTT) and Ranking-constrained penalty loss function (RCPL). These two components can be applied both in CNN-assisted framework and transformer-assisted versions.

1) DISCRIMINANT COLOR TEXTURE TRANSFORMATION

In the previous work [49], as illustrated in Figure 2, the authors separated an image into several smaller patches and extracted features from each patch to assess their quality. They proposed to use local scores so that each patch has its own quality score. This was expected to help the neural network to learn local distortions more efficiently. They also proposed a solution to distinguish between a well-received uniform patch whose value is normalized to 0 and an erroneous patch that is uniform because it has been initialized to 0 by the decoder.

However, we found that simply changing local normalization still does not completely differentiate between the two. Instead, we propose a new discriminant color texture transformation (DCTT) based color space conversion to solve this problem during the conversion of the original YUV image to RGB. This transformation creates a totally different pattern for each channel of an RGB image instead of simply forcing the value to (0,0,0) when a spatial area is lost due to transmission errors. Since the decoder initializes each YUV pixel of an image to (0,0,0) prior to decoding, the value of lost pixels remains zero when an error occurs. Therefore, lost regions are easy to identify. For each pixel where YUV values are all detected as 0 in patch k of image I_r, we apply Eq. (5):

$$\begin{aligned} I_{r,R}(k,i,j) &= 255((-1)^{i+j}+1)/2 \\ I_{r,G}(k,i,j) &= 255((-1)^{i}+1)/2 \\ I_{r,B}(k,i,j) &= 255((-1)^{j}+1)/2 \end{aligned} \tag{5}$$

where $I_{r,R}(k,i,j)$ represents the value after the conversion, of red channel for pixel (i,j) in patch k of image I_r, and, similarly, $I_{r,G}(k,i,j)$, $I_{r,B}(k,i,j)$ represent the values of the green and blue channels, respectively. These patterns do not exist in natural RGB images; they are high-energy high-frequency patterns which do not exhibit the strong inter-channel correlation observed in natural images. We expect that patches having these patterns, being very different from other patches, will be assigned an extremely low quality score through the learning process. Figure 4 shows an example of DCTT applied to two uniform patches. The candidate image at the top has its bottom region initialized to zero in all channels Y,U,V since it was received erroneously. A patch within this region normally appears green. After applying the proposed DCTT-based color space conversion from YUV to RGB, the transformed patch exhibits a texture in all color planes that is visually distinct from the transformed uniform patch shown at the bottom of Figure 4. With these new patch patterns, the neural network will be able to differentiate between uniform patches from erroneous regions, which receive low scores, and those from intact regions, which receive high scores. After normalization, a uniform patch and an erroneous patch become dissimilar, allowing the network to train with different reference scores.

2) RANKING-CONSTRAINED PENALTY LOSS FUNCTION

Previous works [15], [49] used the mean absolute error (L1 loss) function to calculate the loss during training, while the MANIQA model [46] employed the Mean Squared Error (MSE) loss. These single loss functions performed well on intra-coded images, but there remains room for improvement when applied to inter-coded images. Therefore, we consider improving the loss function of our proposed system to further ensure that an intact super-patch receives a higher quality score than a corrupted version, which is particularly important for inter-coded images where distortions resulting from a transmission error are not as severe as for intra-coded images.

As mentioned in section III-B1, we would like to make sure that the estimated score of a super-patch from a tentatively repaired image is smaller than (or equal to) that of the associated intact one. Once trained, we can establish the performance per super-patch with the NPFA method with 'minimum' function as shown in Table 1. We notice that the proper selection of the aggregation function is important but that we should also push the system to avoid such negative differences in the first place by adding a penalty term to the loss function, i.e. add a term $F2$ to the loss function as a penalty when $\hat{S}_{sp}(I_r(k)) > \hat{S}_{sp}(I_i(k))$.

As shown in the Eq.(6), F_1 is the original loss function (mean squared error) used in the MANIQA [46] system. F_2 is the new loss term we add to the loss function, F, during training. By adding F_2, we impose a penalty when the predicted score of the corrupted super-patch exceeds that of the corresponding intact super-patch. I represents an image that was coded, transmitted, and decoded. The image with transmission errors may not be decodable. I_i is defined as the intact image associated with image I, i.e. just coded and decoded without error. I_o is the original image associated with image I, i.e. without any compression. I_r is the repair tentative version r of image I, where $r \in [1, R]$. $\hat{S}_{sp}(I_r(k))$ represents the predicted score of the super-patch k by our proposed

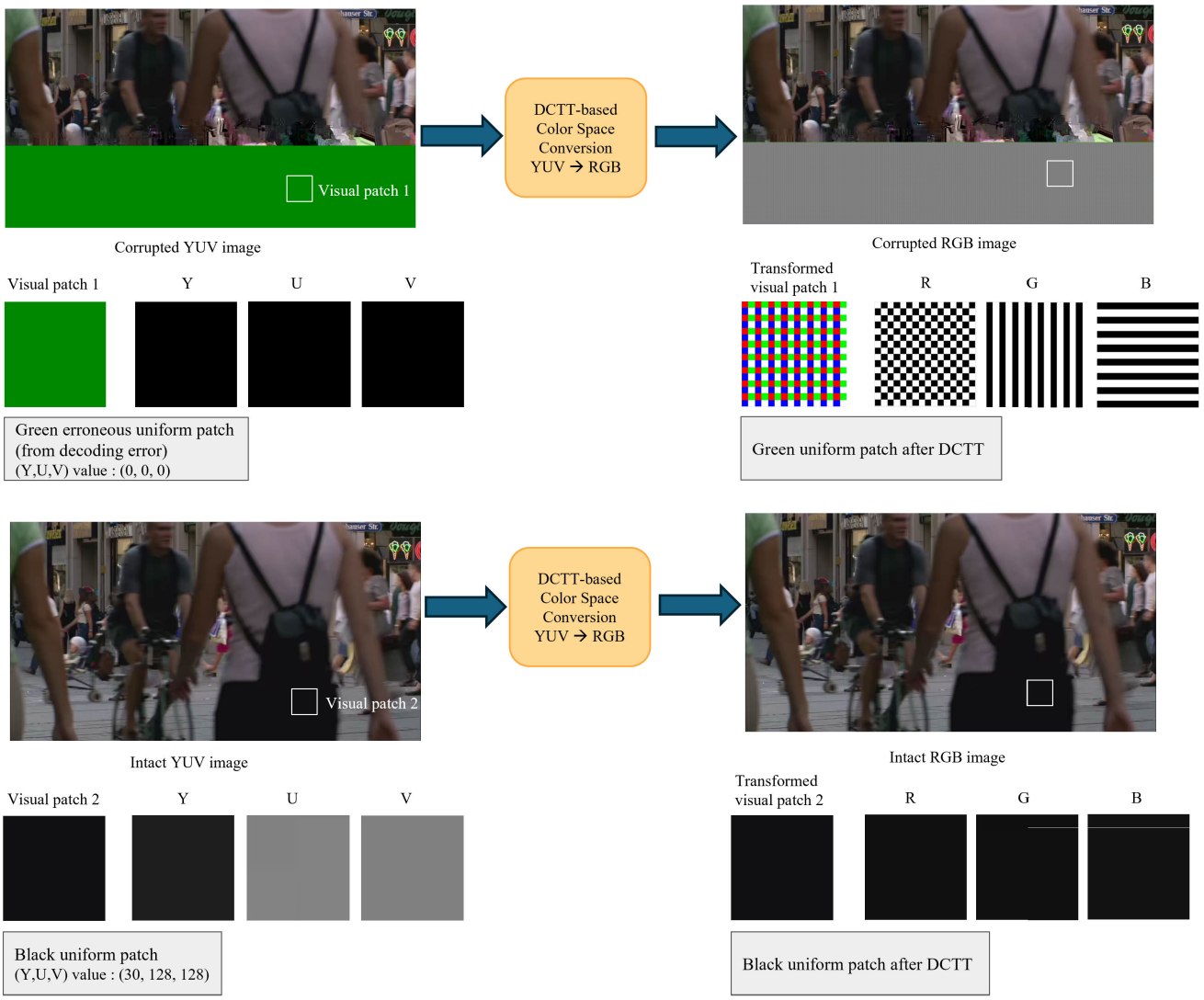


FIGURE 4. Example of the application of the proposed discriminant color texture transformation (DCTT), as part of the YUV to RGB conversion, to an erroneous green patch (top) and to an intact uniform patch (bottom).

Transformer-based model, $S_{sp}(I_r(k), I_o(k))$ is the ground truth score of each super-patch k and $\hat{S}_{sp}(I_i(k))$ is the predicted score by the Transformer-based model of the corresponding intact super-patch. $S_{sp}(I_r(k), I_i(k))$ is the actual score between the corrupted super-patches and the super-patches transmitted without errors. K_{sp} is the number of super-patches in each candidate image.

$$F_1 = \frac{1}{K_{sp}} \sum_{k=1}^{K_{sp}} ||\hat{S}_{sp}(I_r(k)) - S_{sp}(I_r(k), I_o(k))||^2$$

$$F_2 = \frac{1}{K_{sp}} \sum_{k=1}^{K_{sp}} \max(0, \hat{S}_{sp}(I_r(k)) - \hat{S}_{sp}(I_i(k)) + \delta)$$

$$F = \begin{cases} \min F_1, & \text{if } S_{sp}(I_r(k), I_i(k)) = 1 \\ \min(\alpha F_1 + (1-\alpha) F_2), & \text{if not} \end{cases} \tag{6}$$

By adding F_2, we tend towards 0 by trying to ensure that $\hat{S}_{sp}(I_r(k))$ is smaller than $\hat{S}_{sp}(I_i(k))$. At the start of training, we will surely have $\hat{S}_{sp}(I_r(k)) - \hat{S}_{sp}(I_i(k)) > 0$ so the max value between 0 and the difference will take the value of $\hat{S}_{sp}(I_r(k)) - \hat{S}_{sp}(I_i(k))$. When the predicted corrupted super-patch has a score lower than the score of the predicted intact super-patch, then we have 0. The purpose of adding the variable epsilon is to guarantee that $\hat{S}_{sp}(I_r(k)) < \hat{S}_{sp}(I_i(k))$ to avoid equality between corrupted and intact super-patches, so one should apply the constraint function only if $\hat{S}_{sp}(I_r(k)) - \hat{S}_{sp}(I_i(k)) \neq 0$. If at the start, the super-patches scores between the intact and corrupted images are the same, we do not want to penalize the network. So we add a condition of F_2: when we test $S_{sp}(I_r(k), I_i(k)) = 1$, that is to say $I_r(k)$ and $I_i(k)$ have the same reference scores, we only consider F_1. Otherwise, we consider the two loss functions together, where we use $\alpha \in [0, 1]$ to represent the

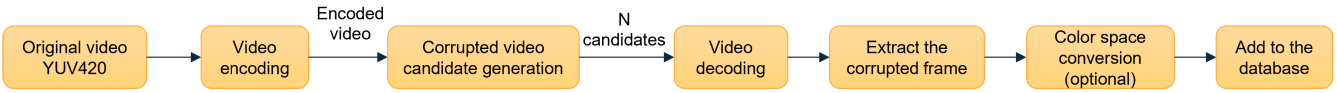


FIGURE 5. The database generation process.

coefficient of the loss function. We hope to train the system to find the parameters allowing the new loss functions to be minimized with a suitable coefficient α.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we present our experimental results. We start by describing the databases used for training and testing. Next, we outline the training methodology and evaluation criteria. Finally, we provide a comprehensive performance evaluation, including results, an ablation study, parameter sensitivity analysis, and discussion.

A. DATABASES USED FOR TRAINING AND TESTING

Most existing datasets for image quality assessment focus on artificially synthesized (simulated) distortions found in user-generated content [23], [28], [32]. However, there lacks datasets that encompass various types of non-uniform distortions resulting from transmission errors, where the content is decoded without error concealment. Indeed, the LIVE database [32] contains distortions resulting from transmission errors, but the erroneous regions are discarded and concealed rather than decoded and rendered as in our case. Therefore, we developed some scripts and instructions to generate the desired database. The database generation process is shown in Figure 5. The process includes several steps: video encoding by the HEVC standard, generating errors by flipping bits in specific positions in the video bitstreams and video list decoding, without error concealment, to obtain the N candidates representing various unsuccessful attempts to correct the video. After decoding all the video candidates, we extract the corrupted frames in each video sequence, if decodable, and add them to our database.

We now provide more details on the generation process. Similar to the database created in [49], we use the same original sequences ([1], [27], [42]). The collected videos are in YUV420 format with a resolution of 1920 × 1024 pixels. We extract the first 10 frames from each video to encode them with the HEVC standard [37] of the low-delay P profile. Among the different possible Quantization Parameter (QP) values, we chose 37 and 22, which correspond, respectively, to the low and high bit rate operating points of the HEVC standard reference software (HM) Common Test Conditions [5]. We assume that each encoded frame is contained in a single video packet. The first frame of the encoded video is an intra-coded (I) frame, and the next 9 frames are inter-coded (P) frames.

We want to simulate the combination of a transmission error followed by list decoding where bits are inverted at different locations, i.e., to spread the error throughout the video frame from beginning to the end. To simplify the process, we place bit errors at various locations in the packet. This method is compatible with list decoding scenarios, where the bits altered to generate candidates appear in random-like, unpredictable locations. For instance, in CRC-based error correction, as mentioned in [6], the candidates exhibit patterns with bits altered in such unpredictable locations.

Therefore, we selected flipped (inverted) bit positions based on the equation $pos = \beta \times M$, where pos indicates the flipped bit position, $\beta = \{0.1, 0.2, 0.3, \ldots, 0.9, 0.99\}$ and M is the size of each packet [49]. This approach ensures a significant diversity in the transmission error patterns to train the system. We incorporate transmission error patterns separately for intra-coded frames and inter-coded frames, depending on the scenario under study. Consequently, errors in inter-coded frames are directly applied to the frame itself, rather than being propagated from errors in previous frames. The candidate frames subject to these various errors are decoded and added, if decodable, to our database. For each sequence and each frame type, we generate 11 candidates, including one error-free (intact) candidate. This results in 990 corrupted images from 90 reference images, forming our sequence-based database.

B. TRAINING AND TESTING METHODOLOGY

Our experiments are conducted using two NVIDIA RTX A6000 GPUs with PyTorch 2.3.0 and CUDA 12.1 for training and testing. We trained our model on our proposed super-patch-based datasets described in the next paragraph. We tested the model on our proposed sequence-based database. Based on the backbone model MANIQA [46], we also use ViT-B/8 [12] as our pre-trained model, which is trained on ImageNet-21k [29] and fine-tuned on ImageNet1k [30] with the patch size P set to 8.

Our database has about 830 000 overlapping super-patches for each frame type in our simulation, with a super-patch size of 224 × 224 pixels. Each super-patch is associated with a peak signal-to-noise ratio (PSNR) [31] score between the reconstructed version and the original version in the interval [0, 50] dB, which is normalized to the interval [0, 1] during training. Based on the previous works [15], [49], PSNR and SSIM scores are both adequate choices for patch-level reference scores. We choose PSNR as the full-reference patch-level fidelity score, as it is a low-complexity widely recognized distortion metric. The PSNR scores are aggregated using the proposed NPFA with *minimum* aggregation function to calculate the super-patches scores from the individual patches of size 32 × 32 (see Eq. (1)). For our experiments, our sequence-based database is randomly split into 60:40 ratio, with 60% sequences used for training

and the remaining 40% for testing. Following the standard training strategy outlined in existing IQA algorithms [46], we created our super-patch-based dataset by cropping our sequence-based database into super-patches and randomly split it into 80:20 ratio, with 80% allocated for training and the remaining 20% for validation. During training, we set the learning rate l to 10^{-5} and the batch size B to 8. We used the ADAM optimizer [18] with weight decay 10^{-5}. The training loss used is the proposed RCPL, where F_1 is the MSE loss, and $\alpha = 0.5$. The final score is generated by averaging the scores predicted for all super-patches in each image.

Note that we train our system on QP = 37 because it represents a higher quantization parameter, which introduces more compression artifacts and a higher level of distortion. This challenging scenario allows the model to learn how to handle significant visual degradation and spatial error propagation, making it robust and effective in improving visual quality under difficult conditions. By testing on both QP = 37 and QP = 22, we can evaluate the model's performance across a range of compression levels, ensuring it is versatile and performs well not only in high-distortion scenarios (QP = 37) but also in lower-distortion, higher-quality scenarios (QP = 22). This comprehensive testing demonstrates the model's ability to generalize and maintain high visual quality across different levels of video compression.

C. PERFORMANCE EVALUATION CRITERIA

We trained and tested the original MANIQA [46] model and our proposed improved version on the newly developed database. We use the accuracy and the metrics described in Eq. (7) to evaluate the performance of the various IQA models with different configurations. In the equation, $\overline{S}_{\text{intact}}$ indicates the average PSNR between the n-th intact image $I_{i,n}$ and its original versions $I_{o,n}$, over all video sequences, where intact versions are compressed but received without transmission errors. Here, S is the PSNR score calculated on the RGB color space, and N represents the total number of video sequences. $\overline{S}_{\text{system}}$ represents the average quality returned by the system, which is calculated by the average PSNR value between the version selected by the system (e.g. $I_{s,n}$) and the original version $I_{o,n}$, over all N sequences. I_s is selected by our system with the highest predicted quality score from all the reconstructed candidate images $I_{r,i}$. $\overline{S}_{\text{diff}}$ gives the difference between the quality returned by the system when intact images are selected and by the proposed deep-learning based list decoding system.

$$\begin{aligned}
\overline{S}_{\text{intact}} &= \frac{1}{N}\sum_{n=1}^{N} S(I_{\text{o},n}, I_{\text{i},n}),\\
\overline{S}_{\text{system}} &= \frac{1}{N}\sum_{n=1}^{N} S(I_{\text{o},n}, I_{\text{s},n}),\\
\text{where } I_{\text{s}} &= \underset{\{I_{r,i},\ 1\le i\le R\}}{\arg\max}\, \hat{S}(I_{r,i}),\\
\overline{S}_{\text{diff}} &= \left|\overline{S}_{\text{intact}} - \overline{S}_{\text{system}}\right| \qquad (7)
\end{aligned}$$

TABLE 2. Performance on intra-coded images with the Transformer model. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
PIQE [41]	30.4%		19.84	14.92
NIQE [25]	5.4%		16.30	18.46
BRISQUE [24]	19.6%		18.29	16.47
CNN_NR_IQA [17]	46.4%		24.78	9.98
CNN_NR_IQA_RE	96.4%	34.76	34.44	0.32
CNN_NR_IQA_NL [49]	**100%**		**34.76**	**0.00**
MANIQA [46]	75.0%		28.84	5.92
CNN_DCTT_RCPL	**100%**		**34.76**	**0.00**
MANIQA_DCTT_RCPL	**100%**		**34.76**	**0.00**

TABLE 3. Performance on inter-coded images with the transformer model. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
PIQE [41]	28.6%		24.38	9.84
NIQE [25]	17.9%		26.48	7.74
BRISQUE [24]	28.6%		22.75	11.47
CNN_NR_IQA [17]	33.9%		28.78	5.44
CNN_NR_IQA_RE	67.9%	34.22	32.12	2.10
CNN_NR_IQA_NL [49]	75.0%		32.20	2.02
MANIQA [46]	55.4%		27.67	6.55
CNN_DCTT_RCPL	92.9%		33.60	0.62
MANIQA_DCTT_RCPL	**96.4%**		**34.22**	**0.0007**

D. RESULTS

1) PERFORMANCE OF TRANSFORMER-ASSISTED AND CNN-ASSISTED SYSTEMS

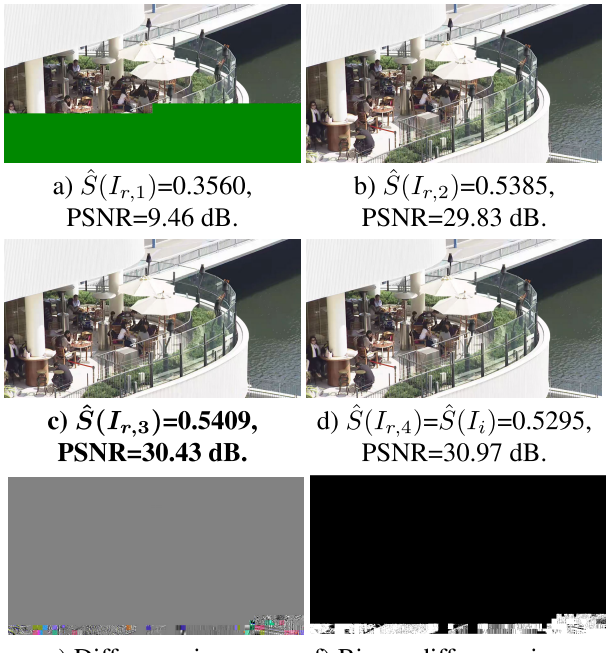
Tables 2 and 3 present the experimental results comparing our approach with state-of-the-art methods, using images encoded in *intra* and *inter* modes, respectively. The best results are highlighted in bold. PIQE [41], NIQE [25] and BRISQUE [24] use packaged functions directly within Matlab for inference simulations. CNN_NR_IQA denotes our use of the pre-trained model from the study [17], which assigns the same score to each patch of the Y component of the image (global score), and is tested with our database containing non-uniform distortions. CNN_NR_IQA_RE refers to our use of the retrained model from the study [17]. This model assigns a patch-level fidelity score (local score) to each patch of the R, G, and B components of the image. The CNN method with the suffix _NL indicates a configuration that incorporates the enhanced local normalization method described in [49]. The MANIQA [46] method indicates our use of the pre-trained Transformer model to test with our proposed databases. The methods suffixed with _DCTT and _RCPL denote the application of our new components proposed in this paper to Transformer-assisted and CNN-assisted systems, respectively.

Applying the proposed DCTT and RCPL components to both Transformer-assisted and CNN-assisted systems for intra-coded and inter-coded images results in improved accuracy and reduced quality differences compared to other pre-trained models. The benefits of using these newly proposed components in our Transformer-assisted system

and retraining on our proposed databases are evident, with precision increasing from 75% to 100% for intra-coded images and from 55.4% to 96.4% for inter-coded images. Similarly, the precision of the CNN-assisted system improves from 46.4% to 100% for intra-coded images and from 33.9% to 92.9% for inter-coded images.

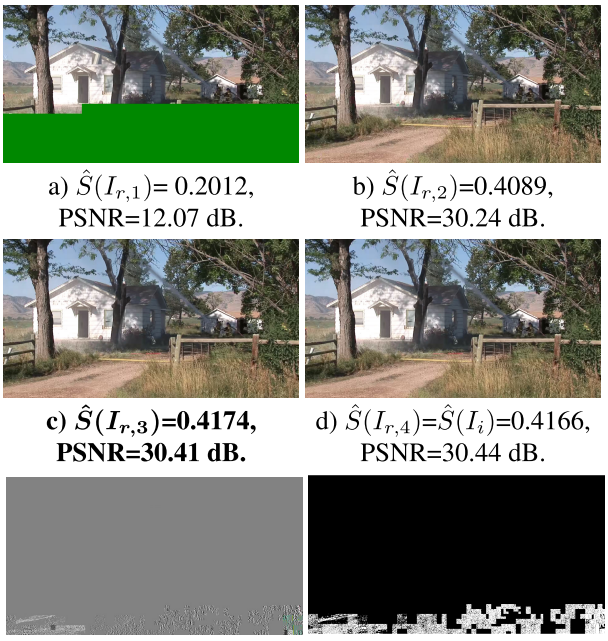
Figure 6 and 7 show examples of 'bad decisions' made by our CNN- and Transformer-assisted systems. In each figure, the candidate images are presented in (a) to (d). The selected candidate images are highlighted in bold. The intact image is depicted in (d). To generate the difference images in (e), each color channel is processed individually. First, the selected image is subtracted from the intact image, producing a difference image. Next, this difference is multiplied by 25 to amplify the variations. Lastly, 128 is added to each pixel value to adjust the brightness, centering the difference around a neutral gray. This process improves the visibility of differences between the images. A corresponding binary mask, highlighting the disparities, is displayed in (f) for both figures, with white representing the locations where differences are found. Despite the lower classification accuracy on inter-coded images in our simulations, we discovered that the incorrectly classified situations almost always chose the corrupted version with the highest ground truth PSNR score among the corrupted candidates.

In other words, the wrong candidate that our system selected is still close to the intact version. We observed that bit errors in inter frames do not cause as much quality loss as



a) $\hat{S}(I_{r,1})$=0.3560, PSNR=9.46 dB.
b) $\hat{S}(I_{r,2})$=0.5385, PSNR=29.83 dB.
c) $\hat{S}(I_{r,3})$=0.5409, PSNR=30.43 dB.
d) $\hat{S}(I_{r,4})$=$\hat{S}(I_i)$=0.5295, PSNR=30.97 dB.
e) Difference image.
f) Binary difference image.

FIGURE 6. Example of bad decision for the CNN_DCTT_RCPL configuration (inter-coded image). The system selects (c) while the intact version is (d). The difference image (e) and its binary version (f) are between the CNN model's choice and the intact version.



a) $\hat{S}(I_{r,1})$= 0.2012, PSNR=12.07 dB.
b) $\hat{S}(I_{r,2})$=0.4089, PSNR=30.24 dB.
c) $\hat{S}(I_{r,3})$=0.4174, PSNR=30.41 dB.
d) $\hat{S}(I_{r,4})$=$\hat{S}(I_i)$=0.4166, PSNR=30.44 dB.
e) Difference image.
f) Binary difference image.

FIGURE 7. Example of bad decision for the MANIQA_DCTT_RCPL configuration (inter-coded image). The system selects (c) while the intact version is (d). The difference image (e) and its binary version (f) are between the Transformer model's choice and the intact version.

in intra frames. This makes it more challenging to train the model on inter frames.

Compared to the CNN-assisted model, the Transformer-assisted model shows greater sensitivity in detecting small distortions in inter-coded images when using the new proposed components. These components enable our model to better learn the quality degradation caused by local distortions due to bit errors.

The experimental results demonstrate that integrating the DCTT and RCPL components significantly enhances model performance, validating their effectiveness in improving the quality assessment framework. The findings indicate that our proposed deep-learning assisted video list decoding framework is robust across various types of encoded images, effectively handling both intra- and inter-coded scenarios.

Our framework minimizes differences in quality among candidate videos, resulting in more consistent visual quality and aiding in the efficient selection of the "best" candidate video from the list generated by list decoding. In subsequent ablation studies, we will further detail the impact and enhancements brought by each new module in our proposed model.

2) IMPACT OF THE NEW COMPONENTS AND THEIR PARAMETERS

We now analyze the impact of the newly proposed components and their parameters.

a: PROPOSED NEW COMPONENTS

Tables 4 and 5 compare the CNN-assisted method from [49] to different simulation configurations incorporating the

TABLE 4. Performance on intra-coded images with the CNN model (QP=37). $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
CNN_RETRAINED [17]	96.4%	34.76	34.44	0.32
CNN_DCTT	**100%**		34.76	**0.00**
CNN_RCPL	98.2%		34.39	0.37
CNN_DCTT_RCPL	**100%**		34.76	**0.00**

TABLE 5. Performance on inter-coded images with the CNN model (QP=37). $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
CNN_RETRAINED [17]	67.9%	34.22	32.12	2.02
CNN_DCTT	73.2%		33.34	0.88
CNN_RCPL	41.1%		21.02	13.20
CNN_DCTT_RCPL	**92.9%**		33.60	**0.62**

proposed DCTT and RCPL components into the CNN-based system. The first row of each table, labeled CNN_RETRAINED, indicates that we retrained the original CNN model from [17] using our created database, where images initially used in YUV format are converted to RGB format for training and inference, without applying the proposed DCTT color space conversion or using RCPL during training. A method with the suffix _DCTT indicates a configuration that applies our DCTT color space conversion to our database (Eq. (1)). The suffix _RCPL indicates a configuration that applies our improved RCPL component as part of the loss function F with α initially set to 0.5 in Eq. (6).

For intra-coded images, applying the proposed DCTT color space conversion and RCPL consistently yields perfect results. For inter-coded images, the use of the proposed DCTT color space conversion significantly improves performance, with precision increasing from 67.9% to 73.2%. However, simply applying the RCPL component to the system does not result in any performance improvement. Notably, applying both new components to the simulation with inter-coded frames shows substantial improvement, with precision increasing from 67.9% to 92.9%. We notice that adding the proposed DCTT color space conversion to our datasets improves the simulation results for both intra-coded and inter-coded frames. This improvement aligns with the enhancement achieved by the local normalization algorithm proposed in [49] (see CNN_NR_IQA_NL in Tables 2 and 3).

Tables 6 and 7 compare the Transformer-assisted method from Section III-B to different simulation configurations incorporating the proposed DCTT and RCPL components. For intra-coded images, the proposed DCTT color space conversion consistently yields perfect results. For inter-coded images, incorporating the RCPL component significantly enhances performance, boosting precision from 92.9% to 96.4%. However, applying only the DCTT color space conversion does not lead to any performance improvement. In fact, it results in a decline. Notably, when both new components are applied to the simulation with inter-coded

TABLE 6. Performance on intra-coded images with the Transformer model (QP=37). $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
MANIQA_RETRAINED [46]	**100%**	34.76	34.76	**0.00**
MANIQA_DCTT	**100%**		34.76	**0.00**
MANIQA_RCPL	67.9%		29.78	4.98
MANIQA_DCTT_RCPL	**100%**		34.76	**0.00**

TABLE 7. Performance on inter-coded images with the Transformer model (QP=37). $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
MANIQA_RETRAINED [46]	92.9%	34.22	33.79	0.43
MANIQA_DCTT	78.6%		30.23	3.99
MANIQA_RCPL	**96.4%**		34.22	0.004
MANIQA_DCTT_RCPL	**96.4%**		34.22	**0.0007**

TABLE 8. Performance on inter-coded images with different coefficients in the loss function on the CNN model. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
CNN_DCTT_RCPL ($\alpha = 0$)	91.1%	34.22	32.77	1.45
CNN_DCTT_RCPL ($\alpha = 0.2$)	**98.2%**		**33.69**	**0.53**
CNN_DCTT_RCPL ($\alpha = 0.5$)	92.9%		33.60	0.62
CNN_DCTT_RCPL ($\alpha = 0.8$)	76.8%		33.41	0.81
CNN_DCTT_RCPL ($\alpha = 1.0$)	73.2%		33.34	0.88

TABLE 9. Performance on inter-coded images with different coefficients in the loss function on the Transformer model. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
MANIQA_DCTT_RCPL ($\alpha = 0.2$)	91.1%	34.22	34.21	0.01
MANIQA_DCTT_RCPL ($\alpha = 0.5$)	**96.4%**		34.22	**0.0007**
MANIQA_DCTT_RCPL ($\alpha = 1.0$)	78.6%		30.23	3.99

frames, there are obvious improvements, with precision rising from 92.9% to 96.4% and smaller differences between the average quality of the selected images and that of the intact versions.

b: COEFFICIENTS OF RCPL

The coefficients α of F_1 and F_2 in Eq. (6) can be varied. We change α to different values (0, 0.2, 0.5, 0.8 and 1) to observe the variations in performance with inter-coded frames on the CNN-assisted and the Transformer-assisted models, as shown in Tables 8 and 9, respectively.

For the CNN-assisted model, we can clearly see that a value near $\alpha = 0.2$ brings the best performance. According to Table 8, when $\alpha = 0.2$, the CNN-assisted system shows the highest accuracy of selecting the best candidate and the lowest difference between the average quality of the selected image and the average quality of the lossless image. This reflects the necessity of properly increasing the weight of F2 and shows the effective improvement of the model's performance by the new proposed loss function.

For the Transformer-assisted model, an α value near 0.5 provides the best overall performance, achieving the highest

TABLE 10. Performance on intra-coded images with our model by changing QP. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
CNN_DCTT_RCPL (QP=37)	100%	34.76	34.76	0.00
CNN_DCTT_RCPL (QP=22)	96.4%	43.62	43.62	0.002
MANIQA_DCTT_RCPL (QP=37)	100%	34.76	34.76	0.00
MANIQA_DCTT_RCPL (QP=22)	96.4%	43.62	43.62	0.002

TABLE 11. Performance on inter-coded images with our model by changing QP. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
CNN_DCTT_RCPL (QP=37)	92.9%	34.22	33.60	0.62
CNN_DCTT_RCPL (QP=22)	96.4%	42.61	42.49	0.12
MANIQA_DCTT_RCPL (QP=37)	96.4%	34.22	34.22	0.0007
MANIQA_DCTT_RCPL (QP=22)	100%	42.61	42.61	0.00

accuracy in selecting the best candidate and the smallest difference between the average quality of the selected image and the average quality of the lossless image.

3) PARAMETERS SENSITIVITY ANALYSIS

a: QUANTIZATION PARAMETER (QP)

The results reported in Tables 10 and 11 demonstrate the performance on intra-coded frames and inter-coded frames, respectively, when evaluating systems trained with QP = 37 on video encoded with QP settings of 37 and 22 in our database.

Encoding with a low QP value, such as 22, results in more residual information being transmitted and present at the decoder. This means that bit errors have less chances of damaging critical information such as motion vectors or coding modes. Consequently, when an error is introduced to the video packet, it has a lesser impact on the decoded picture quality. In contrast, a higher QP transmits less residual information, making errors more likely that a bit error will affect crucial elements of the compressed video. For intra-coded images, the system is not trained on this QP, so performance is worse since it did not learn its subtle artifacts with QP = 22 on the residual. For inter-coded images, it is important to note that although we use the same formula to introduce errors by flipping a single bit in video streams with different QP values, lower QP retains more redundant information. As a result, the bits flipped at QP = 22 may differ from those flipped at higher QP, introducing a degree of randomness in the distortion. There is a possibility that flipped bits at QP = 22 cause more noticeable local distortions, making it easier for our model to select the error-free version. However, overall, our model maintains good stability. For our small dataset, changing the QP does not significantly impact the model's performance.

b: PATCH SIZE

We also explored changing the patch size of our database to observe its impact on our CNN model. Tables 12 and 13 present the results with variable patch sizes for intra-coded frames and inter-coded frames, respectively.

TABLE 12. Performance on intra-coded images with the CNN model by changing the patch size. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
CNN_DCTT_RCPL (patch_size=32)	100%	34.76	34.76	0.00
CNN_DCTT_RCPL (patch_size=64)	100%		34.76	0.00
CNN_DCTT_RCPL (patch_size=128)	100%		34.76	0.00

TABLE 13. Performance on inter-coded images with the CNN model by changing the patch size. $\overline{S}_{\text{intact}}$, $\overline{S}_{\text{system}}$, and $\overline{S}_{\text{diff}}$ are in dB.

Methods	Accuracy	$\overline{S}_{\text{intact}}$	$\overline{S}_{\text{system}}$	$\overline{S}_{\text{diff}}$
CNN_DCTT_RCPL (patch_size=32)	94.6%	34.22	33.60	0.62
CNN_DCTT_RCPL (patch_size=64)	92.9%		33.60	0.62
CNN_DCTT_RCPL (patch_size=128)	85.7%		33.33	0.89

For intra-coded images, changing the patch size does not affect the final classification results, and our model maintains excellent accuracy consistently. For inter-frame coded images, varying the patch size leads to slight fluctuations in the final classification results, but these changes are not significant overall. The stability in accuracy is enabled by our proposed DCTT and RCPL components.

V. CONCLUSION AND PERSPECTIVES

In this paper, we presented a video list decoding framework using a Transformer-assisted method to identify the best candidate from the candidate list. This framework introduces a new NR IQA metric based on Transformer architecture to evaluate the quality of candidate videos, which can be applied to select the highest quality video from the candidate list generated by list decoding methods. The basic model is derived from [46], and we re-train the Transformer with super-patches supervised by NPFA scores, which better consider the local distortions in the horizontal and vertical neighbourhooding patches. We proposed a new DCTT color space conversion to distinguish between a well-received uniform patch and an erroneous patch that is uniform because it has been initialized to 0 by the decoder. Additionally, we applied the improved ranking-constrained penalty loss functions in the Transformer-assisted model, which enhance the model's performance by penalizing cases where a corrupted video obtains a higher score than its intact counterpart.

Our approach achieves remarkable decision accuracy, reaching 100% for intra-frame errors and 96.4% for inter-frame errors, which is a significant improvement over other evaluated methods. This is promising for real-time applications in error-prone network environments where maintaining high visual quality is critical.

The proposed deep-learning-assisted framework demonstrates high accuracy, though its computational requirements, particularly with the added DCTT and RCPL components, may pose challenges for real-time applications. Future work will focus on optimizing the framework to reduce computation time across the entire video decoding process. Additionally, we plan to expand the model's applicability to

a wider range of video compression standards and error configurations. One potential enhancement to our deep-learning system is to modify it to exploit the temporal correlations present in videos. The promising results obtained from this study lay the foundation for developing more robust and intelligent video transmission systems capable of delivering high-quality visual experiences even in challenging network conditions.

REFERENCES

[1] *Xiph.org Video Test Media [Derf's Collection]*. Accessed: Jan. 2023. [Online]. Available: https://media.xiph.org/video/derf/

[2] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard 802.11-2016, 2016, p. 3534.

[3] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, "LLR-based successive cancellation list decoding of polar codes," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5165–5179, Oct. 2015.

[4] S. Bosse, D. Maniry, T. Wiegand, and W. Samek, "A deep neural network for image quality assessment," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3773–3777.

[5] F. Bossen, "Common test conditions and software reference configurations," *JCTVC-L1100*, vol. 12, no. 7, p. 1, 2013.

[6] V. Boussard, S. Coulombe, F.-X. Coudoux, and P. Corlay, "Table-free multiple bit-error correction using the CRC syndrome," *IEEE Access*, vol. 8, pp. 102357–102372, 2020.

[7] V. Boussard, S. Coulombe, F.-X. Coudoux, P. Corlay, and A. Trioux, "CRC-based multi-error correction of H.265 encoded videos in wireless communications," in *Proc. Int. Conf. Vis. Commun. Image Process. (VCIP)*, Dec. 2021, pp. 1–5.

[8] F. Caron and S. Coulombe, "Video error correction using soft-output and hard-output maximum likelihood decoding applied to an H.264 baseline profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 7, pp. 1161–1174, Jul. 2015.

[9] D. Chen, Y. Wang, and W. Gao, "No-reference image quality assessment: An attention driven approach," *IEEE Trans. Image Process.*, vol. 29, pp. 6496–6506, 2020.

[10] M. Cheon, S.-J. Yoon, B. Kang, and J. Lee, "Perceptual image quality assessment with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 433–442.

[11] M. Collotta, G. Pau, T. Talty, and O. K. Tonguz, "Bluetooth 5: A concrete step forward toward the IoT," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 125–131, Jul. 2018.

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[13] F. Golaghazadeh, S. Coulombe, F.-X. Coudoux, and P. Corlay, "Checksum-filtered list decoding applied to H.264 and H.265 video error correction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1993–2006, Aug. 2018.

[14] S. A. Golestaneh, S. Dadsetan, and K. M. Kitani, "No-reference image quality assessment via transformers, relative ranking, and self-consistency," 2021, *arXiv:2108.06858*.

[15] A. Guichemerre, S. Coulombe, A. Trioux, F.-X. Coudoux, and P. Corlay, "Deep learning assisted quality ranking for list decoding of videos subject to transmission errors," in *Proc. 19th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Jun. 2023, pp. 135–142.

[16] T. Jokela and E. Lehtonen, "Reed-solomon decoding algorithms and their complexities at the DVB-H link-layer," in *Proc. 4th Int. Symp. Wireless Commun. Syst.*, Oct. 2007, pp. 752–756.

[17] L. Kang, P. Ye, Y. Li, and D. Doermann, "Convolutional neural networks for no-reference image quality assessment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1733–1740.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[19] J. Koloda, V. Sánchez, and A. M. Peinado, "Spatial error concealment based on edge visual clearness for image/video communication," *Circuits, Syst., Signal Process.*, vol. 32, no. 2, pp. 815–824, Apr. 2013.

[20] K. Kossi, S. Coulombe, C. Desrosiers, and G. Gagnon, "No-reference video quality assessment using distortion learning and temporal attention," *IEEE Access*, vol. 10, pp. 41010–41022, 2022.

[21] W.-Y. Kung, C.-S. Kim, and C.-C. J. Kuo, "Spatial and temporal error concealment techniques for video transmission over noisy channels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 7, pp. 789–803, Jul. 2006.

[22] A. A. Laghari, S. Shahid, R. Yadav, S. Karim, A. Khan, H. Li, and Y. Shoulin, "The state of art and review on video streaming," *J. High Speed Netw.*, vol. 29, no. 3, pp. 211–236, Aug. 2023.

[23] H. Lin, V. Hosu, and D. Saupe, "KADID-10k: A large-scale artificially distorted IQA database," in *Proc. 11th Int. Conf. Quality Multimedia Exper. (QoMEX)*, Jun. 2019, pp. 1–3.

[24] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4695–4708, Dec. 2012.

[25] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a 'completely blind' image quality analyzer," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 209–212, Mar. 2013.

[26] Q. Peng, T. Yang, and C. Zhu, "Block-based temporal error concealment for video packet using motion vector extrapolation," in *Proc. IEEE Int. Conf. Commun., Circuits Syst. West Sino Expo.*, vol. 1, Jun. 2002, pp. 10–14.

[27] M. H. Pinson, "The consumer digital video library [best of the web]," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 172–174, Jul. 2013.

[28] N. Ponomarenko, O. Ieremeiev, V. Lukin, K. Egiazarian, L. Jin, J. Astola, B. Vozel, K. Chehdi, M. Carli, F. Battisti, and C.-C. J. Kuo, "Color image database TID2013: Peculiarities and preliminary results," in *Proc. Eur. Workshop Vis. Inf. Process. (EUVIP)*, 2013, pp. 106–111.

[29] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, "ImageNet-21K pretraining for the masses," 2021, *arXiv:2104.10972*.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[31] U. Sara, M. Akter, and M. S. Uddin, "Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study," *J. Comput. Commun.*, vol. 7, no. 3, pp. 8–18, 2019.

[32] H. Sheikh. (2005). *Live Image Quality Assessment Database Release 2*. [Online]. Available: http://live.ece.utexas.edu/research/quality

[33] S. Shirani, F. Kossentini, and R. Ward, "Error concealment methods, a comparative study," in *Proc. Eng. Solutions Next Millennium. IEEE Can. Conf. Electr. Comput. Eng.*, vol. 2, May 1999, pp. 835–840.

[34] G. J. Sullivan and T. Wiegand, "Video compression-from concepts to the H.264/AVC standard," *Proc. IEEE*, vol. 93, no. 1, pp. 18–31, Jan. 2005.

[35] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[36] *Cisco Visual Networking Index: Forecast and Methodology*, document 1465272001663118, Cisco Syst., Jun. 2016.

[37] V. Sze, M. Budagavi, and G. J. Sullivan, *High Efficiency Video Coding (HEVC): Algorithms Architectures*. Cham, Switzerland: Springer, 2014, doi: 10.1007/978-3-319-06895-4.

[38] G. Tudavekar, S. S. Saraf, and S. R. Patil, "Spatio-temporal inference transformer network for video inpainting," *Int. J. Image Graph.*, vol. 23, no. 1, Jan. 2023, Art. no. 2350007.

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inform. Process. Syst.*, 2017, pp. 5998–6008.

[40] M. T. Vega, D. C. Mocanu, J. Famaey, S. Stavrou, and A. Liotta, "Deep learning for quality assessment in live video streaming," *IEEE Signal Process. Lett.*, vol. 24, no. 6, pp. 736–740, Jun. 2017.

[41] N. Venkatanath, D. Praneeth, M. C. Bh, S. S. Channappayya, and S. S. Medasani, "Blind image quality evaluation using perception based features," in *Proc. 21st Nat. Conf. Commun. (NCC)*, Feb. 2015, pp. 1–6.

[42] H. Wang, W. Gan, S. Hu, J. Y. Lin, L. Jin, L. Song, P. Wang, I. Katsavounidis, A. Aaron, and C.-C. J. Kuo, "MCL-JCV: A JND-based H.264/AVC video quality assessment dataset," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 1509–1513.

[43] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.
[44] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.
[45] K. Xu, L. Liao, J. Xiao, C. Chen, H. Wu, Q. Yan, and W. Lin, "Local distortion aware efficient transformer adaptation for image quality assessment," 2023, *arXiv:2308.12001*.
[46] S. Yang, T. Wu, S. Shi, S. Lao, Y. Gong, M. Cao, J. Wang, and Y. Yang, "MANIQA: Multi-dimension attention network for no-reference image quality assessment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 1191–1200.
[47] J. You and J. Korhonen, "Transformer for image quality assessment," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 1389–1393.
[48] W. Zhang, K. Ma, J. Yan, D. Deng, and Z. Wang, "Blind image quality assessment using a deep bilinear convolutional neural network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 1, pp. 36–47, Jan. 2020.
[49] Y. Zhang, S. Coulombe, F.-X. Coudoux, A. Trioux, and P. Corlay, "Optimisation du décodage par liste de vidéos corrompues basée surune architecture CNN," in *Proc. 22éme Éd. De La Conf. Compress. Et Représent. Des. Signaux Audiovisuels*, 2023, p. 4. [Online]. Available: https://hal.science/hal-04246635

YUJING ZHANG received the M.Eng. degree in electronics and digital technologies from École Polytechnique de l'Université de Nantes, Nantes, France, in 2020. She is currently pursuing the Ph.D. degree with the Department of Software and IT Engineering, École de technologie supérieure, Montreal, QC, Canada, and the Département d'Opto-Acousto-Électronique (DOAE), Institut d'Électronique de Microélectronique et de Nanotechnologie (IEMN), UMR 8520, Valenciennes. Her current research interests include video transmission, image quality assessment, and deep learning.

STÉPHANE COULOMBE (Senior Member, IEEE) received the B.Eng. degree in electrical engineering from École Polytechnique de Montréal, Canada, in 1991, and the Ph.D. degree in telecommunications (image processing) from INRS-Telecommunications, Montreal, in 1996.

From 1997 to 1999, he was with the Nortel Wireless Network Group, Montreal. From 1999 to 2004, he was a Research Engineer at the Nokia Research Center, Dallas, TX, USA, and the Program Manager at the Audiovisual Systems Laboratory. In 2004, he joined École de technologie supérieure (ÉTS is a constituent of the Université du Québec network), where he is currently a Professor with the Department of Software and IT Engineering. From 2009 to 2018, he held the Vantrix Industrial Research Chair in Video Optimization. His research interests include video processing, compression, communication (transport), and systems with a recent focus on immersive video and machine learning for video applications.

FRANÇOIS-XAVIER COUDOUX (Senior Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering from Université Polytechnique Hauts-de-France, Valenciennes, France, in 1991 and 1994, respectively. Since 2004, he has been a Professor with the Département d'Opto-Acousto-Électronique (DOAE), Institut d'Électronique de Microélectronique et de Nanotechnologie (IEMN), UMR 8520, Valenciennes. His research interests include telecommunications, multimedia delivery over wired and wireless networks, image quality, and adaptive video processing.

ALEXIS GUICHEMERRE received the Engineering degree in applied mathematics from EISTI (now CYTECH), the first master's degree in high-performance computing from the University of A Coruña, in 2019, and the second master's degree in information technology from École de technologie supérieure, in 2022, where he is currently pursuing the Ph.D. degree with the Department of System Engineering. His research interests include machine learning, image quality but, most notably, source-free domain adaptation, and weakly supervised object localization in a medical context.

PATRICK CORLAY received the Ph.D. degree from Université Polytechnique Hauts-de-France, Valenciennes, France, in 1994. Since 2016, he has been a Professor with the Département d'Opto-Acousto-Électronique (DOAE), Institut d'Électronique de Microélectronique et de Nanotechnologie (IEMN), UMR 8520, France. His current research interests include telecommunications, multimedia delivery over wired and wireless networks, and optimal quality of service for video transmission.