published in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 3898-3910, Nov. 2020

PEA265: Perceptual Assessment of Video Compression Artifacts

Liqun Lin, Shiqi Yu, Liping Zhou, Weiling Chen, *Member, IEEE*, Tiesong Zhao, *Senior Member, IEEE* and Zhou Wang, *Fellow, IEEE*

***Abstract*—The most widely used video encoders share a common hybrid coding framework that includes block-based motion estimation/compensation and block-based transform coding. Despite their high coding efficiency, the encoded videos often exhibit visually annoying artifacts, denoted as Perceivable Encoding Artifacts (PEAs), which significantly degrade the visual Quality-of-Experience (QoE) of end users. To monitor and improve visual QoE, it is crucial to develop subjective and objective measures that can identify and quantify various types of PEAs. In this work, we make the first attempt to build a large-scale subject-labeled database composed of H.265/HEVC compressed videos containing various PEAs. The database, namely the PEA265, includes 4 types of spatial PEAs (*i.e.* blurring, blocking, ringing and color bleeding) and 2 types of temporal PEAs (*i.e.* flickering and floating). Each containing at least 60,000 image or video patches with positive and negative labels. Based on the PEA265 database, we develop and optimize Convolutional Neural Networks (CNNs) to objectively recognize different types of PEAs. Experiments show that our architecture is capable of identifying the 6 types of PEAs with an accuracy over 86%. To further demonstrate its application, we explore the relationship between collected PEA intensities and subjective quality scores of compressed videos. A quality metric is consequently proposed with superior performance in terms of correlation to Mean Opinion Score (MOS) values. We believe that the PEA265 database and our findings will benefit the future development of video quality assessment methods and perceptually motivated video encoders.**

***Index Terms*—Video coding, video compression, video quality assessment, perceptual encoding artifacts, H.265/HEVC.**

I. Introduction

THE last decade has witnessed a booming of High Definition (HD)/Ultra HD (UHD) and 3D/360-degree videos due to the rapid developments of video capturing, transmission and display technologies. According to Cisco Visual Networking Index (VNI) [1], video content has taken over 2/3 bandwidth of current broadband and mobile networks, and will grow to 80%-90% in the visible future. To meet such a demand, it is necessary to improve network bandwidth and maximize video quality under a limited bitrate or bandwidth constraint, where the latter is generally achieved by lossy video coding technologies.

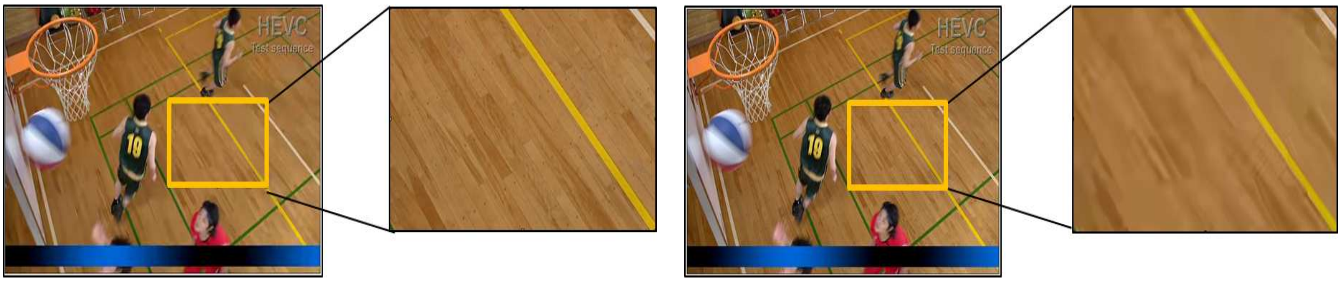
The widely used video coding schemes are lossy for two reasons. Firstly, Shannon's theorem sets the limit of lossless coding, which cannot fulfill the practical needs on video compression. Secondly, the Human Vision System (HVS) [2] is not uniformly sensitive to visual signals at all frequencies, which allows to suppress certain frequencies with negligible loss of perceptual quality. The state-of-the-art video coding schemes, such as H.264 Advanced Video Coding (H.264/AVC) [3], H.265 High Efficiency Video Coding (H.265/HEVC) [4], Versatile Video Coding (VVC) [5], Google VP8/VP9 [6], [7], China's Audio-Video coding Standards (AVS/AVS2) [8], [9], adopt the conventional hybrid video coding structure. This infrastructure, originated from 1980s [10], consists of a group of standard procedures including intra-frame prediction, inter-frame motion estimation and compensation, followed by spatial transmission, quantization and entropy coding. To facilitate these functions in videos of large sizes, the encoder further divides the frames into slices and coding units. Thereby, when the bitrate is not sufficially high, the compressed video encompasses various types of information loss within and across blocks, slices and units, resulting in visually unnatural structure impairments or perceptual artifacts [11]. These Perceivable Encoding Artifacts (PEAs) greatly degrade the visual Quality-of-Experience (QoE) of users [12].

Recent developments have greatly put forward the 4K/8K era and user-centric video coding and delivery has become ever important [13]. Meanwhile, the advancements of computing and networking technologies have enabled deep investigations on recognition and quantification of video artifacts. Gong *et al.* [14] presented a visual-masking-based method to estimate regions with temporal pumping artifacts in video coding. In [15], ringing artifacts were detected and suppressed with sparse approach in image coding. Todd *et al.* [16] presented an approach to detect different types of artifacts introduced by video coding, processing and delivery, in which the video coding artifacts were categorized as the same class for analysis. To eliminate the negative effect of artifacts, great efforts have been contributed to improve image coding [15], [17]–[19] and reduce the blocking artifact in video coding [3]–[9]. Recently, deep learning techniques [20], especially Convolutional Neural Network (CNN) [21], have demonstrated their promise in improving video coding performance [22], [23]. These techniques have also been incorporated in the in-loop filters [24], [25] or post-processing [26], [27] of video encoder to eliminate the blocking and blurring artifacts for an improved visual

This research is supported by the National Natural Science Foundation of China (No. 61671152, 61901119). It is also supported by the Natural Science Foundation of Fujian Province, China (No.2017J01757).

The authors are with the Fujian Key Lab for Intelligent Processing and Wireless Transmission of Media Information, College of Physics and Information Engineering, Fuzhou University, Fuzhou, Fujian 350116, China (E-mails:{lin_liqun, n171120077, n181120080, weiling.chen, t.zhao}@fzu.edu.cn).

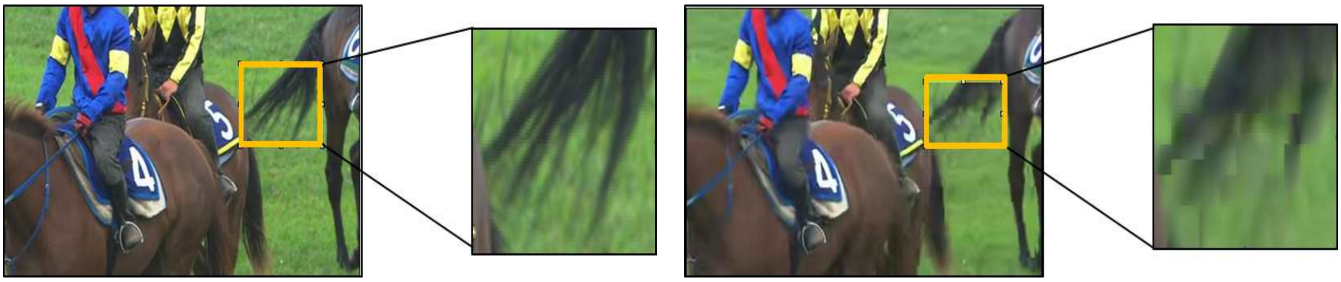
Z. Wang is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada (E-mail: zhou.wang@uwaterloo.ca.)



(a) Reference frame

(b) Compressed frame with blurring artifact

Fig. 1: An example of blurring artifact.



(a) Reference frame

(b) Compressed frame with blocking artifact

Fig. 2: An example of blocking artifact.

quality. In [28], we also utilized the Generative Adversarial Network (GAN) to develop a PEA removal strategy in the post-processing of video coding. On the other hand, the coding artifacts are also utilized to evaluate the compressed video quality besides of conventional quality metrics such as Sum of Absolute Differences (SAD), Sum of Squared Errors (SSE), Peak-Signal-to-Noise Ratio (PSNR), and Structural SIMilarity (SSIM) index [29]. In [30], the compression artifact was observed to have significant impacts on H.264/AVC-compressed video quality, especially for blocking, blurring and color bleeding. Recently, the blocking and blurring artifacts have been exploited to develop no-reference video quality models [31]–[33].

The above great efforts focus on the most common PEAs including blocking and blurring. In [34], the authors elaborated the features and possible reasons of diversified PEAs and provided a detailed taxonomy of PEAs beyond the blocking and blurring artifact. To further analyze these PEAs, high-level processing with deep neural works is strongly required. However, a large-scale dataset is a necessity to develop deep infrastructure for PEA recognition. To address this issue, we have developed both a PEA database and a CNN-based recognition approach. The contributions of this work are summarized as follows:

(1) A large-scale database of compressed videos with subjectively labeled PEAs. 6 types of PEAs are selected for further labelling. We utilize the H.265/HEVC to encode a group of standard sequences and recruit users to mark all types of PEAs. Finally, we cut the marked sequences into image/video patches with positive and negative PEA labels. In total, there are 6 typical PEAs and at least 60,000 positive or negative labels are given for each type of PEA.

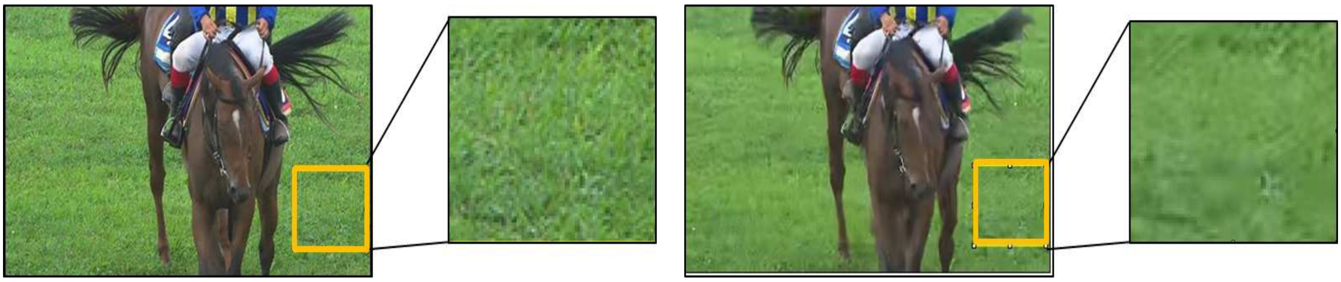
(2) An objective PEA recognition approach based on CNN. For each type of PEA, we construct and compare deep CNNs to identity whether it exists in an image/video patch. The implemented Dense Convolutional Network (DenseNet) [35] and ResNeXt [36] achieve the state-of-the-art performances in terms of PEA recognition.

(3) An objective quality metric based on PEA recognition. By summarizing PEA intensities, we obtain an overall measure on a compressed video, which helps characterize the subjective annoyance of PEAs caused by video coding. The opposite of this measure formulates a quality with high correlation to subjective scoring, which is superior to several existing video quality assessment algorithms.

The rest of the paper is organized as follows. In Section II, we discuss diversified PEAs in H.265/HEVC and select 6 types of PEAs for our database. In Section III, we elaborate the details of our subjective database including video sequence preparation, subjective testing and data processing. Section IV presents our deep-learning-based PEA recognition. Section V introduces the PEA-based measurement and explores its application in video quality analysis. Finally, Section VI concludes the paper.

II. PEA Classification

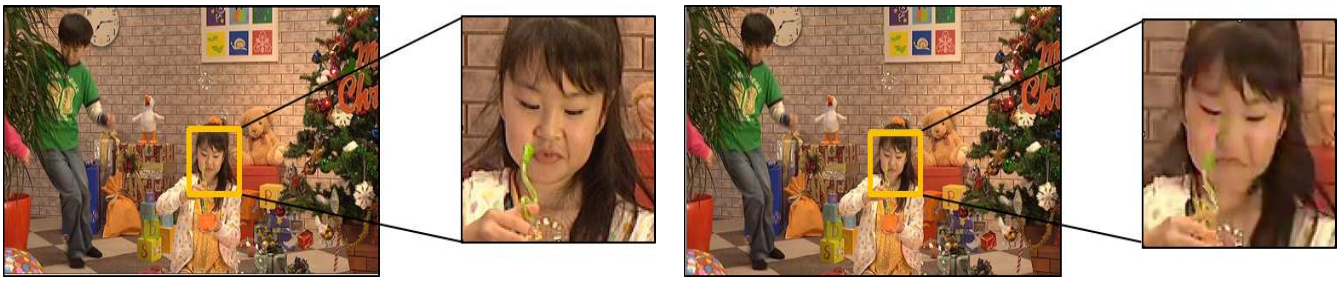
In this section, we review the PEA classification in [34] and select typical PEAs to develop our subjective database. According to this work, the PEAs are classified into spatial and temporal artifacts, where spatial artifacts include blurring, blocking, color bleeding, ringing and basis pattern effect; temporal artifacts include floating, jerkiness and flickering. In this work, we select blurring, blocking, color bleeding, ringing of spatial artifacts and floating, flickering of temporal artifacts



(a) Reference frame

(b) Compressed frame with ringing artifact

Fig. 3: An example of ringing artifact.



(a) Reference frame

(b) Compressed frame with color bleeding artifact

Fig. 4: An example of color bleeding artifact.

in the development of our database. Basis pattern effect and jerkiness artifacts are excluded because: 1) the basis pattern effect has similar visual appearance and has similar origin to the ringing effect; 2) the jerkiness artifacts are caused by image capturing factors such as frame rate instead of compression. We summarize the characteristics and plausible reasons of 6 typical types of PEAs as follows.

A. *Spatial Artifacts*

Block-based video coding schemes create various spatial artifacts due to block partitioning and quantization. The spatial artifacts, with different visual appearances, can be identified without temporal reference.

1) Blurring: Aiming at a higher compression ratio, the HEVC encoder quantizes transformed residuals discrepantly. When video signals are reconstructed, high frequency energy may be severely lost, which may lead to visual blur. Perceptually, blurring usually appears as the loss of spatial details or sharpness of edges or texture regions in an image. An example is shown in the marked rectangular region in Fig. 1 (b), which demonstrates the spatial loss of the basketball field.

2) Blocking: The HEVC encoder is block-based, and all compression processes are performed within non-overlapped blocks. This often results in false discontinuities across block boundaries. The visual appearance of blocking may be different subject to the region of visual discontinuities. In Fig. 2 (b), a blocking example of the horse tail is highlighted in the marked rectangular region.

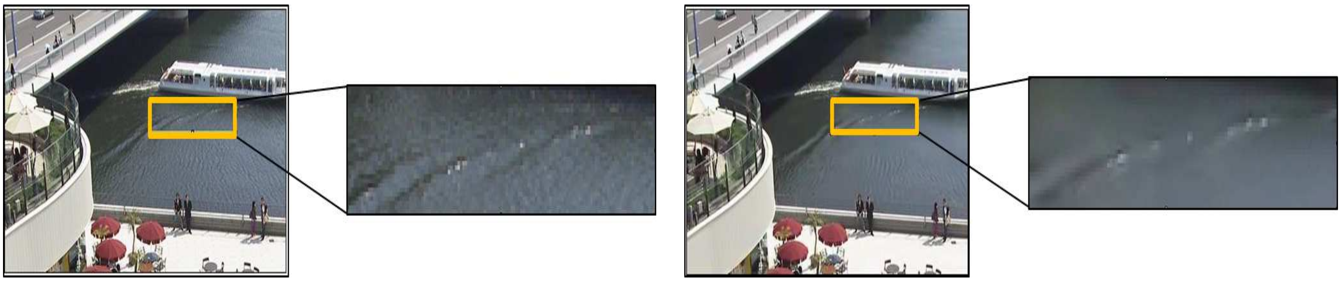
3) Ringing: Ringing is caused by the coarse quantization of high frequency components. When the high frequency component of oscillating structure has a quantization error, the pseudo structure may appear near strong edges (high contrast), which manifests artificial wave-like or ripple structures, denoted as ringing. A ringing example is given in the marked rectangular region in Fig. 3 (b).

4) Color bleeding: The chromaticity information is coarsely quantized to cause color bleeding. It is related to the presence of strong chroma variations in the compressed images leading to false color edges. It may be a result of inconsistent image rendering across the luminance and chromatic channels. A color bleeding example is provided in the marked rectangular region in Fig. 4 (b), which exhibits chromatic distortion and additional inconsistent color spreading in the rendering result.

B. *Temporal Artifacts*

Temporal artifacts are manifested as temporal information loss, and can be identified during video playback.

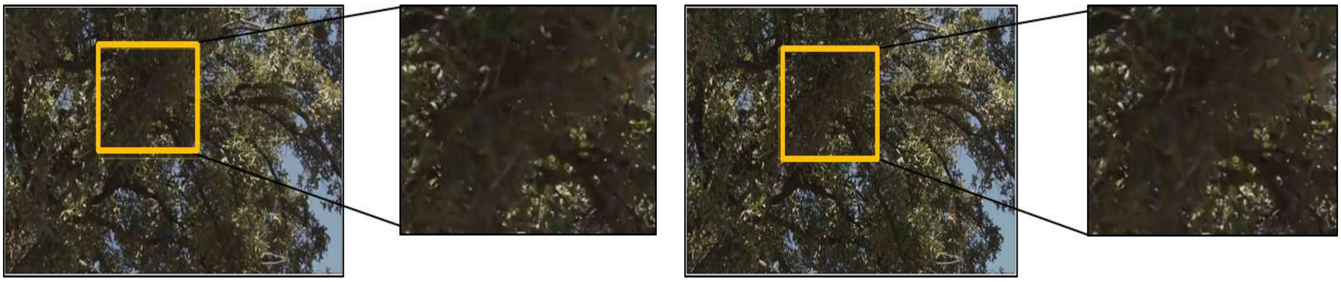
1) Flickering: Flickering is usually frequent brightness or color changes along the time dimension. There are different kinds of flickering including mosquito noise, fine-granularity flickering and coarse-granularity flickering. Mosquito noise is high frequency distortion and the embodiment of the coding effect in time domain. It moves together with the objects like mosquitoes flying around. It may be caused by the mismatch prediction error of the ringing effect and the motion compensation. The most likely cause of coarse-granulating blinking may be luminance variations across Group-Of-Pictures (GOPs). Fine-granularity flickering may be produced by slow motion and blocking effect. An example is given in the marked rectangular region in Fig. 5 (b). Frequent luminance changes on the surface of the water produce flickering artifacts.



(a) Reference frame

(b) Compressed frame with flickering artifact

Fig. 5: An example of flickering artifact.



(a) Reference frame

(b) Compressed frame with floating artifact

Fig. 6: An example of floating artifact.

2) Floating: Floating refers to the appearance of illusory movements in certain areas rather than their surrounding environment. Visually, these regions create a strong illusion as if they were floating on top of the surrounding background. Most often, a scene with a large textured area such as water or trees is captured with cameras moving slowly. The floating artifacts may be due to the skip mode in video coding, which simply copies a block from one frame to another without updating the image details further. Fig. 6 (b) gives a floating example. Visually these regions create a strong illusion as if they were floating on top of the leaves.

III. The PEA265 Database

The development of the PEA265 database is composed of four steps: preparation of test video sequences, subjective PEA region identification, patch labeling, and formation of the PEA265 database.

A. Test Video Sequences

We develop the PEA265 database using the popular video encoder H.265/HEVC. To examine the performance of video encoders, a standard encoding procedure, namely the Common Test Conditions (CTC) [37], was recommended by the Joint Collaborative Team on Video Coding (JCT-VC). The CTC recommends a set of standard video sequences as summarized in Table I, which are set as the test video sequences of our database. These sequences are further categorized by classes, according to their definitions, frame rates and contents. By testing video sequences in all classes, we attempt to cover enough types of PEAs in our database.

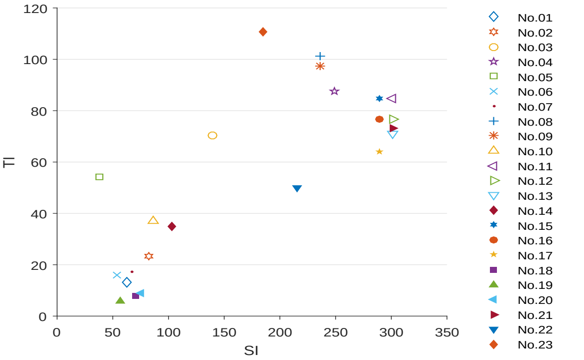
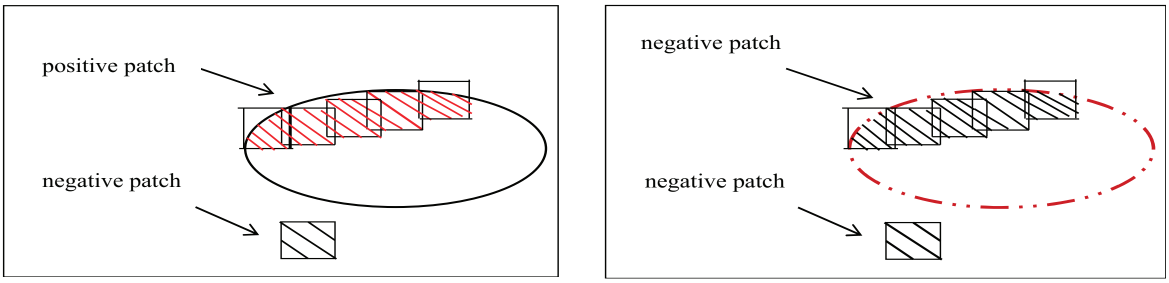


Fig. 7: The distributions of SI and TI for all CTC sequences.

To further examine the representativeness of these video sequences, we also calculate their Spatial Information (SI) and Temporal Information (TI) values. The SI and TI were defined in ITU-T P. 910 [38] to depict the maximal spatial gradient intensity and maximal temporal discontinuity of video contents, respectively. From Fig. 7, the selected sequences cover a vast region of SI and TI values, *e.g.* No. 1 video is relatively simple in spatial and temporal domains, while the No. 16 video is highly complex in both domains. Therefore, the 23 standard video sequences are sufficiently representative and meet the requirements of the database construction.

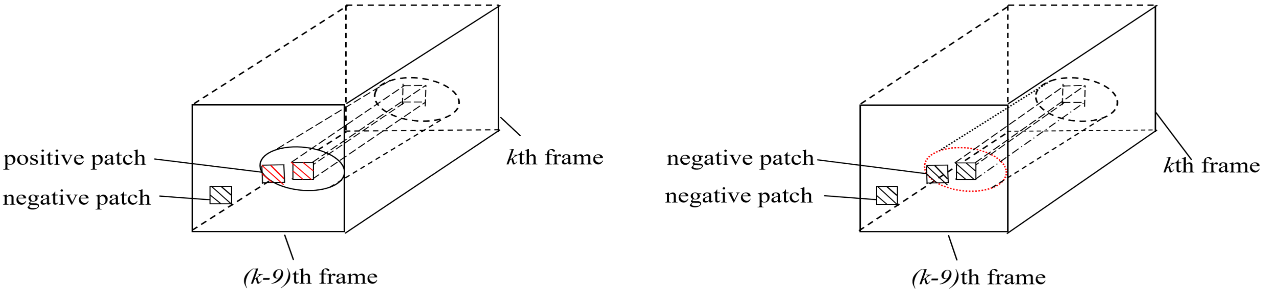
The above video sequences are sampled with YUV4:2:0 format and further compressed video with H.265 video encoder under the settings designated by CTC. Four types of coding structures, including All Intra (AI), Random Access (RA), Low Delay (LD) and Low Delay P (LP) are employed to show their effects on compression and PEAs. It is noted that parts



(a) Patch labeling in a compressed video frame

(b) Patch labeling in corresponding reference video frame

Fig. 8: Positive/negative patch labeling for spatial PEAs.



(a) Patch labeling in compressed video frames

(b) Patch labeling in corresponding reference video frames

Fig. 9: Positive/negative patch labeling for temporal PEAs.

TABLE I: Testing sequences.

No	Class	Sequence (Resolution)	Frames	Frame rate	No	Class	Sequence (Resolution)	Frames	Frame rate
1	A	*Traffic* (2560×1600)	150	30fps	13	C	*BasketballDrill* (832×480)	500	50fps
2	A	*PeopleOnStreet* (2560×1600)	150	30fps	14	D	*RaceHorses* (416×240)	300	30fps
3	A	*NebutaFestival* (2560×1600)	300	60fps	15	D	*BQSquare* (416×240)	600	60fps
4	A	*SteamLocomotive* (2560×1600)	300	60fps	16	D	*BlowingBubbles* (416×240)	500	50fps
5	B	*Kimono* (1920×1080)	240	24fps	17	D	*BasketballPass* (416×240)	500	50fps
6	B	*ParkScene* (1920×1080)	240	24fps	18	E	*FourPeople* (1280×720)	600	60fps
7	B	*Cactus* (1920×1080)	500	50fps	19	E	*Johnny* (1280×720)	600	60fps
8	B	*BQTerrace* (1920×1080)	600	60fps	20	E	*KristenAndSara* (1280×720)	600	60fps
9	B	*BasketballDrive* (1920×1080)	500	50fps	21	F	*BaskeballDrillText* (832×480)	500	50fps
10	C	*RaceHorses* (832×480)	300	30fps	22	F	*SlideEditing* (1280×720)	300	30fps
11	C	*BQMall* (832×480)	600	60fps	23	F	*SlideShow* (1280×720)	500	20fps
12	C	*PartyScene* (832×480)	500	50fps					

of structures may not be supported for some video sequences, subject to the CTC configurations. For each supported pair of sequence and coding structure, four Quantization parameter (Qp) values of 22, 27, 32 and 37 are utilized to show the visual results under different information losses. For consistency, the output bit depths of all videos are set to 8. In total, there are 324 outputs with different contents, resolutions, coding structures and/or Qps.

B. Subjective PEA Region Identification

In order to identify all PEAs, we ask subjects (*i.e.* testees) to label all video sequences. Our testing procedure follows the ITU-R BT.500 [39] document with two phases. In the pre-training phase, all subjects are told about our testing procedures and trained to identify PEAs. In the formal-testing phase, all subjects are asked to watch these sequences and circle PEA regions. The test sequences are presented in random order. All HD/UHD are displayed on a 5K screen while other sequences are watched on an HD screen. Neither zooming nor sampling operation is involved to avoid additional artifacts. Mid-term breaks are set during the formal-testing to avoid visual fatigue. 30 subjects participated in the subjective experiment, including 14 males and 16 females-aged between 20 to 22. We divide the 30 subjects into 6 groups in order to respectively mark the six types of PEAs. In each group, 5 subjects are asked to go through all sequences to circle out the same type of PEA with an ellipse shape. A region is marked by either subject is considered a PEA region. To avoid mislabelling, a tutor is responsible to double-check the results of all subjects and exclude incorrect labels. We are pleased to observe a promisingly high accuracy of labelling.

C. Patch Labeling

During the subjective test, PEA regions were marked and saved in binary format. Based on the marks, we derived positive and negative patches in rectangular or cuboid shapes whilst excluding ambiguous labelling.

1) Spatial artifacts: For spatial artifacts, we label the patches by a sliding window of 72×72. In a compressed video, if at least half of the pixels within the sliding window belong to this circled region, it is labeled as positive; otherwise negative. Patches belonging to the corresponding frame of uncompressed video are randomly selected and categorized as negative, whether or not they are co-located within the circled region. The ratio between the numbers of the two types of negative patches is 1:2. The labeling process is illustrated in Fig. 8.

2) Temporal artifacts: Temporal PEAs appear in a group of successive video frames. Therefore, a few successive frames are extracted when a subject pauses video playback and marks a temporal artifact region. With a tradeoff between the minimum reaction delay (*i.e.* the human reaction speed sets a delay between the first glance of temporal PEA and the PEA marking) and the maximum video fragment (*i.e.* a unified and small size of video patch for easy processing of deep neural works), we utilize the current and its 9 previous frames to formulate the temporal PEA patch. Considering the temporal PEA lasts for frames, the most probable temporal PEA region will be involved. After that, the video fragment is further checked by a spatial sliding window of 72×72: if at least half of the pixels in this window are within the circled region, then the corresponding cuboid is labeled as positive, otherwise negative. Similar to spatial artifacts, negative temporal patches are also obtained from co-located region in the uncompressed sequences. This process is illustrated in Fig. 9.

In summary, considering the temporal PEAs are only visible when the frames are displayed, it is impossible to ask the human testees to track them frame-by-frame and pixel-by-pixel. Instead, we utilize a careful patch labeling strategy as mentioned above to ensure that only the most probable positive and negative clips/patches are included in our database. The ambiguous data are naturally excluded to eliminate outliers in our database.

D. Formation of the PEA265 Database

After marking all types of PEAs, we segmented the labeled sequences into image/video patches with positive and negative PEA labels. A manually examination was performed for random samples to ensure the quality of patch labelling. In total, the PEA265 database covers 6 types PEAs that includes 4 types of spatial PEAs (blurring, blocking, ringing and color bleeding) and 2 types of temporal PEAs (flickering and floating). As shown in Table II, each type of PEAs contains at least 60,000 image or video patches with positive and negative labels, respectively. All types of PEAs are stored in binary format and of size 72×72. Each PEA patch is indexed by its video name, frame number, and coordinate position.

TABLE II: The number of samples in PEA265.

Types	Positive samples	Negative samples
Blocking	26750	41600
Blurring	35268	42336
Color bleeding	27033	33816
Ringing	26325	41333
Flickering	26783	37250
Floating	27000	35668

IV. CNN-based PEA Recognition

In this section, we explore the utility of our PEA265 database by developing CNN-based PEA recognition methods. Due to the high-level syntax in the characteristics of PEAs, it is preferable to utilize deep learning for PEA detection. Our database provides training and testing sets for this task.

A. The Proposed PEA Recognition Models

We exploit the popular CNN architectures of DenseNet and ResNeXt to the detection and identification of PEAs. These architectures are also further improved to aim at a high recognition accuracy.

1) ResNeXt network: As an improved version of popular Residual Network (ResNet) [40], the ResNeXt was proposed by He *et al.* in 2017 [36]. Based on a repeated topology of blocks, this network architecture successfully increases the accuracy of image classification with reduced complexity in hyper-parameters. Due to its advantage, the ResNeXt has been widely applied in processing of various types of images, including face, gesture and medical images.

In this work, we have tuned the parameters of ResNeXt to adapt to the PEA recognition problem. Besides, we also optimize the ResNeXt architecture to identify various PEAs with complex features. This leads to a ResNeXt for PEA Recognition (ResNeXt-PR) model. Squeeze and Excitation (SE) Block [41] is embedded to obtain decent feature extraction. Batch Normalization (BN) [42] is deployed after the convolutions to exclude the impacts of internal covariance shifts. Rectified Linear Unit (ReLU) is performed right after each BN. Finally, the output of ResNeXt-PR is expressed as follows:

$$y_l = h(x_l) + F(x_l, W_l), \tag{1}$$

$$X_{l+1} = f(y_l), \tag{2}$$

where x_l is the input of the lth residual module, $h(x_l)$ refers to an identity map, W_l denotes a set of weights associated with l residual modules. $F(x_l, W_l)$ represents the residual function. $f(y_l)$ is the ReLU activation function. Therefore, $h(x_l)$ and $f(y_l)$ are all equally mapped in ResNeXt-PR, namely $h(x_l) = x_l$ and $f(y_l) = y_l$. Then, in the forward direction of training and backpropagation phase, signals can be passed directly from one unit to another, which simplifies the training process.

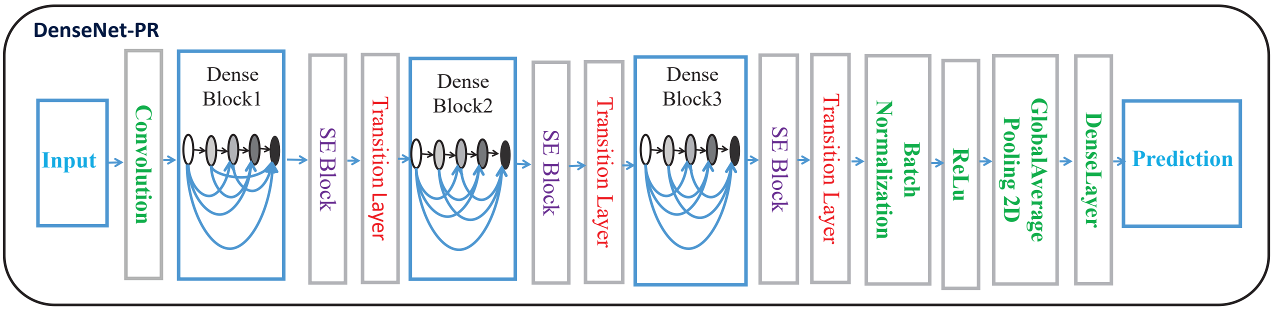


Fig. 10: DenseNet for PEA Recognition (DenseNet-PR).

2) DenseNet network: Another deep CNN architecture examined in this work is the popular DenseNet, which was proposed by Huang *et al.* in 2017 [35]. With a similar basic idea to ResNet, the new network aims to build dense connection between layers in a feed-forward fashion. It strengthens the feature propagation while alleviating the vanishing-gradients. In the framework, each layer obtains additional inputs from all preceding layers and passes on its own feature-maps to all subsequent layers. Consequently, the lth layer receives the feature maps of all preceding layers, x_0,...,x_{l-1}, as input:

$$X_l = H_l\left([x_0, x_1, ..., x_{l-1}]\right), \tag{3}$$

where $H_l(.)$ is defined as a composite function of three consecutive operations: BN, ReLU and a 3×3 convolution. $[x_0, x_1, ..., x_{l-1}]$ refers to the concatenation of the feature-maps produced in layers $0, ..., l-1$.

The DenseNet is also optimized to adapt to our PEA recognition problem, resulting in a DenseNet for PEA Recognition (DenseNet-PR) in Fig. 10. First of all, we introduce a deep separable convolution and SE block to the original bottleneck. To learn the characteristics of feature channel in a deeper level, the 3×3 standard convolution in the Dense Block is split into a 3×3 and a 1×1 pointwise convolution. Then, we embed an SE Block between each Dense Block and the transition layer. The squeeze and excitation operations enhance the important features of the training samples. It also reuses important features of the transition layer to increase the recognition accuracy. The transition layers consist of a BN layer and a 1×1 convolutional layer followed by a 2×2 average pooling layer. Finally, the softmax classifier is applied to return a list of probabilities.

The label with the largest probability is chosen as the final classification. Through a series of nonlinear transformations, the output of single Dense Block is defined as follows:

$$X_l^{'} = H_l\left([x_0, T_l\left(x_0\right), T_l\left(x_1\right) \cdots, T_l\left(x_{l-3}\right), T_l\left(x_{l-2}\right)]\right), \tag{4}$$

where $T_l\left(x_{n-1}\right)$ denotes the input of nth inverted residual block, which is the nonlinear transformed output of $(n-1)$th layer feature connections. Finally, the output of the SE Block is determined as follows:

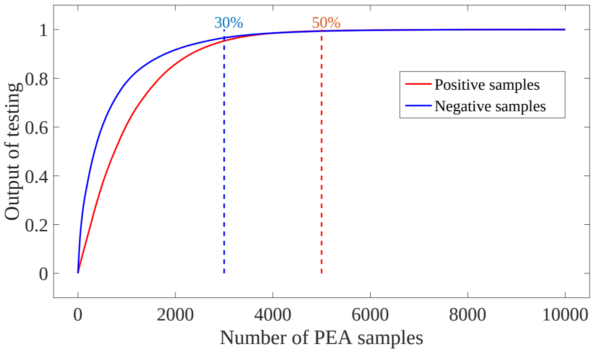


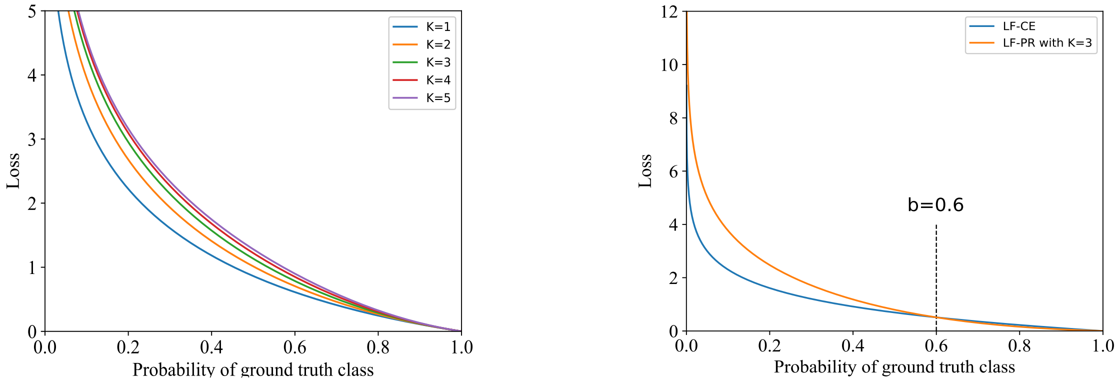
Fig. 11: Threshold distribution of PEAs.

$$\begin{aligned} X_c &= F_{scale}\left(x_c, s\right) \\ &= x_c \times s \\ &= \left[\sum_{S=1}^{C} v_c^s * X_l^{'s}\right] \times \\ &\left[\sigma\left(W_2\delta\left(\frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} x_c\left(i,j\right)W_1\right)\right)\right], \end{aligned} \tag{5}$$

where δ refers to the ReLU function, $W_1 \in \mathbb{R}^{\frac{C}{r}\times C}$ and $W_2 \in \mathbb{R}^{C\times\frac{C}{r}}$. σ and r denote sigmoid activation and reduction ratio, respectively. x_c represents the output from the multipath Dense Blocks via the squeeze operation. s is the output of the expansion operation, which takes the result of x_c as input. v_c denotes the cth convolution core. H and W are the spatial dimensions of feature maps in the SE Block.

B. *Loss Function for PEA Recognition*

In subjective test, the users are unable to mark all PEAs pixel-by-pixel due to the huge amount of data and characteristics of perception artifacts. Instead, we ask the users to present a coarse labelling to video patches, as discussed in Section III. The imperfect dataset leads to a high negative labelling and thus decrease PEA recognition accuracy. In Fig. 11, we randomly select 10,000 samples and observe their thresholds in classification. 50% of positive samples and 70% of negative samples have high prediction accuracies close to



(a) The impact of K on loss function

(b) Comparison between LF-CE and LF-PR.

Fig. 12: Cross Entropy Loss Function for PEA Recognition (LF-PR).

TABLE III: Traning/testing accuracies of ResNeXt, DenseNet and our improved models.

PEAs	ResNeXt		ResNeXt-PR		DenseNet		DenseNet-PR	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing
Blocking	0.9501	0.9320	0.9423	0.9358	0.9503	0.9369	0.9679	**0.9568**
Blurring	0.9491	0.8811	0.9307	0.8845	0.9457	0.9265	0.9551	**0.9387**
Ringing	0.8682	0.8929	0.9040	0.8863	0.9135	0.8879	0.9063	**0.8976**
Color bleeding	0.9412	0.9232	0.9306	0.9230	0.9432	0.9385	0.9589	**0.9403**
Flickering	0.8621	0.8426	0.8552	0.8516	0.9167	0.8956	0.9193	**0.9068**
Floating	0.8398	0.8225	0.8595	0.8396	0.8683	0.8413	0.8852	**0.8606**

1, thus they are considered as easy samples; the others are hard samples. The dominated easy samples would overwhelm training process and inevitably result in degenerate models. A common solution to this problem is hard example mining [43] that samples hard examples during training or more complex sampling/reweighing schemes [44]. However, this approach has a drawback that hard examples are over emphasized in PEA recognition. Thus, a more effective alternative is required to the state-of-the-art approaches for hard examples.

In this work, we propose to reshape the standard Cross Entropy Loss Function (LF-CE) such that it down-weights the loss assigned to well-classified examples. The LF-CE was designed for binary classification:

$$\mathrm{LF{-}CE} = -y \log \delta(x) - (1-y) \log \delta(-x), \tag{6}$$

where $\delta(x)$ represents activation function, $y \in (0, 1)$ denotes the true labels. The LF-CE can be seen in Fig. 12 (b). After classification, the predicted probability p would lay between 0 and 1. Here we set two thresholds, a and b ($0 < a < b < 1$), instead of one threshold 1/2. The probability $p \in (a, b)$ indicates a hard sample; otherwise the test sample is an easy sample. To adjust the importances of different types of examples, we also introduce a K value, that shapes our Loss Function for PEA Recognition (LF-PR) as:

$$\begin{aligned}\mathrm{LF{-}PR} = &- 2y\delta(K(b-p)) \log \delta(x) - \\ & 2(1-y)\delta(K(p-a)) \log \delta(-x),\end{aligned} \tag{7}$$

where the value K varies from 1 to 5 to dynamically scale our entropy loss, as shown in Fig. 12 (a). To observe the impact of LF-PR, a positive example is given in Fig. 12 (b) where its loss curve is compared with that of LF-CE. The value $b = 0.6$ indicates a separatrix that the minimum loss function is different on its both sides. Through replacing LF-CE by LF-PR, the losses of examples are differentiated in order to automatically down-weight the contribution of easy examples and focus on hard examples. Therefore, the function improves the recognition of hard examples without significantly depressing the labelling performance of easy examples. Experiments show an optimal K value of 3 in PEA recognition.

C. *PEA Recognition with CNNs*

For each type of PEAs, we randomly select 50,000 ground-truth samples from the PEA265 database. These samples are further split to 75:25 training/testing sets. The settings of ResNeXt and DenseNet are unified for fair comparison. Stochastic Gradient Descent (SGD) is utilized with a mini-batch size of 256. The momentum is 0.9, and the weight decay is 0.0001. The initial value of learning rate is set to 0.1, and divided by 10 for three times following the schedule in [40]. The weight initialization of [40] is adopted. In ResNeXt and ResNeXt-PR, the depth and cardinality values are set to 50 and 32, respectively. In DenseNet and DenseNet-PR, the width and depth are set to 10 and 46, respectively. We utilize the LF-FR as the loss function in all networks.

TABLE IV: Computational complexity of CNN models. Params and FLOPs represent the numbers of CNN network parameters and floating-point operations, respectively.

CNNs	Params (10^6)	FLOPs (10^7)
ResNeXt	25.00	42.00
ResNeXt-PR	25.00	42.00
DenseNet	0.20	0.98
DenseNet-PR	2.60	1.28

TABLE V: Floating detection accuracy.

Algorithms	Fig. 13 (b)	Fig. 13 (f)	Image3000
Ref [34]	96.10%	54.92%	65.17%
Proposed	97.36%	81.08%	85.69%

By training the recognition model of each type of PEAs, we aim to detect the existence of PEAs in an image/video patch. Note here we do not utilize a multi-target classification because of the non-exclusivity of PEAs (*i.e.* different types of PEAs coexist within one patch). Based on the two above-mentioned architectures, we individually train 6 types of PEA identification models. The training and testing accuracy is defined as follows.

$$\text{Accuracy} = \frac{1}{2}\left(\frac{\text{TP}}{\text{TP+FP}} + \frac{\text{TN}}{\text{FN+TN}}\right), \tag{8}$$

where TP, FP, TN and FN denote the true positive, false positive, true negative, and false negative rates, respectively.

The classification performances of all aforementioned networks are summarized in Table III. Three major conclusions can be drawn. Firstly, the DenseNet has shown its superiority in the PEA classification, as compared with ResNeXt. Secondly, our optimizations on both architectures, ResNeXt and DenseNet, have further improved the accuracy of recognitions. Finally, the DenseNet-PR structure outperforms other architectures in all types of PEAs. The proposed blocking and color bleeding recognition models yield a higher testing accuracy of around 95%. The blurring recognition accuracy is nearly 5.77% higher than that of ResNeXt-PR. In addition, the temporal PEA recognitions based on DenseNet-PR also lead to a higher accuracy over 86%. Compared to ResNeXt-PR, the flickering recognition accuracy increases by nearly 6.09%. Therefore, DenseNet-PR delivers a higher recognition accuracy.

The DenseNet-PR also brings a low computational complexity, as shown in Table IV. Lower complexity in terms of parameter complexity and floating-point operations per second (FLOPs) can be observed for DenseNet and DenseNet-PR, compared with ResNet and ResNeXt. With a tradeoff between complexity and accuracy, we choose the DenseNet-PR for its high accuracy and relatively low complexity. This model is further utilized in the following section to explore its applications in visual quality measurement.

D. Comparison with Other Benchmarks

In order to better illustrate the advantages of the proposed recognition, we compare it with the floating PEA detection method in [34], in which the low-level coding features were extracted to estimate the spatial distribution of floating. Fig. 13 (a) and (e) are two original frames, respectively, and Fig. 13 (b) and (f) are their compressed frames, coded by HEVC with Qp = 42, where the visual floating regions are marked manually. Fig. 13 (c) is the floating map generated by [34], where black regions indicate the floating artifacts. Fig. 13 (d) is the result of the proposed PEA recognition model. In this case, both methods perform reasonably well in floating detection. However, the algorithm in [34] requires content-dependent parameter adjustment and does not generalize consistently. For example, Fig. 13 (g) fails to detect the actual floating region. Compared Fig. 13 (g) with Fig. 13 (h), the proposed floating PEA recognition algorithm performs clearly better. The floating detection accuracy is given in Table V.

V. The Application of PEA Recognition in Video Coding

The PEA recognition has a wide application in lossy video coding, in which the PEA is inevitably produced with high-frequency information loss. Recently, the PEA elimination of in-loop filter and post-processing have been extensively studied, in order to further enhance the visual quality of video coding [24]–[28]. In the above cases, the PEA recognition could provide measurements of their performances. By summarizing all types of PEAs, we also propose two PEA-based metrics in this work: PEA pattern and PEA intensity, which can be further employed in vision-based video processing and coding.

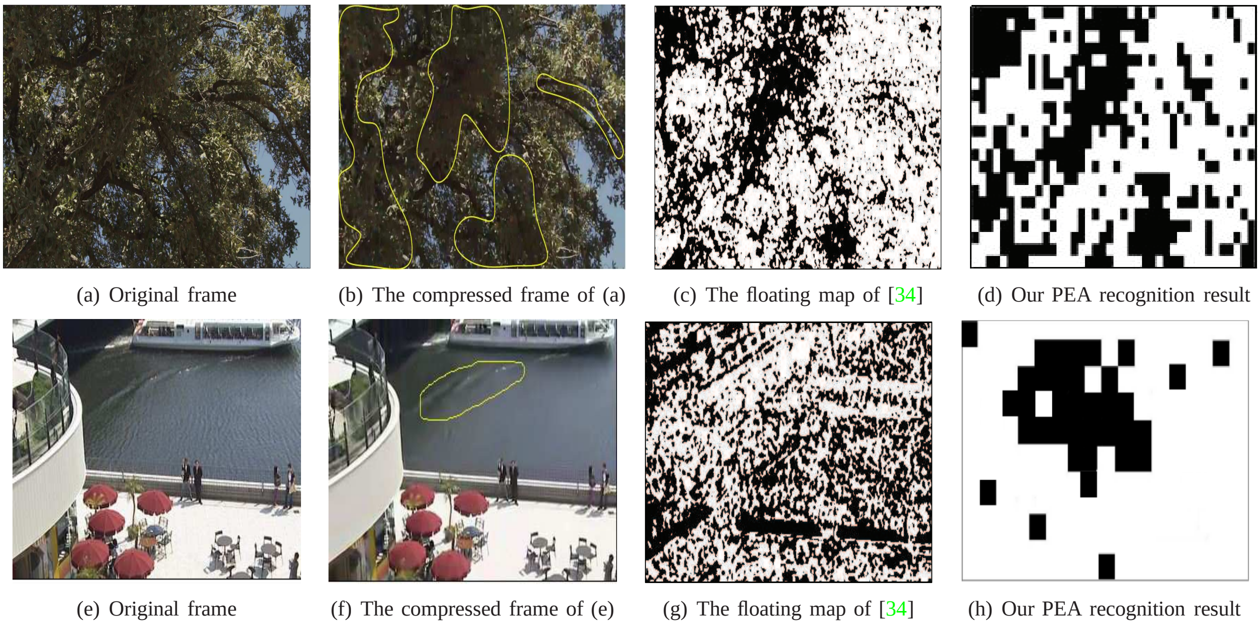
A. The PEA Pattern

We utilize a binary value to represent whether a type of PEA is found within a video patch. For a video slice or frame, all video patches are examined and then visualized as a map to demonstrate the distribution of this type of PEA. Examples of the PEA pattern can be observed in Fig. 14, in which the distributions of six types of PEAs are given in the subfigures (b)-(g). The example frame is from the video PO of LIVE mobile datbase [45], in which the floating artifact has the largest coverage.

For an area with multiple PEA patterns, another method is developed to show a combination of PEA patterns. We utilize a 6-binary value to label whether all types of PEAs are included within a video patch. Each bin in sequence marks the existence of blurring, blocking, ringing, color bleeding, flickering and floating artifacts. An example of combined pattern is given in Fig. 14 (h). As a conclusion, the PEA pattern makes available an intuitive demonstration of PEA distribution. It also serves to the computer vision tasks by providing patch-level artifact labelling.

B. The Intensity of PEAs

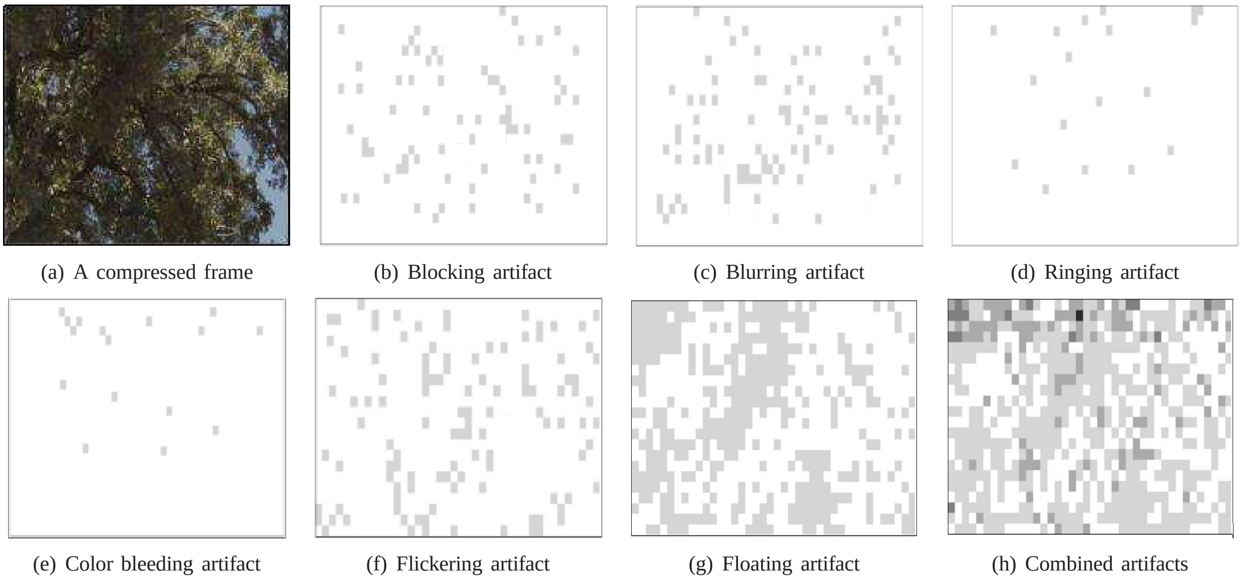
In addition to a map of PEA patterns, we can also measure the overall intensity of PEA or PEAs for a video sequence.



(a) Original frame (b) The compressed frame of (a) (c) The floating map of [34] (d) Our PEA recognition result

(e) Original frame (f) The compressed frame of (e) (g) The floating map of [34] (h) Our PEA recognition result

Fig. 13: An example of floating PEA detection.



(a) A compressed frame (b) Blocking artifact (c) Blurring artifact (d) Ringing artifact

(e) Color bleeding artifact (f) Flickering artifact (g) Floating artifact (h) Combined artifacts

Fig. 14: The individual and overall PEA distributions of a frame.

The intensity of a type of PEA is obtained as the percentage of non-overlapped patches with a positive PEA recognition for this type of PEA. The intensity of a set of PEAs (*e.g.* spatial or temporal PEAs) is obtained as the percentage of non-overlapped patches with positive PEA recognition for either type of PEA in this set. Correspondingly, the overall intensity of PEAs, or PEA intensity (I_{PEA}), is calculated as follows:

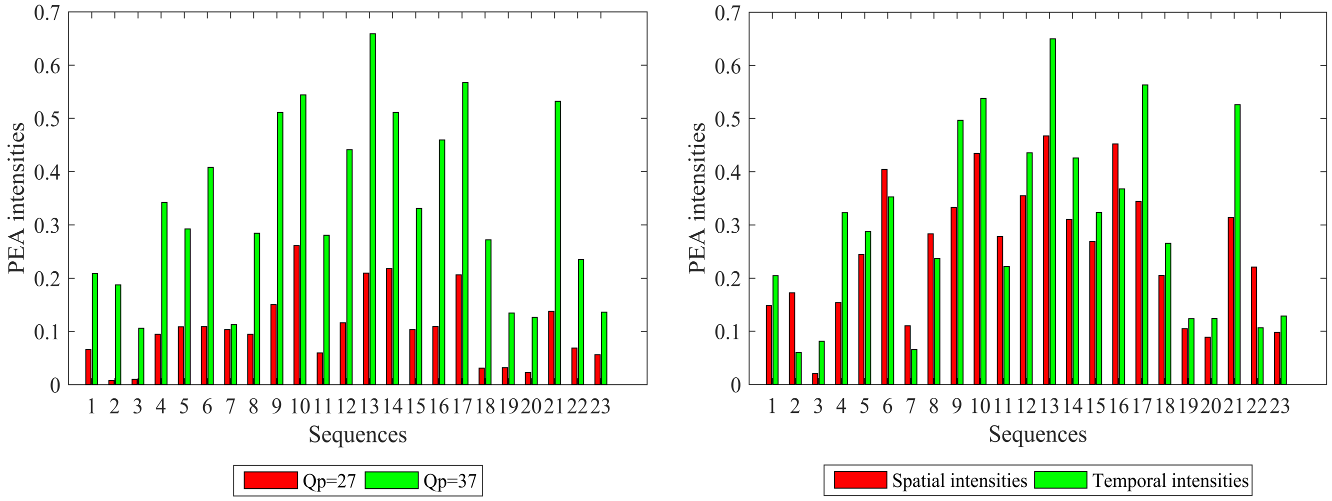
$$\mathrm{PEA_i} = \mathrm{PEA_{i1}}|\mathrm{PEA_{i2}}|\mathrm{PEA_{i3}}|\mathrm{PEA_{i4}}|\mathrm{PEA_{i5}}|\mathrm{PEA_{i6}}, \quad (9)$$

$$I_{\mathrm{PEA}} = \frac{\sum_{i=1}^{N_{\mathrm{total}}} \mathrm{PEA_i}}{N_{\mathrm{total}}}, \quad (10)$$

where $\mathrm{PEA_{i1}}$ to $\mathrm{PEA_{i6}}$ represent the existence of blurring, blocking, ringing, color bleeding, flickering and floating artifacts within an image/video patch, respectively. We set it to 1 if its corresponding PEA exists; otherwise 0. $\mathrm{PEA_i}$ refers to the PEA existence in the ith image/video patch. N_{total} represents the number of non-overlapping patches of a video sequence.

Based on the above metrics, we investigate the CTC sequences and present their PEA intensity I_{PEA}, as well as the intensities for spatial and temporal PEAs in Fig. 15. Several conclusions can be drawn here.

Firstly, the I_{PEA} is, in general, positively correlated to the Qp value. For almost all types of PEAs and videos, the I_{PEA} grows with a higher Qp. This fact highlights the importance of quantization and information loss in the generation mechanism of PEAs. As discussed before, the potential origin of spatial artifacts are interpreted as the loss of high frequency signals, chrominance signals and inconsistency of information loss



(a) PEA intensities under different Qps

(b) Spatial and temporal intensities under Qp=37

Fig. 15: Average PEA intensities of the CTC sequences.

between boundaries, while the temporal artifacts are possibly produced by inconsistent information loss between frames. Therefore, the fact that Qp influences I_{PEA} complies the above interpretations.

Secondly, the I_{PEA} is content-dependent, as it varies subject to video contents. For example, the sequence *BasketballDrive* (1920×1080, No.9), *RaceHorses* (832×480, No.10), *BasketballDrill* (832×480, No.13) and *BasketballPass* (416×240, No.17) have severe PEA intensities while the sequence *NebutaFestival* (2560×1600, No.3) is with low intensities. Although some sequences (*e.g.*, *ParkScene*, 1920×1080, No. 6 and *BlowingBubbles*, 416×240, No. 16) have similar spatial/temporal PEA intensities, their individual PEA intensities are distinct from each other. This fact implies that the video characteristics, including texture and motion, might have an impact on the I_{PEA} when being compressed. It may also provide useful instructions to content-aware video coding optimization.

Thirdly, the frequencies of PEAs can be different subject to its type. For example, the frequencies of blocking, blurring and flickering PEAs are higher than other three PEAs in this database. Meanwhile, the intensities of temporal PEAs are significant compared with spatial PEAs. Furthermore, the impact on visual quality changes for different types of PEAs. All types of PEAs do not have the same impact on HVS and the visual quality of users may be dominated by parts of PEAs, as concluded in [34]. We put this in future work to explore how PEA detection should be combined to best evaluate their impact on visual quality.

C. PEA-based Video Quality Metric

In this section, we utilize the aforementioned PEA intensity to propose a quality metric for compressed videos. Inspired by the conclusions of Section V.B, the video quality is simply measured by a negative value of PEA intensity:

$$Q = -I_{\text{PEA}}, \tag{11}$$

where Q represents the PEA-based Video Quality Metric (P-VQM).

To verify its performance, it is evaluated on the LIVE Video Quality Database [46]. The LIVE database is a popular video quality database with standard subjective scores. It contains 10 reference videos, *Blue_Sky* (217 frames at 25fps), *Pedestrian_Area* (250 frames at 25fps), *River_Bed* (250 frames at 25fps), *Rush_Hour* (250 frames at 25fps), *Tractor* (250 frames at 25fps), *Station* (250 frames at 25fps), *Sunflower* (250 frames at 25fps), and *Shields* (500 frames at 50fps), *Mobile_Calender* (500 frames at 50fps), *Park_Run* (500 frames at 50fps). Four types of distortions are presented with video data and the Mean Opinion Score (MOS). Among them, the distortion of compression is generated by H.264 encoder at bitrates from 200 kbps to 5 Mbps, resulting 40 outputs. This subset is utilized here to evaluate the performance of P-VQM.

The method of P-VQM is also compared with typical video quality metrics including PSNR, SSIM [29], MS-SSIM [47], VIIDEO [48], SpEED-QA [49] and AMB-VQM [33] to show its performance. In addition, the Pearson Linear Correlation Coefficient (PLCC) and Spearman Rank-order Correlation Coefficient (SROCC) are utilized as the performance indicators. The results summarized in Table VI have validated the superior performance of our method when being utilized as a measurement of compressed video quality. In other words, the results also demonstrate a fact that the existence of PEAs is a dominant factor to degrade the state-of-the-art compressed video quality, especially for low resolution videos. Therefore, to eliminate the intensity of PEAs is an effective approach to optimize video quality during video compression.

The high performance of P-VQM supports its application as a testing tool [50] in the state-of-the-art video coding. On the other hand, this metric, as well as most of other video quality measures, are not recommended be utilized as an optimization tool of video coding due to their high computational complexity. The numerously repeated calculation during video coding optimization requires a quality measure of extremely low

TABLE VI: Average PLCC and SROCC of P-VQM compared with existing metrics on LIVE database.

Methods	PLCC	SROCC
PSNR	0.5735	0.4146
SSIM [29]	0.6072	0.5677
MS-SSIM [47]	0.6924	0.7343
VIIDEO [48]	0.6829	0.6593
SpEED-QA [49]	0.7933	0.7895
AMB-VQM [33]	0.4916	0.5189
P-VQM	**0.8653**	**0.8278**

computational overhead. In such a case, we still recommend using PSNR and SSIM for video coding optimization.

VI. Conclusion

We construct a PEA265 database, a first-of-its-kind large-scale subject-labeled database of PEAs produced by H.265/HEVC video compression. This database contains 6 spatial and temporal PEA types, including blurring, blocking, ringing, color bleeding, flickering and floating, each with at least 60,000 samples with positive or negative labels. Based on this database, we optimize popular CNNs to develop effective PEA recognition, in which the improved DenseNet provides high accuracy with a relatively low complexity. We also define qualitative and quantitative measures based on the recognition of PEAs. The proposed P-VQM model shows comparable performance with typical video quality metrics. This work will benefit the future development of video quality assessment algorithms. It can also be used to optimize hybrid video encoders for improved perceptual quality and perceptually-motivated video encoding schemes.

References

[1] Cisco visual networking index: Forecast and trends, 2017-2022. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html

[2] J. P. Allebach, "Human vision and image rendering: Is the story over, or is it just beginning," in *Proc. SPIE 3299, Human Vision and Electronic Imaging (HVEI) III*, Jul. 1998, pp. 26–37.

[3] *H.264: Advanced video coding for generic audiovisual services*, Recommendations, ITU-T, Mar. 2005.

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

[5] J. Chen and E. Alshina, "Algorithm description for versatile video coding and test model 2 (vtm 2)," *document JVET-K1002-v2*, Jul. 2018.

[6] J. Bankoski, P. Wilkins, and Y. Xu, "Technical overview of vp8, an open source video codec for the web," in *IEEE Int. Conf. Multimedia and Expo (ICME)*, Jul. 2011, pp. 1–6.

[7] D. M. et al., "The latest open-source video codec vp9 - An overview and preliminary results," in *Proc. 2013 Picture Coding Symposium (PCS)*, 2013, pp. 390–393.

[8] L. Fan, S. Ma, and F. Wu, "Overview of AVS video standard," in *Proc. IEEE Int. Conf. Multimedia and Expo (ICME)*, Jun. 2004, pp. 423–426.

[9] Z. He, L. Yu, X. Zheng, S. Ma, and Y. He, "Framework of AVS2-video coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2013, pp. 1515–1519.

[10] M. Yuen and H. Wu, "A survey of hybrid MC/DPCM/DCT video coding distortions," *Signal Process.*, vol. 70, no. 3, pp. 247–278, Nov. 1998.

[11] Y. Gong, S. Wan, F. Yang, H. R. Wu, and B. Li, "A frame level metric for just noticeable temporal pumping artifact in videos encoded with the hierarchical prediction structure," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 3034–3038.

[12] A. F. Silva, M. C. Q. Farias, and J. A. Redi, "Perceptual annoyance models for videos with combinations of spatial and temporal artifacts," *IEEE Trans. Multimedia.*, vol. 18, no. 12, pp. 2446–2456, Dec. 2016.

[13] S. Sakaida, Y. Sugito, H. Sakate, and A. Minezawa, "Video coding technology for 8k/4k era," *Journal of the Institute of Electronics Information and Communication Engineers*, vol. 98, pp. 218–224, Mar. 2015.

[14] Y. Gong, S. Wan, K. Yang, and H. Wu, "A visual-masking-based estimation algorithm for temporal pumping artifact region prediction," *Circuits Syst. Signal Process.*, vol. 3, no. 36, pp. 1264–1287, Jun. 2016.

[15] A. V. Umnov and A. S. Krylov, "Sparse approach to image ringing detection and suppression," *Pattern Recognition and Image Analysis*, no. 27, pp. 754–762, Dec. 2017.

[16] T. R. Goodall and A. C. Bovik, "Detecting and mapping video impairments," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2680–2691, Jun. 2019.

[17] H. Chen, X. He, C. An, and T. Q. Nguyen, "Deep wide-activated residual network based joint blocking and color bleeding artifacts reduction for 4:2:0 JPEG-compressed images," *IEEE Signal Process. Lett.*, vol. 26, no. 1, pp. 79–83, Jan. 2019.

[18] A. Mosleh, Y. E. Sola, F. Zargari, E. Onzon, and J. M. P. Langlois, "Explicit ringing removal in image deblurring," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 580–593, Feb. 2018.

[19] M. K. Kaushik, G. C. Chandrakala, and R. Abhinay, "Ringing and blur artifact removal in image processing applications," in *Proc. 2nd Int. Conf. Intelligent Comput. Control Syst. (ICICCS)*, Jun. 2018, pp. 261–264.

[20] L. Yann, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[21] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Convolutional neural network committees for handwritten character classification," in *proc. Int. Conf. Doc. Anal. Recogn. (ICDAR)*, Nov. 2011, pp. 1135–1139.

[22] L. Yu, L. Shen, H. Yang, L. Wang, and P. An, "Quality enhancement network via multi-reconstruction recursive residual learning for video coding," *IEEE Signal Process. Lett.*, vol. 26, no. 4, pp. 557–561, Apr. 2019.

[23] X. He, Q. Hu, X. Zhang, W. Lin, and X. Han, "Enhancing HEVC compressed videos with a partition-masked convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 216–220.

[24] Y. Zhang, T. Shen, X. Ji, Y. Zhang, R. Xiong, and Q. Dai, "Residual highway convolutional neural networks for in-loop filtering in HEVC," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3827–3841, Aug. 2018.

[25] J. Kang, S. Kim, and K. M. Lee, "Multi-modal/multi-scale convolutional neural network based in-loop filter design for next generation video codec," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2017, pp. 26–30.

[26] J. Wang, S. Liu, F. Jiang, X. Sun, and Y. Liu, "A video post-filter deblocking method based on temporal boosting residual networks," in *IEEE Int. Conf. Multimedia and Expo (ICME)*, Jul. 2019, pp. 1174–1179.

[27] Y. Dai, D. Liu, and F. Wu, "A convolutional neural network approach for post-processing in HEVC Intra coding," in *Proc. the 23rd Int. Conf. MultiMedia Model (MMM)*, 2017, pp. 28–39.

[28] S. Yu, B. Chen, Y. Xu, W. Chen, Z. Chen, and T. Zhao, "HEVC artifact reduction with generative adversarial network," in *Proc. the 11th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2019, pp. 1–6.

[29] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[30] J. Xia, Y. Shi, K. Teunissen, and I. Heynderickx, "Perceivable artifacts in compressed video and their relation to video quality," *Signal Process. Image Commun.*, vol. 24, no. 7, pp. 548–556, Aug. 2009.

[31] M. B. Amor, F. Kammoun, and N. Masmoudi, "A quality evaluation model for calculating block and blur effects generated by H. 264 and MPEG2 codecs," *Comput. Stand. Interfaces.*, vol. 61, pp. 36–44, Jan. 2019.

[32] M. K. Rohil, N. Gupta, and P. Yadav, "An improved model for no-reference image quality assessment and a no-reference video quality assessment model based on frame analysis," *Signal, Image and Video Process.*, pp. 205–213, Aug. 2019.

[33] V. Mario, B. Viliams, G. Ratko, and V. Denis, "No-reference artifacts measurements based video quality metric," *Signal Process. Image Commun.*, vol. 78, pp. 345–358, Oct. 2019.

[34] K. Zeng, T. Zhao, A. Rehman, and Z. Wang, "Characterizing perceptual artifacts in compressed video streams," in *Proc. SPIE - The International Society for Optical Engineering*, vol. 9014, Jan. 2014, pp. 173–182.

[35] G. Huang, Z. Liu, L. Maaten, and K. Weinberger, “Densely connected convolutional networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, Jul. 2017, pp. 4700–4708.

[36] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, 2017, pp. 5987–5995.

[37] F. Bossen, “HM 10 common test conditions and software reference configurations,” in *Proc. Joint Collaborative Team on Video Coding Meeting (JCT-VC)*, Jan. 2013, pp. 1–3.

[38] *Methodology for the subjective assessment of video quality in multimedia applications*, ITU-T P.910, International Telecommunication Union, 1999.

[39] *Methodology for the subjective assessment of the quality of television pictures*, ITU-R. BT. 500-13, International Telecommunication Union, 2012.

[40] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, Nov. 2016, pp. 770–778.

[41] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, Jun. 2018, pp. 7132–7141.

[42] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proc. Int. Conf. Mach. Learn. (ICML)*, Mar. 2015, pp. 1–10.

[43] A. Shrivastava, A. Gupta, and R. Girshick, “Training region-based object detectors with online hard example mining,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, Jun. 2016, pp. 761–769.

[44] S. R. Bulo, G. Neuhold, and P. Kontschieder, “Loss maxpooling for semantic image segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, Apr. 2017, pp. 7082–7091.

[45] A. K. Moorthy, L. K. Choi, A. C. Bovik, and G. de Veciana, “Video quality assessment on mobile devices: Subjective, behavioral and objective studies,” *IEEE J. Sel. Top. Signal Process.*, vol. 6, no. 6, pp. 652–671, Oct. 2012.

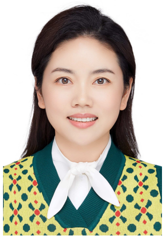
[46] Live video quality assessment database. [Online]. Available: http://live.ece.utexas.edu/research/quality/live_video.html

[47] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multiscale structural similarity for image quality assessment,” in *Proc. the 37th Asilomar Conference on Signals, Systems and Computers*, Nov. 2003, pp. 1398–1402.

[48] A. Mittal, M. A. Saad, and A. C. Bovik, “A completely blind video integrity oracle,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 1, pp. 289–300, Jan. 2016.

[49] C. G. Bampis, P. Gupta, R. Soundararajan, and A. C. Bovik, “SpEED-QA: Spatial efficient entropic differencing for image and video quality,” *IEEE Signal Process. Lett.*, vol. 24, no. 9, pp. 1333–1337, Sep. 2017.

[50] T. Zhao, K. Zeng, A. Rehman, and Z. Wang, “On the use of SSIM in HEVC,” in *Proc. Asilomar Conference on Signals, Systems and Computers*, Nov. 2013, pp. 1107–1111.



Liqun Lin received the M.S. and PhD degrees in communication and information system from Fuzhou University, Fuzhou, China, respectively in 2007 and 2019. She is currently a lecturer in Fuzhou University, Fuzhou, China.

Her research interests include image/video signal processing, visual quality assessment and video coding.

Shiqi Yu received the B.S. degree in electronic information engineering from Fuzhou University in 2017. He is currently pursuing the M.S. degree in signal and information processing with Fuzhou University, Fuzhou, China.

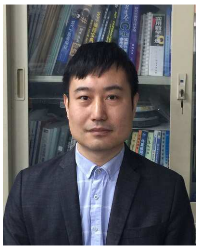
His current research interests include image processing, computer vision and video coding.

Liping Zhou received the B.S. degree in electronic information engineering from the Jiangxi Normal University, Nanchang, China, in 2018. She is currently pursuing the M.S. degree in signal and information processing with Fuzhou University, Fuzhou, China.

Her research interest is mainly video quality assessment.

Weiling Chen (M’19) received the B.S. and PhD degrees in communication engineering from Xiamen University, Xiamen, China, respectively in 2013 and 2018. She is currently a lecturer in Fuzhou University, Fuzhou, China. From Sep. 2016 to Dec. 2016, she was visiting at the School of Computer Science and Engineering, Nanyang Technological University, Singapore.

Her current research interests include image quality assessment, image compression, and underwater acoustic communication.



Tiesong Zhao (S’08-M’13-SM’19) received the B.S. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 2006, and the Ph.D. degree in computer science from the City University of Hong Kong, Hong Kong, in 2011. He served as a Research Associate with the Department of Computer Science, City University of Hong Kong (2011-2012), a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Waterloo (2012-2013) and a Research Scientist with the Ubiquitous Multimedia Laboratory, The State University of New York at Buffalo (2014-2015). He is currently a Professor and a Minjiang Distinguished Professor in the College of Physics and Information Engineering, Fuzhou University, China.

His research interests include multimedia signal processing, coding, quality assessment and transmission. Dr. Zhao has around 50 publications in these fields. Due to his contributions in video coding and transmission, he received the Fujian Science and Technology Award for Young Scholars in 2017. He has also been serving as an Associate Editor of IET Electronics Letters since 2019.

Zhou Wang (S’99-M’02-SM’12-F’14) received the Ph.D. degree from the University of Texas at Austin in 2001. He is currently a Canada Research Chair and Professor in the Department of Electrical and Computer Engineering, University of Waterloo, Canada.

His research interests include image and video processing and coding; visual quality assessment and optimization; computational vision and pattern analysis; multimedia communications; and biomedical signal processing. He has more than 200 publications in these fields with over 40,000 citations (Google Scholar). Dr. Wang serves as a Senior Area Editor of IEEE Transactions on Image Processing (2015-present). Previously, he served as a member of IEEE Multimedia Signal Processing Technical Committee (2013-2015), an Associate Editor of IEEE Transactions on Circuits and Systems for Video Technology (2016-2018), IEEE Transactions on Image Processing (2009-2014), Pattern Recognition (2006-present) and IEEE Signal Processing Letters (2006-2010), and a Guest Editor of IEEE Journal of Selected Topics in Signal Processing (2013-2014 and 2007-2009). He is a Fellow of Royal Society of Canada and Canadian Academy of Engineering, and a recipient of 2017 Faculty of Engineering Research Excellence Award at University of Waterloo, 2016 IEEE Signal Processing Society Sustained Impact Paper Award, 2015 Primetime Engineering Emmy Award, 2014 NSERC E.W.R. Steacie Memorial Fellowship Award, 2013 IEEE Signal Processing Magazine Best Paper Award, and 2009 IEEE Signal Processing Society Best Paper Award.