Deep Learning-Based Picture-Wise Just Noticeable Distortion Prediction Model for Image Compression

Huanhua Liu, Yun Zhang, *Senior Member, IEEE*, Huan Zhang, *Student Member, IEEE*, Chunling Fan, Sam Kwong, *Fellow, IEEE*, C.-C. Jay Kuo, *Fellow, IEEE*, and Xiaoping Fan

***Abstract*— Picture Wise Just Noticeable Difference (PW-JND), which accounts for the minimum difference of a picture that human visual system can perceive, can be widely used in perception-oriented image and video processing. However, the conventional Just Noticeable Difference (JND) models calculate the JND threshold for each pixel or sub-band separately, which may not reflect the total masking effect of a picture accurately. In this paper, we propose a deep learning based PW-JND prediction model for image compression. Firstly, we formulate the task of predicting PW-JND as a multi-class classification problem, and propose a framework to transform the multi-class classification problem to a binary classification problem solved by just one binary classifier. Secondly, we construct a deep learning based binary classifier named perceptually lossy/lossless predictor which can predict whether an image is perceptually lossy to another or not. Finally, we propose a sliding window based search strategy to predict PW-JND based on the prediction results of the perceptually lossy/lossless predictor. Experimental results show that the mean accuracy of the perceptually lossy/lossless predictor reaches 92%, and the absolute prediction error of the proposed PW-JND model is 0.79 dB on average, which show the superiority of the proposed PW-JND model to the conventional JND models.**

***Index Terms*— Just noticeable distortion, convolutional neural network, visual perception, image quality assessment.**

I. Introduction

THE Ultra-High-Definition (UHD), Three Dimensional (3D) [1], and Virtual Reality (VR) images and videos with the ability to provide a more immersive and realistic experience than conventional multimedia, are becoming more and more popular with consumers in streaming services [2]. However, the bandwidth and storage required to support UHD, 3D, and VR streaming services are several or even more times the size of that required for the traditional images and videos, which has been a bottleneck of the streaming services industry. The main-stream of the current image/video coding techniques [3] are signal-processing-based, which mainly consider the statistical properties of visual content. They are becoming difficult to achieve further improvement in reducing the size of images and videos without perceptual quality degradation. As we know, the ultimate receiver of most visual content is the Human Visual System (HVS), therefore it is important to develop image/video processing technologies incorporating the characteristic of HVS [4] for streaming services industry.

It is well known that humans cannot perceive the small changes in the images/videos due to the psychological and physiological mechanism of HVS. Therefore the processed images/videos have visual redundancy which can be removed without any perceptual quality degradation. Just Noticeable Difference (JND) refers to the minimum distortion HVS can perceive, which has been widely used in image/video processing, *e.g.*, perceptual image/video coding [5]–[7], image enhancing [8], and objective quality estimation [9]. The existing JND models can be divided into two categories: 1) pixel-domain models [10]–[15] calculate JND threshold for each pixel directly in the pixel domain; 2) sub-band domain models transfer pixel domain images to the sub-band domain, *e.g.*, Discrete Cosine Transformation (DCT), then calculate the JND threshold for each sub-band [16]–[20].

Manuscript received July 23, 2018; revised February 4, 2019 and May 27, 2019; accepted July 29, 2019. Date of publication August 13, 2019; date of current version September 25, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61871372, in part by the Guangdong Natural Science Foundation for Distinguished Young Scholar under Grant 2016A030306022, in part by the Key Project for Guangdong Provincial Science and Technology Development under Grant 2017B010110014, in part by the Shenzhen International Collaborative Research Project under Grant GJHZ20170314155404913, in part by the Shenzhen Science and Technology Program under Grant JCYJ20170811160212033, in part by the Guangdong International Science and Technology Cooperative Research Project under Grant 2018A050506063, in part by the Membership of Youth Innovation Promotion Association, Chinese Academy of Sciences under Grant 2018392, and in part by the Shenzhen Science and Technology Plan Project under Grant JCYJ20180507183823045. The associate editor coordinating the review of this article and approving it for publication was Prof. Damon M. Chandler. *(Corresponding author: Yun Zhang.)*

H. Liu is with the School of Computer Science and Engineering, Central South University, Changsha 410075, China, and also with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: hhl_csu@csu.edu.cn).

Y. Zhang is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: yun.zhang@siat.ac.cn).

H. Zhang and C. Fan are with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with the Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: huan.zhang@siat.ac.cn; fancl@siat.ac.cn).

S. Kwong is with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: cssamk@cityu.edu.hk).

C.-C. J. Kuo is with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: cckuo@sipi.usc.edu).

X. Fan is with the School of Information Technology and Management, Hunan University of Finance and Economics, Changsha 410205, China, and also with the School of Automation, Central South University, Changsha 410075, China (e-mail: xpfan@csu.edu.cn).

Digital Object Identifier 10.1109/TIP.2019.2933743

Pixel domain JND models mainly focus on the background luminance adaptation and spatial contrast masking. In [10], a novel spatial masking function was introduced, which was combined with luminance adaptation to deduce the overall JND thresholds. Wu *et al.* [11] suggested that there exists disorderly concealment effect resulting in high JND thresholds of the disorderly region, and proposed a free energy based JND model aiming to improve the accuracy of JND threshold estimation of texture regions. In [12], a novel pattern masking function deduced from luminance contrast and structural uncertainty was incorporated into the proposed JND model. Wang *et al.* [13] proposed an edge profile reconstruction based JND model for screen content images. Each edge profile was decomposed into its luminance, contrast, and structure part, each of which was evaluated respectively. Wu *et al.* [14] proposed an improved pattern masking function based model, where the pattern complexity was calculated as the diversity of orientation in local region. In [15], a new JND model was proposed which considers visual saliency. However, the pixel domain JND models can hardly be incorporated into sub-band image/video compression systems.

The sub-band domain JND models mainly focus on Contrast Sensitive Function (CSF), luminance adaptation, contrast masking, and foveated masking. Wei and Ngan [16] proposed a CSF-based spatio-temporal JND model, in which gamma correction was introduced to compensate luminance adaptation effect. In [17], a luminance adaptation JND model was proposed which took frequency characteristics into the luminance adaption. Bae and Kim [18] proposed a novel JND model being applicable to any size of transform kernel, which introduced a new texture complexity metric to measure contrast masking effect. In [19], foveated masking was incorporated into the proposed temporal-foveated masking model which also considered the difference between moving and still objects. Recently, learning techniques have also been applied to estimating the JND thresholds. In [20], Ki *et al.* proposed a regression based method to estimate the JND thresholds under the distortion with energy reduction. However, the sub-band domain models require a DCT transform, and can hardly estimate thresholds of the complicated texture regions accurately since each block is isolated from its surrounding.

The pixel/sub-band domain JND models compute the JND threshold for each pixel/sub-band separately. By a simple summation of the estimated JND thresholds, they may not reflect the total masking effect of a picture accurately. The contribution of different regions to the image quality is different, and some critical regions together with the worst quality ones determine the image quality [21]. Moreover, the traditional JND models mainly focused on the pristine images/videos but not on the distorted ones, which limits the application areas, since the images/videos fed into the real-world applications are usually degraded. Recent studies [22]–[25] demonstrated that humans cannot perceive continuous-scale visual quality that changes over a range of coding bit rate, and this phenomenon was quantified based on the notion of JND. Hu *et al.* [22] proposed a subjective methodology to find the JND images under Joint Photographic Experts Group (JPEG) compression, which are the transition images between two adjacent perceptual quality levels. The distortion of JND image reflects the total masking effect of a picture accurately, which can be defined as Picture Wise Just Noticeable Difference (PW-JND) referring to the minimum difference of a picture that can be perceived by the HVS. Jin *et al.* [23] constructed the first JND based image quality data set MCL-JCI. Wang *et al.* [24] proposed a subjective methodology to find the video-wise JND videos and constructed the first video-wise JND based quality data set MCL-JCV. Fan *et al.* [25] studied the PW-JND of symmetrically and asymmetrically compressed stereoscopic images for JPEG2000 and H.265 intra coding. Two PW-JND based stereo image quality datasets have been provided: one for symmetric compression and one for asymmetric compression. Huang *et al.* [26] proposed a machine learning approach to predict the mean of group-based JND distribution by using the extracted features of videos. As every one knows, subjective prediction methods are too time consuming to apply into the real-world systems. Therefore, it deserves to devise objective PW-JND prediction method, which is more challenging than to estimate the pixel/sub-band JND threshold. There are more factors affecting the PW-JND thresholds, *e.g.*, distortion type, contrast masking, and luminance adaptation. In this paper, we propose a PW-JND model to predict PW-JND for pristine and distorted images. The main contributions of our work can be summarized as:

1) We formulate the task of predicting PW-JND as a multi-class classification problem, and propose a framework to transform the multi-class classification problem to a binary classification problem.
2) We construct a deep learning based binary classifier named perceptually lossy/lossless predictor. It can predict whether a distorted image is perceptually lossy to its reference or not. The experimental results show that its mean accuracy reaches 92%.
3) We propose a sliding window based search strategy to predict the PW-JND based on the prediction results of the perceptually lossy/lossless predictor.

The paper is organized as follows. In Section II, we present the motivation of predicting PW-JND which is formulated as a multi-class classification problem. Section III presents the framework of the proposed PW-JND model which transforms the multi-class classification problem to a binary classification problem. In Section IV, we propose a deep learning based perceptually lossy/lossless predictor which can predict whether a distorted image is perceptually lossy to its reference or not, and evaluate the performance of the predictor. In Section V, we propose a sliding window based PW-JND search strategy. In Section VI, we report the experimental results. Section VII concludes this paper.

II. Motivation and Problem Formulation

The traditional Rate-Distortion (R-D) function shown in Fig. 1(a) is continuous and convenient for the computation of coding systems. However, the visual quality experience of humans is a discrete rather than continuous process. Recent studies [22], [24] demonstrated that humans can only distinguish several limited quality levels of the image/video changing in a range of bit rate. A perceptual distortion function

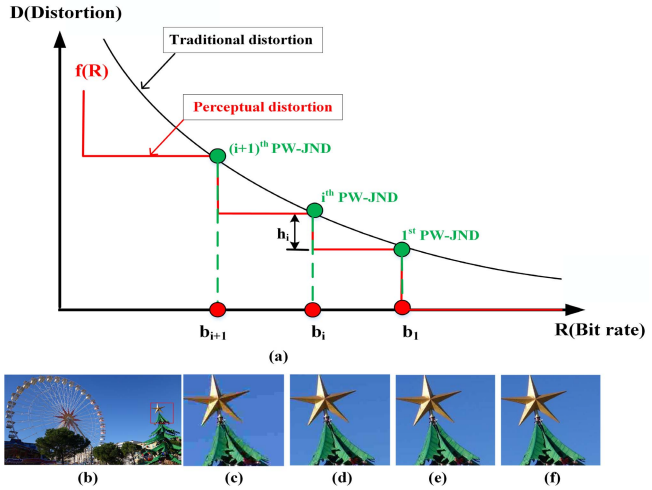


Fig. 1. Illustration of perceptual distortion of JPEG-compressed images, (c) to (f) are enlarged patches. (a) The difference between JND based stair R-D function and the traditional R-D function [24]. (b) Pristine image, size = 6220 KB, MSE = 0. (c) The third PW-JND image, size = 80 KB, MSE = 199.5. (d) The second PW-JND image, size = 159KB, MSE = 99.6. (e) The first PW-JND image, size = 235 KB, MSE = 70.2. (f) JPEG-compressed image with QF 100, size = 1728 KB, MSE = 3.59.

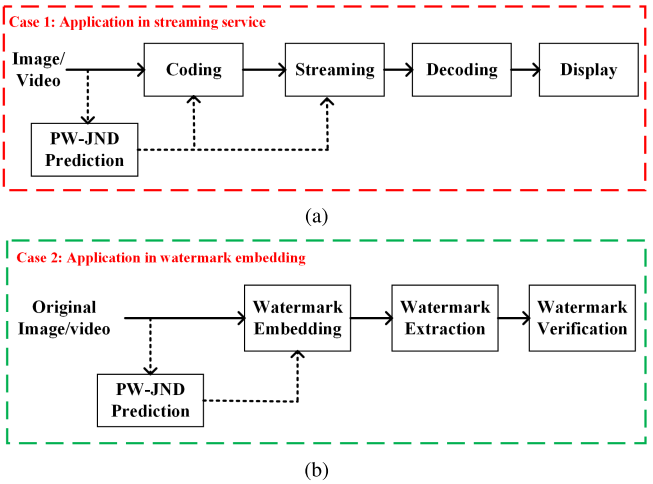


Fig. 2. Application scenarios of PW-JND model. (a) Application in streaming service. (b) Application in watermark embedding.

f(R) shown in Fig. 1(a) was proposed, which is a stair step function about bit rate. In *f(R)*, the jump points denoted by the circles between two adjacent quality levels are JND points [22], [23], *e.g.*, the first JND point jumps from the best to the secondary quality level. It can be seen from *f(R)* that the bit rate of the compressed images with the same perceptual quality vary greatly, and JND points have the lowest bit rate in a given quality level. For example, Fig. 1(b) is the pristine image with size 6220 KB, Fig. 1(f)-(c) are enlarged patches of JPEG-compressed images coded from Fig. 1(b) with different QF. Fig. 1(f) is cropped from the compressed image coded with QF 100, Fig. 1(e)-(c) are cropped from the first, second, and third JND images of Fig. 1(b) [23]. The size of the associated images of patch Fig. 1(f)-(c) are 1728 KB, 235 KB, 159 KB, and 80 KB, and the Mean Squared Error (MSE) values are 3.95, 70.2, 99.6, and 199.5 respectively. The perceptual quality of Fig. 1(e) is nearly equal to that of Fig. 1(f), but the size of the associated image of Fig. 1(e) is much smaller than that of Fig. 1(f). We can also see the similar phenomenon from Fig. 1(c) and (d). We define the bit rate of the first, second, and third JND images as the first, second, third PW-JND respectively. Therefore, PW-JND prediction can be used to guide coding, which can help save bit rate without perceptual quality degradation. As far as we know, it is the first work to predict PW-JND, which is different from conventional Mean Option Score (MOS) or Difference Mean Option Score (DMOS) predictors, *e.g.*, SSIM (MS-SSIM) [27], FSIM [28], GMSD [29], and VSI [30]. First of all, there is an assumption [22]–[26] that the perception model on quality of HVS is discrete in PW-JND prediction, and PW-JND is the boundary between two adjacent perceptual quality levels of the reference image. However, in conventional MOS/DMOS prediction it is continuous. Secondly, in PW-JND prediction, sample label is a relative value which denotes whether the difference between a distorted image and its reference can be perceived by humans or not. In MOS/DMOS evaluating, sample label is an absolute score describing the overall image quality. Thirdly, PW-JND prediction model will mainly be used to predict distorted images of which the difference cannot be perceived by humans. The conventional MOS/DMOS evaluators [31] were mainly used to evaluate perceptually lossy images.

Two applications of PW-JND prediction model are listed in Fig. 2. For streaming media systems, high visual quality requires large bit rate, and lower bit rate can only provide low quality visual content. However, higher bit rate than what it needs means a waste of storage and bandwidth, but lower bit rate will damage the consumers' visual experience. Fig. 2(a) shows a typical framework of streaming media system including coding, streaming, decoding, and display process. In the streaming systems, the PW-JND prediction model can be used for coding or selecting the images/videos with the smallest size but best quality, which can help save the bandwidth without damaging consumers' experience. Fig. 2(b) shows a digital watermarking system which includes watermark embedding, watermark extraction, and watermark verification blocks. Watermark embedding block is responsible for embedding the digital watermark (*i.e*, ownership) into digital media for copyright protection, source tracking, and so on. The embedded digital watermarks are often required to be only perceptible under certain conditions for human beings. Therefore, PW-JND prediction model can be used to guide the embedding process. Although PW-JND prediction model has a wide range of applications, there are many challenges in designing an accurate PW-JND model. First of all, the PW-JND has a wide range of values affected by the visual content which varies greatly. Secondly, the distortion can be introduced at different stages, *e.g.*, pre-processing, compression, and transmission, each of which affects PW-JND thresholds in different manners. Thirdly, we can hardly build an accurate mathematical PW-JND model because it is not clear about the mechanism of HVS in processing visual signals.

In order to formulate the problem of predicting PW-JND, we define the perceptual distortion function $f(R)$ shown

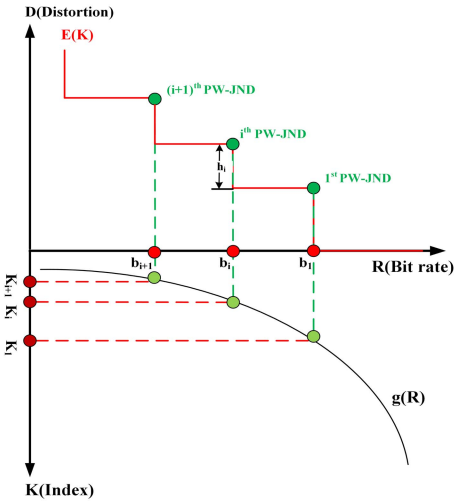


Fig. 3. The relationship between D and R/K.

in Fig. 1 as

$$f(R) = \sum_{i=1}^{n} h_i \delta(R - b_i), \tag{1}$$

where b_i is the i^{th} PW-JND which we need to predict, h_i is the difference between two adjacent quality levels, and $\delta(\cdot)$ is defined as

$$\delta(x) = \begin{cases} 0, & x > 0 \\ 1, & x \leq 0. \end{cases} \tag{2}$$

The PW-JND can be described as bit rate or other metrics, *e.g.*, Quality Factor (QF), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index Measurement (SSIM). Therefore, we introduce a monotone increasing function $g(R)$ to map R to K domain shown in Fig. 3. K can be continuous (*e.g.*, PSNR) or discrete (*e.g.*, QF). In this paper, K is the index of a distorted image, which denotes that we predict the PW-JND in a discrete domain. Now, we replace the perceptual distortion function $f(R)$ with $E(K)$ in the discrete K domain as

$$E(K) = \sum_{i=1}^{n} h_i \Psi(K - K_i), \quad K \in [1, \ldots, m], \tag{3}$$

where K_i is the index of the i^{th} PW-JND we need to predict, and $\Psi(\cdot)$ is defined as

$$\Psi(x) = \begin{cases} 0, & x \in Z^+ \\ 1, & x \notin Z^+, \end{cases} \tag{4}$$

where Z^+ denotes positive integer, which is the definition of symbol Z^+. It is clear that the K_i is a positive integer here which ranges from 1 to m. Therefore, the task of predicting the i^{th} PW-JND of image x can be formulated as a multi-class classification problem. It can be described as

$$K_i = \Omega(x), \tag{5}$$

where the input x is a predicting image, and the output K_i is the i^{th} PW-JND of x, which is considered as a class label that belongs to $[1, \ldots, m]$ (m is the number of classes). In the following section, we will introduce how to predict the i^{th} PW-JND in detail.

III. Proposed Framework of PW-JND Prediction Model

A. Classification Modeling

In the previous section, we modelled the task of predicting PW-JND for a given image x as a multi-class classification problem. The straightforward method to solve the multi-class classification problem is to construct a multi-class classifier $\Omega(x)$ shown in Fig. 4(a). The circle denotes the multi-class classifier and rectangles denote the classes. This model is severely limited to the training data. The multi-class classification problem was usually converted to a combination of binary classification problems, *e.g.*, one-versus-all, one-versus-one, and hierarchy combinations. The hierarchy combination is the most popular model due to its superior performance. One-All (O-A) hierarchy and binary hierarchy model are the most used hierarchy models, which are illustrated in Fig. 4(b) and Fig. 4(c) respectively. The circles denote binary classifiers and rectangles denote classes. O-A hierarchy needs $(m-1)$ binary classifiers, which often suffers from the problem of sample imbalance. The binary hierarchy needs 2^{L-1} binary classifiers, where L is number of the levels and $L = \log_2 m$. For the above models, it is required that each class has enough data to train the classifiers. Due to the shortage of training data samples, the above models can hardly employ deep learning tools directly which achieved impressive success in both high level [32] and low level [33] computer vision tasks. As shown in Fig. 4(d), we propose a PW-JND model consisting of an input part, a binary classifier $\Phi(x, Dist_i)$ denoted by the circles, a search strategy, and a output part. The input part comprises a test image x and a distorted image set $\mathbf{D}$ consisting of distorted images $Dist_i$, where i denotes the image index belonging to $[1, \ldots, m]$. The binary classifier $\Phi(x, Dist_i)$ is designed to predict whether a distorted image $Dist_i$ is perceptually lossy from the test image x or not. The search strategy will be used to predict PW-JND based on the prediction results of the binary classifier. The output part is PW-JND prediction result. The proposed PW-JND model can be used to predict PW-JND for test image x under different types of distortion. In this work, we predict PW-JND for x under JPEG compression. $Dist_i$ is a JPEG-compressed image with QF i, and bigger QF value means higher quality.

For comparison, the required binary classifiers, comparison times, and mean accuracy of the above four models are listed in Table I. From the second column, we can see that the proposed PW-JND model needs to train only one binary classifier, which is the biggest advantage. From the third column, we can see that the proposed PW-JND model needs m computing times in predicting stage, which has the maximum time cost. In order to obtain the mean accuracy, we assume m as the number of classes, χ_i as the probability of class i. For the multi-class classifier $\Omega(x)$, e is assumed as the accuracy. For O-A hierarchy model, we assume the accuracy of classifier C_i to be e_i, and the mean accuracy is about $\sum_{i=1}^{m} \chi_i e_i$. For the binary hierarchy model, $e_{j,q}$ is assumed as the accuracy of the

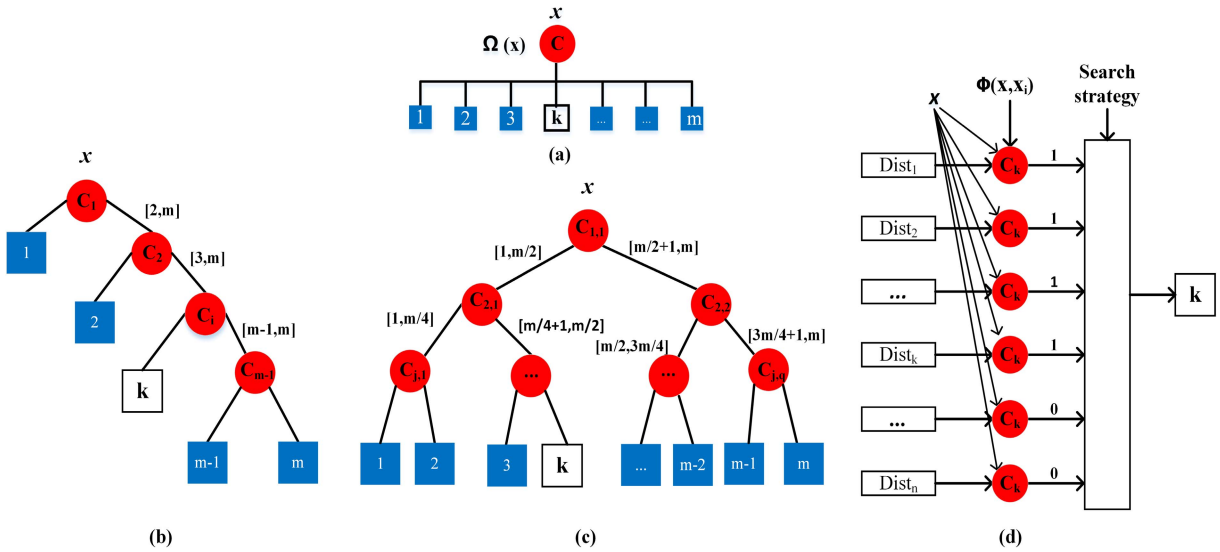


Fig. 4. Different decompositions of multi-class classification problem. (a) Multi-class classifier. (b) O-A hierarchy decomposition. (c) Binary hierarchy decomposition. (d) Proposed model.

TABLE I

COMPARISON OF DIFFERENT DECOMPOSITIONS OVER REQUIRED CLASSIFIERS, COMPARE TIMES, AND MEAN ACCURACY

Decompositions	NO. Classifiers	Compare times(min,max,avg)	Mean accuracy
multi-class classifier	1	(1,1,1)	e
O-A Hierarchy	m-1	(1, m-1, (m-1)/2)	$\sum_{i=1}^{m} \chi_i e_i$
Binary Hierarchy	2^{L-1}	(L, L, L)	$\prod \chi_i e_{j,q}$
Proposed	1	m	$1-\tau(e_k, p, \varepsilon)$

classifier $C_{j,q}$, where j represents the j^{th} layer and q represents the q^{th} node in the j^{th} layer. The mean accuracy can be computed as $\prod \chi_i e_{j,q}$, $j \in [1, L], q \in [1, 2^L]$ (L denotes the levels). $e_{j,q}$ is the accuracy of the binary classifier between class i to $C_{1,1}$. The mean accuracy of proposed model is about $(1-\tau(e_k, p, \varepsilon))$. $\tau(e_k, p, \varepsilon)$ is the error rate which can be nearly obtained by

$$\begin{aligned}\tau(e_k, p, \varepsilon) \approx\ & (1-e_k) + 3(1-e_k)e_k^{p-1} \\ & + (2p+3)(1-e_k)^2 e_k^{p-2} \\ & + \frac{(p-1)!}{(\varepsilon-1)!(p-\varepsilon-1)!}(1-e_k)^{p-\varepsilon+1} e_k^{\varepsilon-1}. \quad (6)\end{aligned}$$

e_k is the mean accuracy of the perceptually lossy/lossless predictor C_k. p ($p \geq 1$) and ε ($\varepsilon \leq p$) are window size and threshold of the sliding window, which will be described in Section V-A in detail. If the accuracy of each classifier in the above models is assumed to be v and the probability of each class is assumed to be equal, the mean accuracy is v, $\frac{1}{m}\sum_{i=1}^{m} v$, $\prod_{i=1}^{L} v$, and $(1-\tau(v, p, \varepsilon))$ for multi-class classifier, O-A hierarchy, binary hierarchy, and the proposed model respectively. Although the mean accuracy of the proposed model is not the highest, it overcomes the limitation of PW-JND data problem and just needs one binary classifier at the cost of more computing time in predicting phase.

In general, there are several advantages of the proposed PW-JND model. First of all, the proposed model just needs to train one binary classifier (perceptually lossy/lossless predictor), which simplifies the problem of predicting PW-JND. Moreover, for the proposed perceptually lossy/lossless predictor, training data are perceptually lossy/lossless samples. The number of lossy/lossless samples is many times that of PW-JND samples. It can be said that the proposed model augments the available samples effectively in an indirect way, which is helpful for deep learning. Secondly, the proposed search strategy can tolerate some mistakes made by the perceptually lossy/lossless predictor. Thirdly, the proposed model can predict all PW-JNDs for the test image x, which denotes that the proposed model can predict PW-JND for distorted images. The PW-JND model also has some shortcomings, *e.g.*, it needs m comparison times and a distorted image set $\mathbf{D}$ in addition to the test image x.

B. The Framework of the Proposed PW-JND Prediction Model

Fig. 5 shows the proposed framework of PW-JND model which includes a training and predicting stage. At the training stage, we need to train a patch-based perceptually lossy/lossless predictor $\Phi(x, Dist_i)$ built on Convolutional Neural Network (CNN), which can predict whether a distorted image is perceptually lossy from its reference or not. The proposed perceptually lossy/lossless predictor includes the following blocks: 1) patch selection module selects the patches from the reference and distorted images; 2) CNN-based feature extractor is to extract distinguished features from the selected patches; 3) patch feature fusion block is responsible for concatenating the features that extracted from the distorted and reference image; 4) patch-wise quality measure block measures the quality of selected patches from the distorted image; 5) picture-wise perceptually lossy/lossless predictor uses the

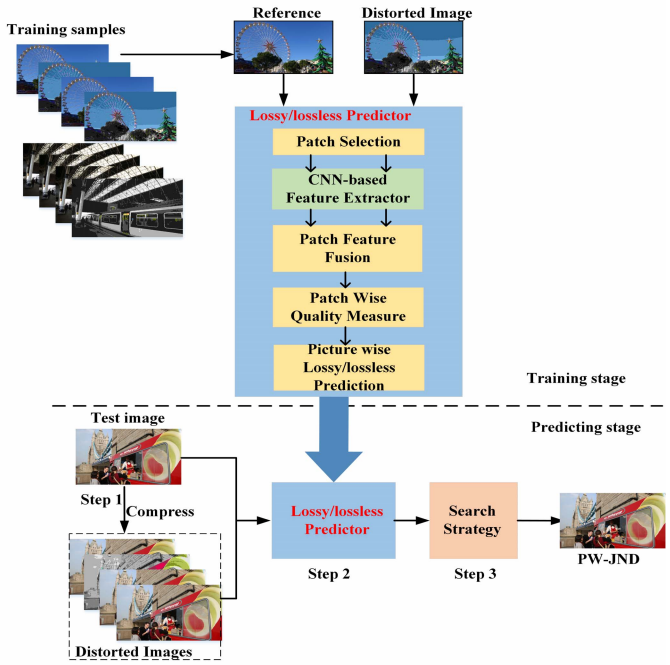


Fig. 5. Framework of the proposed PW-JND model.

patch-wise quality index to classify $Dist_i$ as perceptually lossy or lossless categories.

The predicting stage includes three steps. Firstly, the test image x should be compressed with different quality factors to obtain distorted image set $\mathbf{D}$, where the number of quality levels depends on the distortion type. In this work, we take JPEG coder to compress x with QF ranging from 1 to 100 and obtain $\mathbf{D}$ including 100 JPEG-compressed images with different quality. Secondly, each distorted image $Dist_i$ in $\mathbf{D}$ will be compared with the test image x by the well trained perceptually lossy/lossless predictor. The prediction class label of $Dist_i$ will be one if $Dist_i$ is perceptually lossy from the test image x, otherwise zero. Thirdly, the prediction results of the perceptually lossy/lossless predictor will be fed into the search strategy which is designed to predict PW-JND finally.

IV. The CNN Based Perceptually Lossy/Lossless Predictor

A. Convolutional Neural Networks Architecture

The proposed perceptually lossy/lossless predictor based on deep CNN is trained for predicting whether a distorted image is perceptually lossy from its reference or not. As shown in Fig. 6, the predictor consists of a patch selection strategy, a Local Quality Assessment Network (LQAN), and a Global Classifier Network (GCN). The LQAN includes the feature extractor, patch feature fusion, and local quality measure block.

The configurations of the LQAN and GCN are shown in Table II. The distorted and its reference image are firstly divided into patches with size $M \times M$, and N patches are selected from the reference and distorted image in the same location respectively. The patch size M and patch number N are parameters which will be discussed later. We borrow

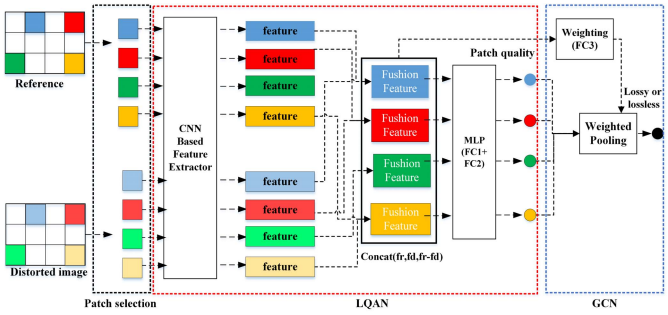


Fig. 6. Architecture of the proposed perceptually lossy/lossyless predictor.

TABLE II
Configurations of the Perceptually Lossy/Lossless Predictor Network

Network	Layer	Type	Kernel	Stride	Outputs
LQAN	1	Conv.	3×3	2×2	32
	2	Conv.	3×3	2×2	32
	3	Conv.	3×3	2×2	64
	4	Conv.	3×3	2×2	64
	5	Conv.	3×3	2×2	128
	6	Conv.	3×3	2×2	128
	7	Conv.	3×3	2×2	256
	8	Conv.	3×3	2×2	256
	9	Conv.	3×3	2×2	512
	10	Conv.	3×3	2×2	512
	Concat	–	–	–	1536
	FC1	FC	–	–	512
	FC2	FC	–	–	1
GCN	FC3	FC	–	–	1

the architecture from [34] to build LQAN which consists of ten Convolutional (Conv.) layers, one Concatenation (Concat) layer, and two Full Connected (FC) layers. Each convolutional layer is activated by the Rectified Linear Unit (ReLU) [35], and a max pooling layer follows each two convolutional layers. A concatenation layer following the ten convolutional layers is to concatenate the features learned from the distorted and reference patches. The two FC layers FC1 and FC2 adopt dropout regulation with ratio 0.5. The output of the LQAN is the quality score of a selected distorted patch, and larger score denotes worse quality here. The GCN is a binary classification network which includes one FC layer (FC3) activated by sigmoid function. The dimension of FC3 is the same as the patch number N. Each element (a positive value) in FC3 is the weight of a selected patch, which denotes the contribution of the corresponding patch to the whole image quality. The weights are initialized the same as [34] and updated with LQAN simultaneously. The sigmoid function in the last layer rescales the output among [0, 1]. If the output is larger than 0.5, the distorted image is determined as perceptually lossy from the reference, otherwise lossless.

B. Loss Function

Let $F(x_t, \theta)$ denotes the end-to-end mapping function, in which x_t is the input image pair and θ is the set of weights. Therefore, we need to estimate θ mapping x_t to its ground truth label. We optimize θ by minimizing the cross-entropy loss. Given a set of n training sample pairs (x_t, y_t), where x_t consists of a reference image and a distorted image, and

y_t is the ground truth label. We update θ by minimizing the following loss function

$$L(\theta) = -\frac{1}{n}\sum_{t=1}^{n}[y_t \log F(x_t, \theta) + (1 - y_t) log(1 - F(x_t, \theta))]. \tag{7}$$

C. Dataset Generation

MCL-JCI dataset [23] is the first PW-JND based image quality dataset. This dataset comprises 50 pristine images, and each pristine image has 100 JPEG-compressed images with different QF ranging from 1 to 100. Most of the pristine images have 4 to 8 PW-JND images found by subjects, each of which is the transitional image between two adjacent perceptual quality levels. In order to train the proposed predictor, we make use of PW-JND images to generate perceptually lossy/lossless training samples. A perceptually lossy/lossless sample can be described as (x_t, y_t) which consists of image data x_t and label y_t: 1) x_t consists of the reference $ref_{k'}$ and distorted image $Dist_i$, where $ref_{k'}$ is the k^{th} PW-JND image with QF value of k' (larger k denotes smaller QF), i is the QF value of its corresponding distorted images $Dist_i$, which ranges from 1 to $(k'-1)$. In particular, $ref_{0'}$ represents the pristine image, and the QF value i of its corresponding distorted images ranges 1 to 100. 2) y_t is labeled as one when i ranges 1 to $(k+1)'$ denoting the QF value of the $(k+1)^{th}$ PW-JND image, which denotes $Dist_i$ is perceptually lossy to $ref_{k'}$, otherwise y_t is labeled as zero, which denotes $Dist_i$ is perceptually lossless from $ref_{k'}$. The first, second, third, and fourth ground truth PW-JND in MCL-JCI data set were used as reference images to generate training samples. Finally, we generated 5003 positive and 3974 negative image samples.

For the convenience of training, the generated data set was split into five subsets. Firstly, the 50 pristine images were randomly divided into five equal groups I_1, I_2, I_3, I_4, and I_5, each of which includes 10 images. Then, the samples generated from the images in the same group were added into one subset. Finally, the generated data set was split as $\{S_1, S_2, S_3, S_4, S_5\}$ according to the division of the pristine images $\{I_1, I_2, I_3, I_4, I_5\}$. There is no overlap among the five subsets.

D. Validation and Optimal Parameters Determination

Each image fed into the proposed perceptually loss/lossless predictor is represented by N patches with size $M \times M$. The patches share the convolutional neural networks, and each patch can be seen as a sample. Therefore, the training samples are many times that of image samples. Random cropping data augmentation was used in training, all of the patches were randomly cropped from the image every epoch to ensure that as many different patches as possible can be used [34]. In validation, the N random patches for each image were only sampled once at the beginning of training in order to avoid noise. In the experiment, patch size $M \times M$ was set to 32×32 and the number of patches N was set to 32. The variable-controlling approach was taken to determine M and N, which will be described in detail in the following paragraph. Truncated normal initializer was used to initialize the network weights and Adam [36] was taken as the gradient descent optimization method. The learning rate was initialized as 1×10^{-4}, which decreases as the number of iterations increases. Each mini-batch contains 4 images, each represented by 32 randomly sampled image patches, which leads to the effective batch size of 128 patches. All of the training parameters were updated one time when a mini-batch was processed. It is worth noting that the proposed predictor is a patch-based network which predicts an image-wise result by pooling the quality of the selected patches. Therefore, the selected patches cannot cross mini-batch. We implemented the networks based on the Tensorflow 1.2.0 and Python 3.5.2, then trained them on the machine with NVIDIA GTX1080Ti GPU and memory 32G.

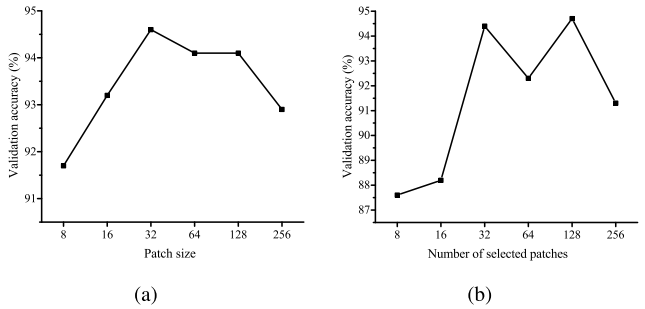
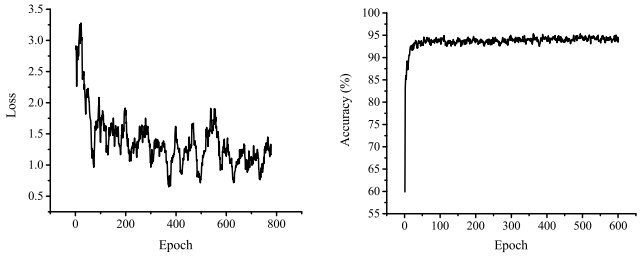


Fig. 7. Validation accuracy of the perceptually lossy/lossless predictor with different patch sizes and numbers of selected patches. (a) Different patch sizes. (b) Different numbers of selected patches.

The patch size M and number of patches N are two key factors for the proposed perceptually lossy/lossless predictor. However, there are so many combinations of (M, N) that it is difficult to find the global optimal combination through exhaustive searching methods. Firstly, the number of patches N was fixed as 32 [34], the other variables was controlled stable except patch size. The training and validation set are randomly selected from the generated data set $\{S_1, S_2, S_3, S_4, S_5\}$. Six patch sizes, $M \in [8, 16, 32, 64, 128, 256]$, were tested. The validation accuracy is shown in Fig. 7(a), where x-axis denotes patch size and y-axis denotes validation accuracy. It can be seen that the validation accuracy is improved with the increase of patch size when the patch size is not beyond 32. The reason lies in the quality of a larger block is closer to the quality of the entire image. However, the situation is just the opposite when the patch size is larger than 64. The reason may be that the parameters of the networks increase exponentially with the increase of the patch size, which means that we need more training data. Secondly, patch size M was fixed as 32 (32×32), and $N \in [8, 16, 32, 64, 128, 256]$ were tested. The validation accuracy is shown in Fig. 7(b), where x-axis denotes patch size and y-axis denotes validation accuracy. The accuracy is low when N is 8 and 16, and it reaches a high value when N is 32. The reason may be that a very small number of selected patches ($N = 8, 16$) can hardly represent the whole image quality. It can also be observed that there is not a significant gain in validation accuracy with the increase of N when N is larger than 32. Since a larger N brings more



(a) (b)

Fig. 8. Training peformance of the perceptually lossy/lossless predictor. (a) Training loss. (b) Validation accuracy.

TABLE III
THE VALIDATION AND TEST ACCURACY IN FIVE-FOLD CROSS VALIDATION

S_{tr}	S_{va}	S_{te}	Predictor	acc_v (%)	acc_t (%)
$\{S_2, S_3, S_4\}$	S_5	S_1	D_1	90.7	91.1
$\{S_3, S_4, S_5\}$	S_1	S_2	D_2	90.8	94.1
$\{S_4, S_5, S_1\}$	S_2	S_3	D_3	93.5	90.0
$\{S_5, S_1, S_2\}$	S_3	S_4	D_4	94.4	93.3
$\{S_1, S_2, S_3\}$	S_4	S_5	D_5	90.5	91.3
—	—	—	average	91.98	91.96

computation time, we selected the number of patches N as 32. Finally, we selected 32×32 as the patch size and 32 as the number of selected patches in this work.

We randomly chose three subsets for training, one for validation, and the rest one for testing from the generated dataset. The training loss is shown in Fig. 8(a), where x-axis and y-axis denote the training epoch and the training loss respectively. We can see that the training loss drops down rapidly at the beginning, and after about 80 training epochs the loss fluctuates slightly in the later training epochs. The validation prediction accuracy is shown in Fig. 8(b), where x-axis and y-axis denote training epoch and validation accuracy respectively. We can see that the validation accuracy rises rapidly at the beginning and keeps stable later. From the above observations, we can conclude that the predictor converges at last.

E. Testing of the Perceptually Lossy/Lossless Predictor

In order to further evaluate the generalization capabilities of the proposed perceptually lossy/lossless predictor, five-fold cross-validation was made in this work. Firstly, three subsets were selected to train the predictor, one for validation, and the other for testing from the generated data set $\{S_1, S_2, S_3, S_4, S_5\}$. Once a stable predictor was well trained, we rotated the test set, and finally we trained five predictors D_1, D_2, D_3, D_4, and D_5. The cross validation results are shown in Table III, where acc_v and acc_t denote the mean validation accuracy and testing accuracy. S_{tr}, S_{va}, S_{te} are training, validation, and testing set respectively. We can see from Table III that: 1) All of the validation and test accuracy is over 90%. The mean validation and test accuracy are 91.98% and 91.96% respectively, which denotes that the accuracy of the predictor is high. 2) The validation accuracy is close to the corresponding test accuracy, which means a stable performance. In this work, the five predictors D_1-D_5 were trained one time on the entire generated data set including samples generated from the first to fourth ground truth PW-JND. There is no need to re-train them in predicting the first to fourth PW-JND.

V. PROPOSED PW-JND SEARCH STRATEGY

A. Sliding Window Based PW-JND Searching

If the accuracy of the perceptually lossy/lossless predictor is assumed to be 100%, we can obtain the ideal case shown in Fig. 9(a), in which x-axis represents distorted images $Dist_i$ (larger i means better quality), and y-axis denotes the labels predicted by the perceptually lossy/lossless predictor $\Phi(ref, Dist_i)$. $y' = 1$ (or 0) denotes the distorted image $Dist_i$ is perceptually lossy (or lossless) from the reference ref. The distorted image $Dist_k$ can be determined as the PW-JND image and the corresponding index k can be predicted as the PW-JND, which satisfy: 1) $H(\upsilon) = 1$ when $\upsilon \in [0, \ldots, k-1]$, where $H(\upsilon)$ is defined as

$$H(\upsilon) = \Phi(ref, Disk_{k-\upsilon}); \tag{8}$$

2) $T(n) = 0$ when $n \in [1, \ldots, m-k]$, where $T(n)$ is defined as

$$T(n) = \Phi(ref, Disk_{k+n}). \tag{9}$$

It is easy to locate the PW-JND image by searching from right to left (or left to right) when all of the υ (or n) distorted images are predicted accurately by the perceptually lossy/lossless predictor. However, it can hardly predict all υ (or n) distorted images accurately, since the predictor may have a error prediction. It can easily cause that the predicted PW-JND thresholds differ greatly from the ground truth. Therefore, we propose a sliding window based PW-JND search method shown in Fig. 9(b) to determine the PW-JND image. We slide the window from right to left, and determine the distorted image $Dist_k$ as the PW-JND image which satisfies

$$\sum_{0}^{p} H(\upsilon) \geq \varepsilon, \tag{10}$$

where p is the window size, and ε is a given threshold. The proposed PW-JND search strategy can tolerate some mistakes of the perceptually lossy/lossless predictor. Take Fig. 9(b) as an example, the window size p is set as 5 and ε is set as 4. Although the point A and C are predicted incorrectly, it will not affect the result that point B will be determined as the PW-JND image, which is consistent with the ground truth. The mean accuracy of the proposed PW-JND model can be derived from (6), and we can see that the accuracy of the perceptually lossy/lossless predictor is a key factor, and thresholds p and ε are also important factors.

B. The Parameter Determination for the Sliding Window

The window size p and threshold ε affect the performance of the PW-JND search strategy. In order to select a suitable combination of p and ε, we selected D_3 as the perceptually lossy/lossless predictor, I_3 as the test image set. The aim is to ensure that the test images are randomly selected from MCL-JCI dataset and the test images have never been seen by the perceptually lossy/lossless predictor. We predicted the

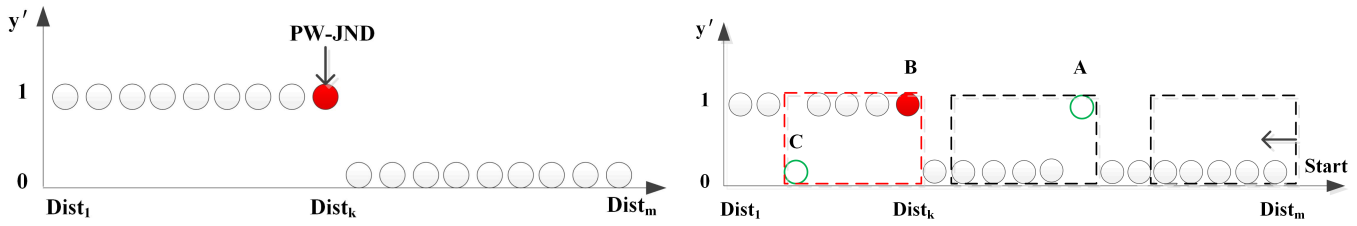


Fig. 9. PW-JND image searching. (a) Ideal case. (b) The proposed strategy.

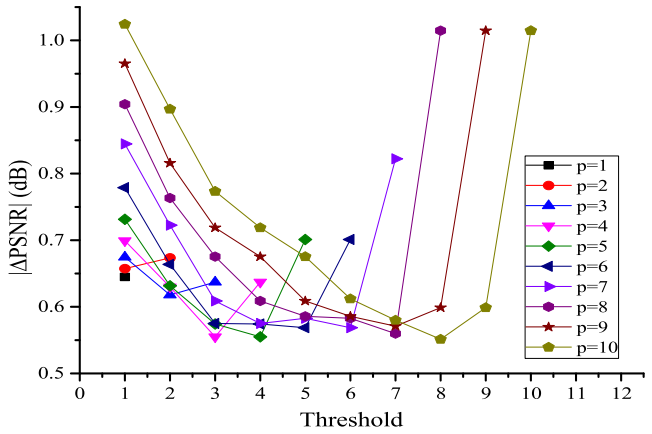


Fig. 10. The performance of the proposed sliding window based PW-JND searching strategy with different window sizes and thresholds.

first PW-JND of the test images by fixing window size p, then changing ε ($\varepsilon \leq p$). In order to reduce the computational complexity, ten window sizes $p \in [1, 2, \ldots, 9, 10]$ were tested in this work. The prediction results are shown in Fig. 10, where x-axis represents threshold ε and y-axis represents $|\Delta\text{PSNR}|$. ΔPSNR is the value of ground truth PSNR minus prediction PSNR, and $|\cdot|$ is absolute value operator. A smaller $|\Delta\text{PSNR}|$ denotes a better performance. Each line represents a window size p ($p = 1$ is only one point), and each point denotes a different threshold in a fixed p. It can be seen that: 1) For each fixed window size p, there is an inflection point with the smallest $|\Delta\text{PSNR}|$. On the left of the inflection point, $|\Delta\text{PSNR}|$ decreases with the increase of ε. The reason may be that a very small ε can easily result in an underestimation, which means the predicted PSNR is larger than the ground truth PSNR and ΔPSNR is negative. The underestimation becomes small with the increase of ε. It means that ΔPSNR becomes larger, but $|\Delta\text{PSNR}|$ becomes smaller. On the right of the inflection point, $|\Delta\text{PSNR}|$ increases with the increase of ε. The reason may be that a large ε may result in an overestimation, which means the predicted PSNR is smaller than the ground truth PSNR and ΔPSNR is positive. The overestimation increases when ε becomes bigger, both of ΔPSNR and $|\Delta\text{PSNR}|$ become larger. The inflection point can be seen as the boundary between underestimation and overestimation. 2) The $|\Delta\text{PSNR}|$ of different inflection points of different window sizes p (except $p = 1, 2, 3$) are very close. It means that there are many different parameter combinations (p, ε) to be chosen, such as $(4, 3)$, $(5, 4)$, $(6, 5)$ and so on. For such candidate parameter combinations, the prediction accuracy is similar. We select $p = 6$ and $\varepsilon = 5$ in our experiment.

VI. Experimental Results and Analyses

A. Experimental Settings

The performance of the proposed PW-JND model was evaluated on MCL-JCI data set mentioned in Section IV-C. The five well trained perceptually lossy/lossless predictors D_1 to D_5 mentioned in Section IV-E were used to predict PW-JND for I_1 to I_5 mentioned in Section IV-C, respectively. The aim is to ensure that the test images have never been seen by the perceptually lossy/lossless predictors. The prediction results of the 50 images were obtained by combining the prediction results of the five predictors. It is worth noting that I_3 has been used to select parameters for the proposed PW-JND search strategy in Section V-B. The five predictors were shared in predicting the first and second PW-JND. In predicting the first PW-JND, the test images x was set to the pristine image, and the distorted image set $\mathbf{D}$ consists of the 100 JPEG-coded images. In predicting the second PW-JND, x was set to the first ground truth PW-JND image, and $\mathbf{D}$ consists of all of the JPEG-compressed images with smaller QF than that of the first ground truth PW-JND image. The parameters p and ε of the proposed sliding window based search strategy were set the same ($p = 6, \varepsilon = 5$) in predicting the first and second PW-JND.

In order to compare the performance among the proposed PW-JND model and conventional pixel domain JND models, the Free Energy Principle based Pixel domain JND (FEP_PJND) model [11] and Enhanced Pattern Complexity based Pixel domain JND (EPC_PJND) model [14] were selected as comparison models. They estimate the JND threshold for each pixel and return a JND thresholds map. Since the pixel domain models estimate JND threshold for each pixel but not the whole image, we devise a method for the comparison models to predict PW-JND of the test image x as: 1) $Z(x, Dist_i)$ is designed to predict whether a distorted image $Dist_i$ is perceptually lossy from x or not, of which the function is the same as that of the proposed perceptually lossy/lossless predictor $\Phi(x, Dist_i)$ in Fig. 4(d). 2) Each JPEG-compressed image in the distorted image set $\mathbf{D}$ will be compared with x, and $Z(x, Dist_i)$ is designed to output 1 if $Dist_i$ is perceptually lossy from x, otherwise 0. 3) The distorted image with prediction label 1 and the largest QF will

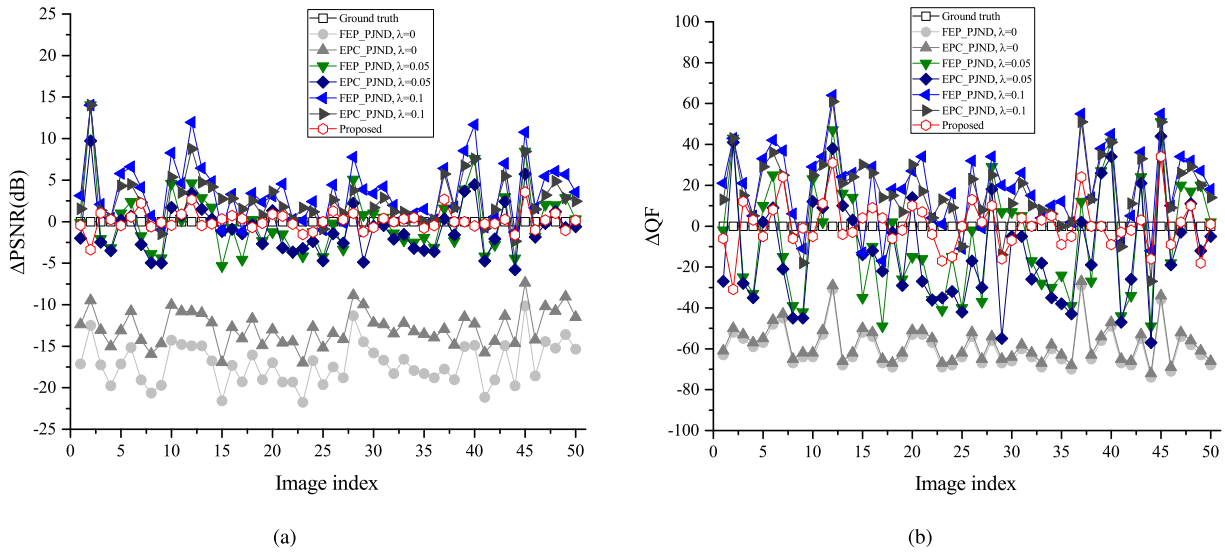


Fig. 11. Performance comparison in predicting the first PW-JND. (a) PSNR. (b) QF.

be determined as the PW-JND for x. $Z(x, Dist_i)$ is defined as

$$Z(x, Dist_i) = \begin{cases} 0, & S \leq T \times \lambda \\ 1, & S > T \times \lambda, \end{cases} \tag{11}$$

where T is the number of pixels of x, S is the number of the pixels that changes over the corresponding JND thresholds, and λ is a given threshold. S is defined as

$$S = \sum_{i=1}^{m} \sum_{j=1}^{n} U_{i,j}, \tag{12}$$

where i, j is the pixel index, $U_{i,j}$ describes whether the change of pixel $Dist(i, j)$ is over its JND threshold or not. $U_{i,j}$ is defined as

$$U_{i,j} = \begin{cases} 0, & D_{i,j} \leq M_{i,j} \\ 1, & D_{i,j} > M_{i,j}, \end{cases} \tag{13}$$

where i, j is the pixel index, $D_{i,j} = ref_{i,j} - Dist_{i,j}$, and $M_{i,j}$ is the estimated JND threshold calculated by pixel domain JND models for pixel $x_{i,j}$. The test image x and distorted image set **D** were set the same as the proposed model mentioned in the previous paragraph. The JND map M was obtained by pixel domain JND model from the test image x. In predicting the first and second PW-JND, λ was set as 0, 0.05, and 0.1. Particularly, $\lambda = 0$ denotes that as long as there is any one pixel changing over its JND threshold in the distorted image, it will be considered as a perceptually lossy image.

QF, PSNR, SSIM, Feature Similarity Index Measurement (FSIM), Gradient Magnitude Similarity Deviation (GMSD) [29], Visual Saliency Index (VSI) [30], and Perceptually Weighted Mean Squared Error (PWMSE) [37] metrics were selected to describe PW-JND. The difference in the above metrics between the predicted PW-JND and the ground truth PW-JND was selected as the evaluation index. For example, ΔQF is the result of ground truth QF minus prediction QF. A positive ΔQF denotes an underestimation, a negative ΔQF means an overestimation, and $\Delta QF = 0$ denotes the estimation is consistent with the ground truth. For a further analysis, the absolute difference ($|\cdot|$) was selected as another evaluation index. For example, |ΔQF| is the absolute value of ΔQF. A larger |ΔQF| means a greater prediction error.

B. Performance of the Proposed PW-JND Prediction Model Evaluation

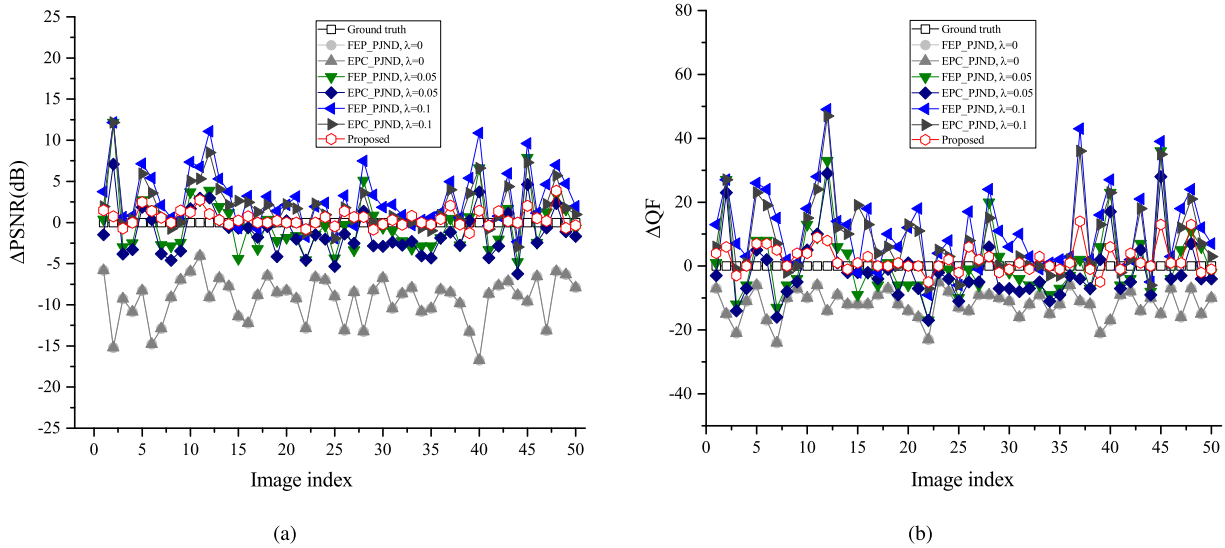
1) Predicting the First PW-JND: The prediction results of the first PW-JND are plotted in Fig. 11(a) and Fig. 11(b), where x-axis represents image index, and y-axis in Fig. 11(a) and Fig. 11(b) represent ΔPSNR and ΔQF respectively. From Fig. 11(a), it can be observed that: 1) When $\lambda = 0$, all of the ΔPSNR values of FEP_PJND and EPC_JND are below x-axis, which denotes all the PW-JNDs were underestimated. On the other hand, when $\lambda = 0.1$, all of the ΔPSNR values of the two comparison models are over x-axis, which denotes all the PW-JNDs were overestimated. 2) Most of ΔPSNR values of the proposed PW-JND are very close to zero, which means the proposed model has a high prediction accuracy. 3) Compared with the two pixel domain models, all ΔPSNR values of the proposed PW-JND are closer to zero, which denotes the proposed PW-JND model has the highest prediction accuracy. From Fig. 11(b), we can also come to the conclusion that the proposed model has the highest accuracy.

For a further comparison, the mean and variance of the absolute difference are listed in Table IV. From the mean part, it can be observed that the mean of |ΔQF| and |ΔPSNR| of the proposed PW-JND are 8.7 and 0.8 respectively, which are the smallest among all of the compared models. The similar phenomenon can be obtained in other metrics. Therefore, we can conclude that the accuracy of the proposed model is the highest. From the variance part in Table IV, we can see that all of variance values of the proposed model are the smallest, which denotes the proposed model is the most stable one. Therefore, we can conclude that the proposed model performs best in predicting the PW-JND for pristine images.

TABLE IV

COMPARISON OF THE FIRST PW-JND PREDICTION IN TERMS OF THE MEAN AND VARIANCE OF ABSOLUTE PREDICTION DIFFERENCE

Index	λ	Models	$\lvert\Delta QF\rvert$ $(\times 10^{1})$	$\lvert\Delta PSNR\rvert$	$\lvert\Delta SSIM\rvert$ $(\times 10^{-2})$	$\lvert\Delta FSIM\rvert$ $(\times 10^{-3})$	$\lvert\Delta GMSD\rvert$ $(\times 10^{-2})$	$\lvert\Delta VSI\rvert$ $(\times 10^{-3})$	$\lvert\Delta PWMSE\rvert$
Mean	0	FEP_PJND [11]	5.94	17.1	2.05	3.66	1.67	1.51	0.41
		EPC_PJND [14]	5.74	12.8	1.90	3.00	1.60	1.00	0.41
	0.05	FEP_PJND [11]	2.46	2.91	2.30	7.00	1.94	2.00	1.21
		EPC_PJND [14]	2.41	2.51	1.34	5.00	1.30	1.00	1.07
	0.10	FEP_PJND [11]	2.29	3.93	5.00	21.0	4.60	6.00	1.65
		EPC_PJND [14]	2.04	2.95	3.40	12.0	3.00	4.00	1.27
	—	Proposed	**0.87**	**0.82**	**0.40**	**1.00**	**0.40**	**0.40**	**0.37**
Variance	0	FEP_PJND [11]	9.40	6.38	5.76	1.20	3.31	2.68	6.50
		EPC_PJND [11]	9.40	4.36	5.62	1.18	3.31	2.66	6.51
	0.05	FEP_PJND [11]	20.2	5.72	205	33.1	72.7	226	83.4
		EPC_PJND [14]	22.0	3.46	24.4	28.0	15.6	41.3	54.1
	0.10	FEP_PJND [11]	22.1	11.3	349	76.7	195	523	165
		EPC_PJND [14]	18.0	6.96	243	38.1	101	261	104
	—	Proposed	**6.80**	**0.66**	**1.24**	**0.877**	**1.17**	**1.02**	**0.0012**



(a) (b)

Fig. 12. Performance comparison in predicting the second PW-JND. (a) PSNR. (b) QF.

2) Predicting the Second PW-JND: The prediction results of the second PW-JND are plotted in Fig. 12(a) and Fig. 12(b). It can be seen that the phenomenon of Fig. 12(a) is very similar to that shown in Fig. 11(a). We can obtain the following conclusions: 1) When $\lambda = 0$ all the ΔPSNR were underestimated by the two comparison models, and when $\lambda = 0.1$ all the ΔPSNR were overestimated by the two comparison models. 2) The ΔPSNR values of the proposed model are closer to zero than that of the two comparison models, which denotes that the proposed model has the highest accuracy. Another phenomenon is that the ΔPSNR values are closer to ground truth compared with the first PW-JND thresholds predicting, which denotes it is easier to predict the second PW-JND than to predict the first PW-JND. The reason may be that the degradation between the distorted image and test image x is becoming larger. The conclusions also can be convinced by Fig. 12(b), and we will not give a further analysis for Fig. 12(b).

The mean and variance of the absolute difference are listed in Table V. From the mean part, we can see that the mean of $|\Delta QF|$, $|\Delta PSNR|$, and $|\Delta SSIM|$ of the proposed PW-JND are 3.14, 0.76, and 0.53×10^{-2}, which are the smallest compared with the other models. From the variance part, it can be seen that the variances of $|\Delta QF|$, $|\Delta PSNR|$, and $|\Delta SSIM|$ of the proposed model are 12.3, 0.65, and 0.29×10^{-4}, which are also the smallest. Compared with the two comparison models with different thresholds in different metrics, the mean and variance of the proposed model are the smallest except the variance of ΔFSIM. Therefore, we can conclude that the proposed model performs best among the comparison models.

C. Visual Quality Comparison

Moreover, subjective quality of the prediction results for "ImageJND_SRC13" and "ImageJND_SRC39" are shown in Fig. 13 and Fig. 14 respectively, in which the enlarged patches are the most quality sensitive regions. Take Fig. 13 as an example, (a) is the source image, (b) to (d) are enlarged patches of QF 100, the first ground truth PW-JND (QF 31), and the prediction result of the proposed model (QF 35), respectively. From the first row, we can see that the perceptual quality of (b), (c), and (d) are very similar meanwhile the image size and PSNR of (c) and (d) are very close. It demonstrates

TABLE V

COMPARISON OF THE SECOND PW-JND PREDICTION IN TERMS OF THE MEAN AND VARIANCE OF ABSOLUTE PREDICTION DIFFERENCE

Index	λ	Models	$\lvert\Delta$QF$\rvert$	$\lvert\Delta$PSNR$\rvert$	$\lvert\Delta$SSIM$\rvert$ $(\times 10^{-2})$	$\lvert\Delta$FSIM$\rvert$ $(\times 10^{-3})$	$\lvert\Delta$GMSD$\rvert$ $(\times 10^{-2})$	$\lvert\Delta$VSI$\rvert$ $(\times 10^{-2})$	$\lvert\Delta$PWMSE$\rvert$
Mean	0	FEP_PJND [11]	11.9	9.27	2.73	6.20	2.46	0.23	2.17
		EPC_PJND [14]	11.9	9.27	2.73	6.20	2.46	0.23	2.17
	0.05	FEP_PJND [11]	8.21	2.66	2.17	7.21	1.87	2.21	0.59
		EPC_PJND [14]	7.38	2.51	1.48	4.52	1.33	2.12	0.46
	0.10	FEP_PJND [11]	12.7	3.44	4.39	19.4	4.13	2.55	1.11
		EPC_PJND [14]	9.94	2.41	2.75	10.6	2.46	2.28	0.75
	—	Proposed	**3.14**	**0.76**	**0.53**	**2.12**	**0.45**	**0.04**	**0.18**
Variance	–	–	$(\times 10^{1})$	$(\times 10^{0})$	$(\times 10^{-4})$	$(\times 10^{-5})$	$(\times 10^{-5})$	$(\times 10^{-6})$	$(\times 10^{0})$
	0	FEP_PJND [11]	1.98	7.46	1.61	**2.14**	8.07	1.03	3.68
		EPC_PJND [14]	1.98	7.46	1.61	**2.14**	8.07	1.03	3.68
	0.05	FEP_PJND [11]	5.86	4.45	17.8	26.2	64.9	19840	0.47
		EPC_PJND [14]	3.98	2.58	1.52	8.43	9.93	19827	0.09
	0.10	FEP_PJND [11]	13.2	9.55	30.6	68.4	167.1	19678	1.16
		EPC_PJND [14]	11.4	6.32	20.5	32.1	93.9	19686	0.74
	—	Proposed	**1.23**	**0.65**	**0.29**	4.12	**1.77**	**0.21**	**0.02**

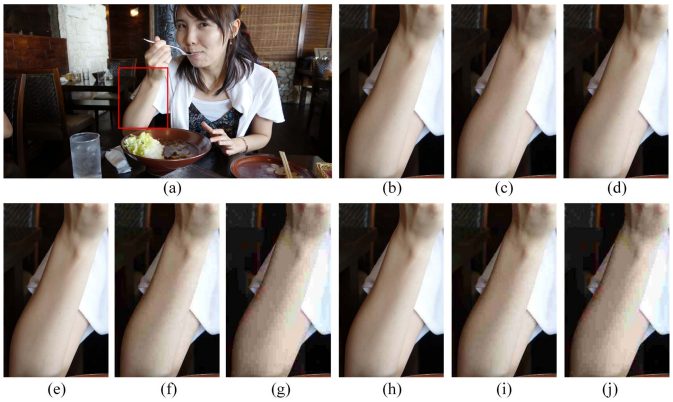


Fig. 13. Visual quality comparison of source image "ImageJND_SRC13" in MCL-JCI, (b)-(j) are enlarged patches, and {*, *, *} denotes QF, image size (KB), PSNR (dB) of the associated images. (a) Original image. (b) Image with best quality under JPEG compression, {100, 1334.1, 50.03}. (c) The first ground truth PW-JND, {31, 105.7, 34.67}. (d) Proposed, {35, 113.7, 35.12}. (e) to (g) are prediction results of EPC_PJND [14] with $\lambda = 0, 0.05, 0.10$, {97, 762.4, 45.69}, {21, 85.4, 33.19}, {10, 59.2, 29.96}. (h) to (j) are prediction results of FEP_PJND [11] with $\lambda = 0, 0.05, 0.10$, {99, 1148.2, 49.63}, {15, 71.8, 31.83}, {7, 51, 28.23}.

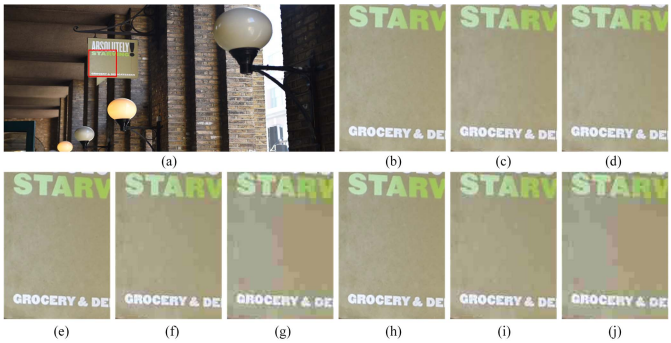


Fig. 14. Visual quality comparison of source image "ImageJND_SRC39" in MCL-JCI, (b)-(j) are enlarged patches, and {*, *, *} denotes QF, image size (KB), PSNR (dB) of the associated images. (a) Original image. (b) Image with best quality under JPEG compression, {100, 1143.4, 49.12}. (c) The first ground truth PW-JND, {44, 132.9, 35.67}. (d) Proposed, {41, 127.4, 35.41}. (e) to (g) are prediction results of EPC_PJND [14] with $\lambda = 0, 0.05, 0.10$, {97, 684.8, 47.03}, {23, 90.9, 33.26}, {11, 61.7, 30.20}. (h) to (j) are prediction results of FEP_PJND [11] with $\lambda = 0, 0.05, 0.10$, {99, 967.9, 48.59}, {20, 83.7, 32.72}, {5, 53.3, 28.66}.

the effectiveness of the proposed model. In the second row, (e) to (g) and (h) to (j) are enlarged images of prediction results of EPC_PJND [14] and FEP_PJND [11] with $\lambda = 0, 0.05, 0.10$, respectively. When $\lambda = 0$, the perceptual quality of (e) and (h) is almost the same as that of (b), however the size of corresponding images is much larger than that of (d). When $\lambda = 0.05$, the distortion of (f) and (i) can be easily perceived by HVS. When $\lambda = 0.10$, the perceptual quality of (g) and (j) is unacceptable. From the second row, we can conclude that the proposed model performs better than EPC_PJND and FEP_PJND model. Similar phenomenon can be seen from Fig. 14.

D. Computational Complexity of the PW-JND Model

The computational complexity of the proposed model mainly includes the time spent in compressing the test image and predicting whether a JPEG-compressed image is perceptually lossy from the test image or not. The time spent in PW-JND searching can be ignored. We used MATLAB code "imwrite (image, imageName, 'jpeg', 'Quality', QF)" to compress the test image, and implemented the predictor on Tensorflow 1.2.0 and Python 3.5.2. All of the tests were finished on a desktop computer with Intel CPU i-7-6700K, GPU GTX1080Ti, and 32G memory. The running time spent in predicting the first PW-JND for the 50 pristine images in MCL-JCI dataset is shown in Table VI. The test sets in the first column are the same as Table III, and each set includes ten 1920×1080 images. As the penultimate row shows, the mean compressing, predicting, and total time of predicting the first PW-JNDs for 50 images are 229.38, 406.88, and 636.26 seconds respectively. We can also see from the

TABLE VI

THE RUNNING TIME SPENT IN PREDICTING THE FIRST PW-JND (SECONDS)

Test set	Compressing time	Predicting time	Total
S_1	45.66	83.31	128.97
S_2	45.73	81.22	126.95
S_3	45.10	80.43	125.53
S_4	48.96	85.02	133.98
S_5	43.86	76.89	120.75
total	229.38	406.88	636.26
Mean time per image	**4.58**	**8.13**	**12.71**

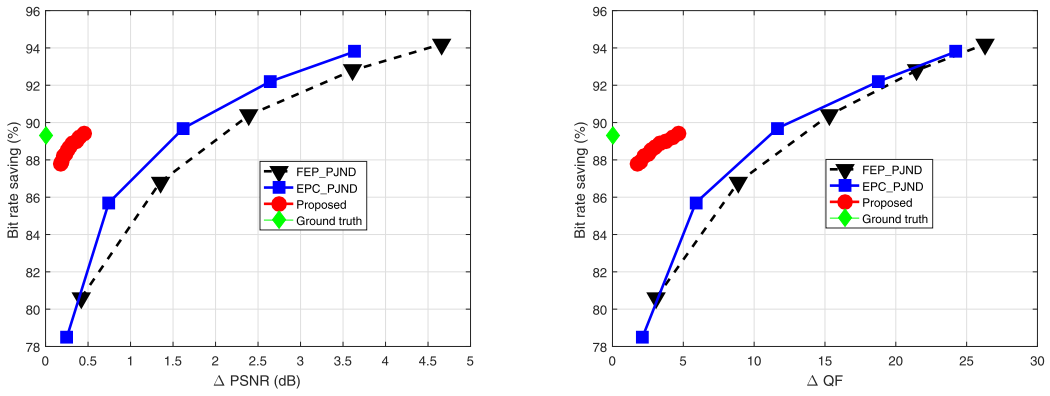


Fig. 15. Bit rate saving versus ΔPSNR and ΔQF of the first prediction PW-JND. (a) Δ PSNR. (b) ΔQF.

last row that predicting the first PW-JND for a pristine image needs about 12.71 seconds (mean) which includes compressing time 4.58 seconds (mean) and predicting time 8.13 seconds (mean). From the above observations, it can be seen that the computational complexity of the proposed model is acceptable. We will focus on reducing the computational complexity in the future work.

E. Application in Image Compression and Transmission

The PW-JND reveals the minimum difference of a picture that HVS can perceive, which can be used for selecting parameters in image compression. We take JPEG coder to compress the 50 images in MCL-JCI dataset with the predicted PW-JND (QF). The mean of bit rate saving C and prediction error ΔQ were designed as evaluation indexes. C is defined as

$$C = \left(R_{te} - R_p\right) \big/ R_{te} \times 100\%, \tag{14}$$

where R_{te} and R_p are the bit rate of the test image x and predicted PW-JND image respectively. In predicting the first PW-JND, x is JPEG-compressed image with QF 100. In predicting the second PW-JND, x is the first ground truth PW-JND image; ΔQ ($Q \in$ PSNR, QF) is defined as

$$\Delta Q = \begin{cases} Q_{tr} - Q_p & Q_{tr} > Q_p \\ 0 & Q_{tr} \le Q_p, \end{cases} \tag{15}$$

where Q_{tr} and Q_p are the ground truth and predicted PW-JND image respectively. When $Q_{tr} > Q_p$, the predicted PW-JND image has a perceptual loss, which is regarded as a prediction error.

The experimental results of the first PW-JND are shown in Fig. 15, where y-axis represents the mean of bit rate saving, and x-axis represents the mean of ΔPSNR and ΔQF in Fig. 15(a) and Fig. 15(b) respectively. In Fig. 15, the diamond denotes the ground truth, the red circles from right to left represent the results of adding $n \in [1, 2, \ldots, 9]$ to the predicted QF of the proposed model. The dotted and solid line denote the FEP_PJND and EPC_PJND model. The squares and triangles represent the results when λ (see (11)) was set to 0.025, 0.05, 0.075, 0.1, and 0.125, respectively. From the red circles, we can see that the bit rate saving of the proposed model increases from 0.878 to 0.894 when n increases from 1 to 9, ΔPSNR increases from 0.175 to 0.451 (see Fig. 15(a)), and ΔQF increases 1.78 from to 4.68 (see Fig. 15(b)). It can be concluded that the proposed model has a high bit rate saving which is close to that of the ground truth. It also has a stable performance when the QF varies. From the dotted and solid line, we can see that the performance of the FEP_PJND and EPC_PJND model is very similar. The bitrate saving, ΔPSNR, and ΔQF vary greatly when λ increases from 0.025 to 0.125. The comparison models need to pay large ΔPSNR (or ΔQF) to achieve a high bit rate saving. If we fix the ΔPSNR (or ΔQF), the proposed model has a larger bit rate saving. If we fix the bit rate saving, the proposed model has a lower ΔPSNR and ΔQF. Therefore, we can conclude that the proposed model has a better performance than the two pixel domain models. The experimental results of the second PW-JND are plotted on Fig. 16, where the x-axis, y-axis, lines, and legends are the same with that in Fig. 15. We set λ the same as the first PW-JND case and $n \in [1, 2, \ldots, 5]$. The bit rate saving of the proposed model is close to that of the ground truth, which is smaller than that of the first PW-JND case. The reason may be that the JPEG coder compresses image more at a high QF level. We can also see from Fig. 16 the proposed model has the best performance.

As mentioned in Section I, we can predict PW-JND in the bit rate domain (R) or other discrete/continuous domains, *e.g.*, QF, and PSNR. Therefore, we analyse the correlation between the prediction and ground truth PW-JNDs in QF, PSNR, and bit rate domain. The scatter plots map of predicted PW-JNDs and ground truth PW-JNDs are shown in Fig. 17, where x-axis represents ground truth PW-JNDs, and y-axis represents the predicted PW-JNDs. We can see from Fig. 17 that: 1) The correlation of QF ($R^2 = 0.12$) and bit rate prediction ($R^2 = 0.615$) is low. Especially, the correlation of QF prediction has the lowest correlation. The ground truth PW-JND in MCL-JCI data set is a statistical value, around which there is an ambiguous region in perceptual quality. The quality difference among the distorted images in such region is very hard to distinguish by humans. The width of the region is determined by image content and subjects, which is big for most images in QF. There is a great possibility

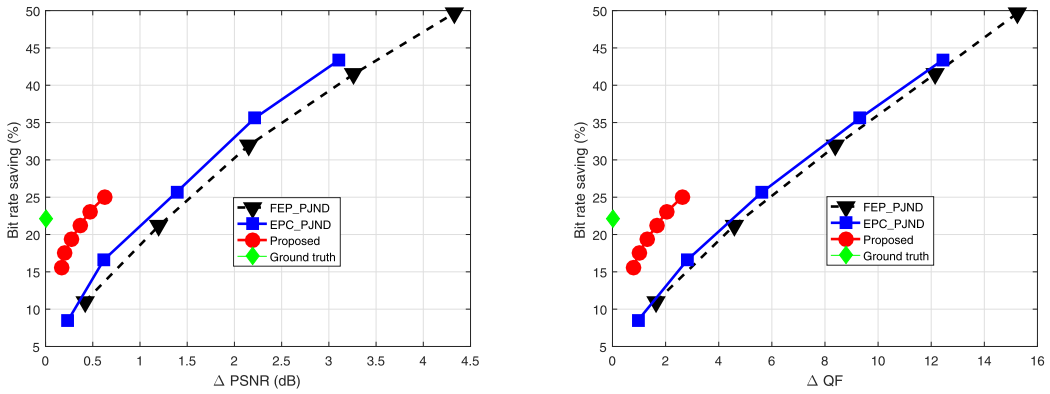


Fig. 16. Bit rate saving versus ΔPSNR and ΔQF of the second prediction PW-JND. (a) Δ PSNR. (b) ΔQF.

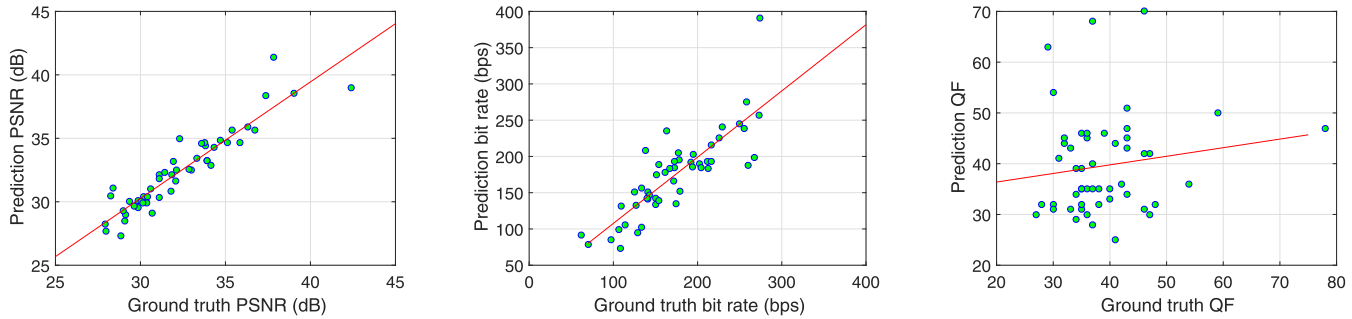


Fig. 17. Prediction of the first PW-JND versus ground truth PW-JND in different metrics. (a) PSNR: $R^2 = 0.908$. (b) Bit rate: $R^2 = 0.615$. (c) QF: $R^2 = 0.12$.

that the predicted QF falls within this region. The wider the ambiguous region, the greater possibility the predicted QF would deviate from the ground truth, thus leading to a lower correlation of QF prediction. 2) The PSNR has the highest correlation ($R^2 = 0.908$), which denotes that we can predict the PW-JNDs in PSNR domain. The predicted PSNR can be used in different image/video processing algorithms: 1) It can be used to compress image/video with the lowest bit rate without perceptual quality degradation. 2) It is also helpful for streaming system to select the images/videos with the smallest size but best quality, which can save the bandwidth without damaging consumers' experience. 3) It can be used to guide watermarks embedding, which ensure the impairment of the embedded digital watermarks cannot perceived by the humans.

VII. Conclusion

In this paper, we propose a deep learning method based Picture Wise Just Noticeable Difference (PW-JND) prediction model. Firstly, the task of predicting PW-JND is formulated as a multi-class classification problem, which is transformed to a binary classification. Secondly, we construct a deep learning based binary classifier named perceptually lossy/lossless predictor to predict whether a distorted image is perceptually lossy to its reference or not. Finally, we propose a sliding window based PW-JND search strategy to predict the PW-JND. Experimental results on comparison with the conventional just noticeable difference models demonstrate the effectiveness of the proposed model.

References

[1] H. Zhang, Y. Zhang, H. Wang, Y.-S. Ho, and S. Feng, "WLDISR: Weighted local sparse representation-based depth image super-resolution for 3D video system," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 561–576, Feb. 2019.

[2] L. Toni and P. Frossard, "Optimal representations for adaptive streaming in interactive multiview video systems," *IEEE Trans. Multimedia*, vol. 19, no. 12, pp. 2775–2787, Dec. 2017.

[3] Y. Zhang, X. Yang, X. Liu, Y. Zhang, G. Jiang, and S. Kwong, "High-efficiency 3D depth coding based on perceptual quality of synthesized video," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5877–5891, Dec. 2016.

[4] L. Xu *et al.*, "Free-energy principle inspired video quality metric and its use in video coding," *IEEE Trans. Multimedia*, vol. 18, no. 4, pp. 590–602, Apr. 2016.

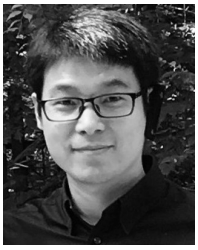
[5] S.-H. Bae, J. Kim, and M. Kim, "HEVC-based perceptually adaptive video coding using a DCT-based local distortion detection probability model," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3343–3357, Jul. 2016.

[6] X. Zhang, S. Wang, K. Gu, W. Lin, S. Ma, and W. Gao, "Just-noticeable difference-based perceptual optimization for JPEG compression," *IEEE Signal Process. Lett.*, vol. 24, no. 1, pp. 96–100, Jan. 2017.

[7] Z. Luo, L. Song, S. Zheng, and N. Ling, "H.264/advanced video control perceptual optimization coding based on JND-directed coefficient suppression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 6, pp. 935–948, Jun. 2013.

[8] H. Su and C. Jung, "Perceptual enhancement of low light images based on two-step noise suppression," *IEEE Access*, vol. 6, pp. 7005–7018, 2018.
[9] T. Zhu and L. Karam, "A no-reference objective image quality metric based on perceptually weighted local noise," *EURASIP J. Image Video Process.*, vol. 5, pp. 1–8, Jan. 2014.
[10] J. Wu, F. Qin, and M. Shi, "Self-similarity based structural regularity for just noticeable difference estimation," *J. Vis. Commun. Image Represent.*, vol. 23, no. 6, pp. 845–852, Aug. 2012.
[11] J. Wu, G. Shi, W. Lin, A. Liu, and F. Qi, "Just noticeable difference estimation for images with free-energy principle," *IEEE Trans. Multimedia*, vol. 15, no. 7, pp. 1705–1710, Nov. 2013.
[12] J. Wu, W. Lin, G. Shi, X. Wang, and F. Li, "Pattern masking estimation in image with structural uncertainty," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4892–4904, Dec. 2013.
[13] S. Wang, L. Ma, Y. Fang, W. Lin, S. Ma, and W. Gao, "Just noticeable difference estimation for screen content images," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3838–3851, May 2016.
[14] J. Wu, L. Li, W. Dong, G. Shi, W. Lin, and C.-C. J. Kuo, "Enhanced just noticeable difference model for images with pattern complexity," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2682–2693, Jun. 2017.
[15] H. Hadizadeh, A. Rajati, and I. V. Bajić, "Saliency-guided just noticeable distortion estimation using the normalized laplacian pyramid," *IEEE Signal Process. Lett.*, vol. 24, no. 8, pp. 1218–1222, Aug. 2017.
[16] Z. Wei and K. N. Ngan, "Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 3, pp. 337–346, Mar. 2009.
[17] S.-H. Bae and M. Kim, "A novel DCT-based JND model for luminance adaptation effect in DCT frequency," *IEEE Signal Process. Lett.*, vol. 20, no. 9, pp. 893–896, Sep. 2013.
[18] S.-H. Bae and M. Kim, "A novel generalized DCT-based JND profile based on an elaborate CM-JND model for variable block-sized transforms in monochrome images," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3227–3240, Aug. 2014.
[19] S. Bae and M. Kim, "A DCT-based total JND profile for spatiotemporal and foveated masking effects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 6, pp. 1196–1207, Jun. 2017.
[20] S. Ki, S.-H. Bae, M. Kim, and H. Ko, "Learning-based just-noticeable-quantization-distortion modeling for perceptual video coding," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3178–3193, Jul. 2018.
[21] X. Liu, Y. Zhang, S. Hu, S. Kwong, C.-C. J. Kuo, and Q. Peng, "Subjective and objective video quality assessment of 3D synthesized views with texture/depth compression distortion," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4847–4861, Dec. 2015.
[22] S. Hu, H. Wang, and C.-C. J. Kuo, "A GMM-based stair quality model for human perceived JPEG images," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2016, pp. 1070–1074.
[23] L. Jin *et al.*, "Statistical study on perceived JPEG image quality via MCL-JCI dataset construction and analysis," *Electron. Imag.*, vol. 2016, no. 13, pp. 1–9, Feb. 2016.
[24] H. Wang *et al.*, "VideoSet: A large-scale compressed video quality dataset based on JND measurement," *J. Vis. Commun. Image Represent.*, vol. 46, pp. 292–302, Jul. 2017.
[25] C. Fan, Y. Zhang, H. Zhang, R. Hamzaouic, and Q. Jiang, "Picture-level just noticeable difference for symmetrically and asymmetrically compressed stereoscopic images: Subjective quality assessment study and datasets," *J. Vis. Commun. Image Represent.*, vol. 62, pp. 140–151, Jul. 2019.
[26] Q. Huang, H. Wang, S. C. Lim, H. Y. Kim, S. Y. Jeong, and C.-C. J. Kuo, "Measure and prediction of HEVC perceptually lossy/lossless boundary QP values," in *Proc. IEEE Data Compress. Conf. (DCC)*, Apr. 2017, pp. 42–51.
[27] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, vol. 2, Nov. 2003, pp. 1398–1402.
[28] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.
[29] W. Xue, L. Zhang, X. Mou, and A. C. Bovik, "Gradient magnitude similarity deviation: A highly efficient perceptual image quality index," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 684–695, Feb. 2014.
[30] L. Zhang, Y. Shen, and H. Li, "VSI: A visual saliency-induced index for perceptual image quality assessment," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4270–4281, Aug. 2014.
[31] L. Xu *et al.*, "Multi-task rank learning for image quality assessment," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 9, pp. 1833–1843, Sep. 2017.
[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[33] L. Zhu, Y. Zhang, S. Wang, H. Yuan, S. Kwong, and H.-H. S. Ip, "Convolutional neural network-based synthesized view quality enhancement for 3D video coding," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5365–5377, Nov. 2018.
[34] S. Bosse, D. Maniry, K. R. Müller, T. Wiegand, and W. Samek, "Deep neural networks for no-reference and full-reference image quality assessment," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 206–219, Jan. 2018.
[35] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.
[36] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980
[37] S. Hu, L. Jin, H. Wang, Y. Zhang, S. Kwong, and C.-C. J. Kuo, "Compressed image quality metric based on perceptually weighted distortion," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5594–5608, Dec. 2015.

Huanhua Liu received the B.S. degree in computer science and technology from Changsha University, China, in 2009, and the M.S. degree in computer science and technology from the University of South China, China, in 2013. He is currently pursuing the Ph.D. degree with Central South University. Since 2017, he has been with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. His current research interests include learning-based image/video quality assessment and video coding.



Yun Zhang (M'12–SM'16) received the B.S. and M.S. degrees in electrical engineering from Ningbo University, Ningbo, China, in 2004 and 2007, respectively, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2010. From 2009 to 2014, he was a Postdoctoral Researcher with the Department of Computer Science, City University of Hong Kong, Hong Kong. From 2010 to 2017, he was an Assistant Professor and an Associate Professor with the Shenzhen Institutes of Advanced Technology (SIAT), CAS, Shenzhen, China, where he is currently a Professor. His research interests include video compression, 3D video processing, and visual perception.

Huan Zhang received the B.S. degree from the Civil Aviation University of China, Tianjin, China, in 2010, and the M.S. degree from Tsinghua University, Beijing, China, in 2013. She is currently pursuing the Ph.D. degree with the University of Chinese Academy of Sciences, China. Her research interests include image restoration and image/video quality assessment.

Chunling Fan received the M.S. degree from Nanjing Normal University, Nanjing, in 2011. She is currently pursuing the Ph.D. degree with the Shenzhen Institutes of Advanced Technology (SIAT), University of Chinese Academy of Sciences, Shenzhen, China. Her research interests include image processing and image quality assessment.

Sam Kwong (F'13) received the B.S. degree in electrical engineering from the State University of New York at Buffalo in 1983, the M.S. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of Hagen, Germany, in 1996. From 1985 to 1987, he was a Diagnostic Engineer with Control Data Canada. He joined Bell Northern Research Canada as a member of Scientific Staff. In 1990, he became a Lecturer with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, where he is currently a Professor with the Department of Computer Science. His research interests include video and image coding and evolutionary algorithms.

C.-C. Jay Kuo (F'99) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1980, and the M.S. and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge MA, USA, in 1985 and 1987, respectively. He is currently the Director of the Multimedia Communications Laboratory and a Professor of electrical engineering, computer science and mathematics with the Ming-Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA. He has coauthored about 200 journal papers, 850 conference papers, and ten books. His research interests include digital image/video analysis and modeling, multimedia data compression, communication and networking, and biological signal/image processing. He is a fellow of The American Association for the Advancement of Science and The International Society for Optical Engineers.

Xiaoping Fan received the B.S. degree in electrical engineering from the Jiangxi University of Technology (Nanchang University), Nanchang, China, in 1981, the M.S. degree in traffic information engineering and control from Changsha Railway University (Central South University), Changsha, Hunan, in 1984, and the Ph.D. degree in control science and engineering jointly from the South China University of Technology, Guangzhou, and The Hong Kong Polytechnic University, Hong Kong, in 1998. He was a Professor with the School of Information Science and Engineering, Central South University. Since 2010, he has been a Professor with the Laboratory of Networked Systems, Hunan University of Finance and Economics. He is the author of two books, over 300 journal and conference papers, and 15 inventions. His research interests include robot control, wireless sensor networks, data mining, and intelligent transportation systems.