Towards Coding for Human and Machine Vision: Scalable Face Image Coding

Shuai Yang, *Member, IEEE*, Yueyu Hu, *Student Member, IEEE*, Wenhan Yang, *Member, IEEE*, Ling-Yu Duan, *Member, IEEE*, and Jiaying Liu, *Senior Member, IEEE*

***Abstract*—The past decades have witnessed the rapid development of image and video coding techniques in the era of big data. However, the signal fidelity-driven coding pipeline design limits the capability of the existing image/video coding frameworks to fulfill the needs of both machine and human vision. In this paper, we come up with a novel face image coding framework by leveraging both the compressive and the generative models, to support machine vision and human perception tasks jointly. Given an input image, the feature analysis is first applied, and then the generative model is employed to reconstruct image with compact structure and color features, where sparse edges are extracted to connect both kinds of vision and a key reference pixel selection method is proposed to determine the priorities of the reference color pixels for scalable coding. The compact edge map serves as the basic layer for machine vision tasks, and the reference pixels act as an enhanced layer to guarantee signal fidelity for human vision. By introducing advanced generative models, we train a decoding network to reconstruct images from compact structure and color representations, which is flexible to accept inputs in a scalable way and to control the imagery effect of the outputs between signal fidelity and visual realism. Experimental results and comprehensive performance analysis over the face image dataset demonstrate the superiority of our framework in both human vision tasks and machine vision tasks, which provide useful evidence on the emerging standardization efforts on MPEG VCM (Video Coding for Machine).**

***Index Terms*—Generative compression, image coding, scalable coding, video coding for machine.**

I. Introduction

GREAT efforts have been made in the evolution of image processing technologies, to handle the vast amount of visual information in real-world applications. All these images need to be encoded before they are transmitted and displayed by clients, or processed and analyzed by servers. While in the past, most of the images are to be displayed or stored for human examination, in the era of intelligent visual computing, an increasing amount of visual data could at the same time serves human perception and powers machine vision intelligent systems. For now, existing methods for visual compression are mainly designed for either human vision or machine vision, leading to different visual processing paradigms.

The conventional processing paradigms for images include a compression scheme to first encode the images into bit-stream and reconstruct them back to pixels for human vision analysis. For this purpose, transform-based hybrid coding standards [1]–[3] have been already widely deployed. In recent works, image compression methods that utilize machine learning techniques [4], [5] further improve the rate-distortion performance, achieving the minimal pixel-level distortion under the bit-rate constraints. However, these methods are shown to be incapable of efficiently supporting machine vision tasks at low-bit-rate conditions [6]. The full-resolution image is redundant in information entropy, which inevitably results in high bit-rates with the image coding methods. Comparatively, machine vision analysis usually only relies on very compact visual representations. Besides, existing coding methods focus on maintaining pixel-wise signal fidelity, which does not guarantee the preservation of high-level semantics. Thus, it is not desirable to utilize the conventional image coding method to support high-efficient machine vision analysis. The new generation of coding methods are expected.

Later on, several works have made efforts in addressing the problem of video analytics on massive data by directly extracting and compressing features used for machine vision tasks into a compact form, rather than compressing the whole high-quality videos. Several typical features are developed, *e.g.*, Scale-Invariant Feature Transform (SIFT) [7] and skeleton for human action recognition [8]. Based on that, techniques to encode the compact descriptors for visual search [9] and video analysis [10] have been further developed and standardized (ISO/IEC15938-13 and ISO/IEC15938-15). In this way, the process of feature extraction, compression and transmission becomes light-weighted and less amount of bit-streams are to be handled. Though these features are compact and highly effective for machine vision tasks, they completely squeezed out the appearance information and are designed only for specific tasks. As no algorithm can guarantee perfect accuracy, some visual contents may need further human examination in some large-scale machine vision systems, *e.g.*, Smart Cities, Internet of

Manuscript received September 17, 2020; revised December 20, 2020; accepted December 27, 2020. Date of publication March 24, 2021; date of current version September 24, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0102702, in part by the Fundamental Research Funds for the Central Universities and in part by the National Natural Science Foundation of China under Contract 61772043. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. M. Mrak. *(Corresponding author: Jiaying Liu.)*

The authors are with Peking University, Beijing 100080, China (e-mail: williamyang@pku.edu.cn; huyy@pku.edu.cn; yangwenhan@pku.edu.cn; lingyu@pku.edu.cn; liujiaying@pku.edu.cn).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TMM.2021.3068580.

Digital Object Identifier 10.1109/TMM.2021.3068580

Things (IoT). Few of the existing compression schemes provide the scalability to collaboratively support high-efficiency machine analytics and human perception. Besides, for other real-time joint machine-human visual applications, *e.g.*, telepresense and augmented reality (VR), bit-streams for human vision reconstruction and machine analysis are transmitted separately, causing duplication in bit-rate.

Nowadays, the need of a novel compression scheme to collaboratively support machine vision and human vision has been emerging, while in the big data context, it is still an open problem to support a scalable coding paradigm to satisfy both kinds of vision. Although the processing paradigms for human vision and machine vision have apparent differences, they are at the same time closely connected. Both of them follow the processing paradigm to first extract features from the visual content and then generate outputs by further analysis on the features. We are therefore able to explore the possibilities to support machine and human vision tasks jointly in a flexible way, which is expected in the new coding paradigm of video coding for machine (VCM) [11]. To this end, we aim to extract the expressive compact feature that can jointly facilitate machine vision analysis and power human vision examination and design the compression framework to encode and decode the compact feature for various tasks, including visual reconstruction. We provide an evidence for the possibilities of collaborative coding in the VCM paradigm.

In this work, we start the exploration of our new coding paradigm with human face images, as human faces are usually the salient area in images and lead to a very important research area of computer vision. It has also been a focus in image/video coding [12], [13], including the general video object plane technique [14], [15] in MPEG-4. Our work contributes to facilitating the emerging video conferencing applications by reducing the bandwidth to transmit pictures of human faces in high quality, which has been an important research topic in image and video communication [16]. In addition, there are rich classification and detection models to facilitate the evaluation on machine vision tasks.

In this paper, we focus on the face images and explore to utilize the edge representation with the corresponding color information as the compact feature to build a scalable framework for human-machine collaborative compression. By leveraging both compressive and generative models, a scalable face image coding framework is constructed to support machine and human vision tasks jointly. In this framework, the source image is represented via a compressive model as edge maps and sparse key reference pixels. The edges are parameterized into vectors as the base layer of the coding bits to obtain a compact feature representation, which only takes a small portion of coding bits. Furthermore, the information in our edge maps is shown to be efficient for machine vision tasks, *e.g.* facial landmark detection and gender classification. To better reconstruct the high-quality frame, reference pixels, sampled in accordance with the edges, can be transmitted as a second layer to the decoder. With the reference pixel values, the decoder is able to faithfully reconstruct the image. We adopt a generative model to reconstruct high-quality images from the sparse edge representations. Experiments on both machine and human vision show significant improvements compared with existing methods, which provide useful evidence on the emerging standardization efforts on MPEG VCM.

Compared with our previous work [17], we further explore more scalable color representations on the encoding phase and imagery effect control on the decoding phase. First, our improved single model simultaneously works for both human vision and machine vision, while our previous model has to be trained separately for each task. Second, we propose key reference pixel selection for more scalable color representation, which not only saves bit-rate for human vision, but also increases the signal fidelity at the same time. Furthermore, even under the same bit-rate, our improved decoder allows users to control the imagery effect of the reconstructed image, which provides greater flexibility for different vision tasks. In addition, comprehensive experiments are conducted to analyze the image coding performance of the proposed model, including additional comparison results for qualitative and quantitative evaluations, analysis on the color selection and parameter settings of our proposed techniques, and discussions on the dataset and quality evaluation issues of VCM. In summary, the contributions of this work are threefold:

- We propose a face image coding framework that leverages the compressive model to extract compact representations of an image and faithfully reconstruct the original image from the bitstreams with the generative model.
- We design the vision-driven compact representations for image compression, where the critical image structure and color information is sparsely encoded. Color information is further scalably encoded using the proposed key reference pixel selection method.
- A deep generative network is proposed to effectively recover images from our compact representations in a both color scalable and imagery effect controllable manner.
- A good balance between human and machine vision is stricken, where we achieve over 99% and 80% human vision preferences in terms of realism and fidelity, respectively, and achieve an error drop of 33.43% in the facial landmark detection and an improvement of 15.5% in gender classification accuracy.

The rest of this paper is organized as follows. In Section II, we review related works in image coding and image generation. Section III defines the image compression problem for both machine and human vision, and gives an overview of the framework of our method. In Section IV and V, the details of the proposed vision-driven scalable image coding model and multi-task generative image decoding model are presented, respectively. We validate our method by conducting extensive experiments and thorough analyses in Section VI and discuss the emerging issues of VCM in Section VII. Finally, we conclude our work in Section VIII.

II. Related Work

A. Feature Based Image Coding

Besides the mainstream transform based codecs [2], [3], there have been other approaches to explore encoding representative image features for reconstruction. In [18], a generative compression framework is proposed to encode an image into low-bit-rate latent code and exploit recurrent generative networks for reconstruction. With compressive variational auto-encoders (VAE) [19], generative networks are also utilized in [20] to reconstruct images from edges and latent features produced by neural networks. Though these frameworks encode compact feature representations of images, they are not shown to both satisfy the need of human and machine vision. In [21], a deep-based encoder is designed to produce latent code that simultaneously serves for machine vision tasks and image reconstruction. In [22], a scalable image compression scheme is proposed, where a base layer serves feature representation for machine vision and an enhancement layer serves texture representation for human vision. In [23], a bit allocation and rate control strategy is developed for object detection, where an importance map is created to guide bit allocation to the important areas for object detection. In [24], a semantically structured coding framework is developed to generate semantically structured bit-stream (SSB). Each part of the bit-stream represents a certain object and can be directly used for a series of analytic tasks. There are a series of works [25]–[28] paying attention to machine-human collaborative intelligence. However, the encoded feature representation is non-scalable as the full bit-stream is needed to support the machine vision tasks, neglecting the sparsity of the features for machine vision. In this work, we explore to encode a base layer of features to facilitate machine vision and an additional layer to improve signal fidelity.

B. Image Generation

Image generation investigates generating new images to approach the target image distribution. Recent image generation methods focus on the powerful generative adversarial networks (GAN) [29] to learn data distribution using two adversarial networks. By incorporating additional information such as the text, labels, segmentation maps and edges as inputs, users are able to control the output with these conditions. For images as conditional inputs, the problem becomes a specialized image-to-image translation problem. Isola *et al.* [30] put forward image-to-image translation and designed a pix2pix network based on UNet and PatchGAN [30] to accomplish this task. Zhu *et al.* [31] improved pix2pix by incorporating variational autoencoder to enhance the diversity of the generated images.

The advanced GAN has shown impressive capability of data distribution learning to recover abundant information that well matches human visions from limited conditions. Such an advantage is also verified by the closely related image inpainting task, where plausible image content is generated from very sparse contextual information [32]. Image inpanting targets at reconstructing the missing regions of an image. Pathak *et al.* [33] proposed Context Encoder to leverage the training data for semantic inpainting. In [34], [35], image edge and color are predicted sequentially, which shows promising results. It demonstrates the potential for vision-driven image coding, which forms our research focus in this paper.

Similar to the aforementioned models for image-to-image translation and inpainting, the proposed model leverages GAN as decoder but pay more attentions to the image fidelity in addition to generating plausible image content. Moreover, our model is able to further accept scalable compact representation as input and generates images under different imagery effect to better adapt to various bit-rate constraints and vision tasks.

III. Problem Formulation and Framework

Our work focuses on developing a scalable face image compression framework that meets the need of machine vision with the very compact feature representation, while being capable of reconstructing the full images with the additional bit-rates in a scalable way. Formally, our target is to maximize the multi-task performance from the perspective of both machine and human visions, with the resource usage constraint based on a series of scalable features. Following the paradigm of VCM [11], our framework can be formulated as follows,

$$\begin{aligned} &\arg\min_{\theta}\ D^M + D^H, \\ &\text{s.t. } S(R_M) + S(R_{M\rightarrow H}) \leq S_T, \end{aligned} \tag{1}$$

where M and H denote machine and human visions, D^M and D^H are the degradation in performance of machine vision tasks and human vision tasks, respectively, when the total bit-rate of the visual representations is constrained by S_T. θ is the parameters of the coding architecture. $S(\cdot)$ calculates the bit-rate of encoded features. The first term calculates the bit-rate cost to encode the abstract feature R_M that serves machine vision. The second term corresponds to the cost to encode the features $R_{M\rightarrow H}$ for human vision, conditioned on the already encoded R_M in the first part.

To solve the problem in Eq. (1), we develop the compression framework in Fig. 1, with an encoder and a decoder, to encode the images into bit-streams and reconstruct them from the bit-streams to serve tasks of machine and human vision.

In the encoder side, we propose the design to encode I into two layers of compact representations, corresponding to R_M and $R_{M\rightarrow H}$ in Eq. (1), respectively, where the bit-rate usage of the representations satisfies the constraint. Specifically, we first extract sparse edges to depict the key structure information of the input image (Section IV-A) and obtain the vision-driven compact structure code based on the edge sparsity (Section IV-B). Then we extract sparse reference pixels through compressive analysis over the edges and the original images (Section IV-C) and obtain scalable color code by selecting the pixels with key color information based on the feedback from the decoder side (Section IV-D). The structure code serves as a base layer R_M to facilitate machine vision and the color code serves as a scalable enhanced layer $R_{M\rightarrow H}$ to improve signal fidelity to fulfill the real need in both machine vision and human vision.

In the decoder side, we train a deep neural network as the decoder to reconstruct the original image from our compact representation (Section V-B), where we tune the decoder to minimize

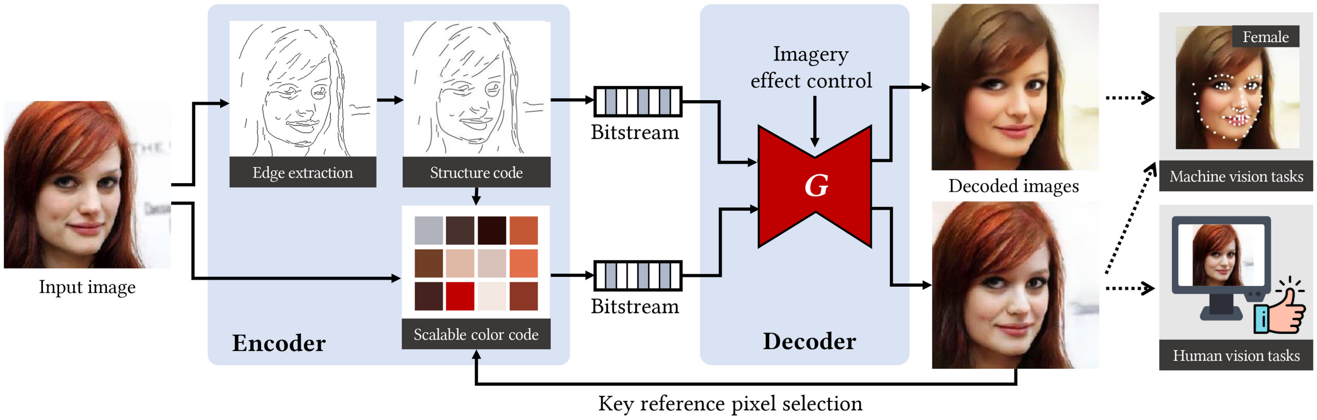


Fig. 1. Overview of the proposed human vision and machine vision co-driven face image coding framework.

the degradation in performance of both machine vision tasks and human vision tasks. As it is hardly tractable to directly conduct the optimization in Eq. (1), we instead train the generative model in the decoder to minimize the discrepancy of the original images and ones generated from the compact representations, with the loss function to reduce the domain gap as well as the distortion between the reconstructed images and the ground truths. Compared with existing image codecs which directly minimize the pixel-level error, the proposed framework can better reduce D^M and D^H under the constraints on the bit-rates, for the following reasons. First, our encoder is designed to extract the key features that well support machine vision and human vision tasks to generate the compact representations. So the critical information has been preserved. Second, our decoder is trained to maximize the domain consistency and the visual fidelity between the reconstructed images and the original ones, thus it better facilitates machine vision analysis and human perception. With the proposed framework, we achieve a better solution to the problem defined in Eq. (1) than existing methods. In addition, our decoder accepts scalable color code and enables users to adjust the imagery effects between *fidelity* and *realism* of the decoded images, which provides much flexibility to adapt to different vision tasks (Section V-C).

IV. Vision-Driven Scalable Face Image Coding

A. Sparse Edge Extraction

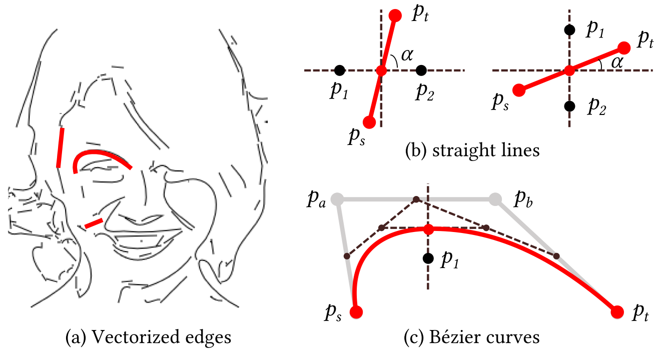
To build our human vision and machine vision co-driven face image coding scheme, We would like to find the good representation that can be scalable and understandable for machines and humans simultaneously. In this work, we choose the edges as the basic representation for images. Edges are one of the most highly abstract and sparse image representations, and it is also light-weight. It usually contains the vital information needed by most of the machine vision tasks. Meanwhile, edges efficiently convey the key structural information of the image, which is also consistent with the human vision. For example, humans are able to identify the objects from several lines and even infer fine details such as the colors and textures. To this end, we are inspired to build our compact representation using sparse edges. We will show later that images can be plausibly reconstructed purely from its edges based on the robust data distribution learned by GAN.

Specifically, we first extract sparse edges from an input image I using the structured-forest-based edge detection method [36]. These edges are further binarized with trivial short edges discarded based on the post processing algorithm suggested by pix2pix [30].

B. Compact Structure Representation Extraction

Edge map is a special sparse kind of images, which contains only binary fixed-width edges. It is not straight-forward to code such maps into compact bit-streams. Existing works in feature-based image compression exploit recurrent generative neural networks [18] or resort to HEVC Screen Content Coding [20], [37]. However, these methods mainly rely on pixel-level representations or image partitions for natural images, which do not make full use of the sparsity of the edges, leading to low efficiency when coding such binary maps.

In our approach, to fully exploit the sparsity of binary fixed-width edges, we propose to trace the edges into vector graphics for more effective edge map encoding. Specifically, the line tracing tool [38] is adopted to translate the binary edge maps into vectorized representations. We follow the Scalable Vector Graphics (SVG) syntax to approximate edges by straight lines and Bézier curves. More specifically, three operation markers of **M**ove, **L**ine and **C**urve are used. As illustrated in Fig. 2, let p_s be a starting point in the edge map. Then, operation $\mathbf{M}(p_t)$ refers to moving from p_s to a target point p_t. Operation $\mathbf{L}(p_t)$ indicates drawing a straight line from p_s to the target point p_t and moving to p_t. Operation $\mathbf{C}(p_a, p_b, p_t)$ denotes drawing a cubic Bézier curve from the current point to the target point p_t with the intermediate points p_a and p_b, and moving to p_t. Since edges are mostly smooth in natural images, we can approximate them well using the above-mentioned straight lines and curves with few parameters. In the end, these parameters are quantized and losslessly compressed into compact bit-streams with Partial Matching (PPM) [39] compression scheme, which further eliminates redundancy.



(a) Vectorized edges

(b) straight lines

(c) Bézier curves

Fig. 2. Illustration of our vectorized structure representation and point samplings for color representation. (a) A vectorized edge map. (b) For straight lines, two points are selected as the reference, according to the slope α. (c) For Bézier curves, one inner point is selected.

C. Sparse Reference Pixel Extraction

Based on the edges, we further introduce color representations to tackle the problem of color ambiguity. Color is another critical important information for human perception. It constitutes the main characteristics of the spaces circumscribed by edge lines, which helps maintain visual fidelity. Moreover, as a basic low-level feature, it can even impact some high-level concepts such as emotions. On the other hand, color information is optional for some machine vision tasks like facial landmark detection, which suggests potential scalability. Thus, in accordance with the aforementioned structure representation, we propose to extract pixel-level sparse color representation to better support the scalable coding scheme.

To be specific, we sparsely sample pixels near the straight lines and curves. For a straight line, two reference points p_1 and p_2 are sampled near the midpoint based on the slope of the line. As illustrated in Fig. 2(b), if the line is more close to vertical, *i.e.*, $\alpha \geq 45°$, we sample p_1 and p_2 horizontally, while if $\alpha < 45°$, we sample vertically. As illustrated in Fig. 2(c), for a Bézier curve with parameters $\{p_s, p_a, p_b, p_t\}$, we first extract the contact point of the curve and the tangent line in parallel with the vector $\overrightarrow{p_s p_t}$. As with the straight line, the slope of the tangent line is used to determine the sampling direction of the point. In addition, to control the bit-rate, we only sample the point p_1 at the inner side of the curve, which is expected to maintain the most representative color information in the spaces circumscribed by curve. The pixel, represented in RGB value, is signaled to the decoder in order as a second layer to provide more fidelity in color. The proposed sparse reference pixel extraction has two advantages. First, the decoder could place the received reference color points following the same rules that the encoder extracts those points, based on the edge maps. Thus, no additional bits are needed to record the positions of the selected pixels. Second, as we will show later in the experiment, such pixels selected adaptively to the edge maps are more representative and informative.

D. Scalable Color Representation Extraction

For some high-level color-robust machine vision tasks such as facial landmark detection and gender classification that do not strictly rely on accurate color information, or under some circumstance of strict bit-rate constraint that the full enhanced layer can not be satisfied, a coding framework that accepts scalable enhanced layer is expected to further save bit-rate. Furthermore, the reference pixels extracted in Section IV-C inevitably contain redundant and noisy color information. Removing those pixels kills two birds with one stone: saving bit-rate and improving fidelity simultaneously. To this end, we are motivated to investigate scalable color representation extraction, *i.e.*, estimating the priorities of the reference pixels and removing a certain amount of less important or noisy reference pixels to satisfy the bit-rate requirement.

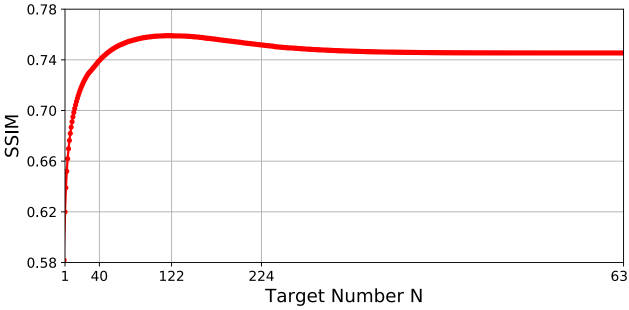


Fig. 3. Illustration of the target reference pixel number N and the averaged SSIM of the corresponding reconstructed images.

Our key idea is to determine the priority of pixel removal based on the feedback from the decoder G. The detail of G will be given in Section V, which decodes the structure code E and color code C to the reconstructed image I_G. Let P be the full reference pixel set with $|P|$ pixels obtained in Section IV-C. $N < |P|$ is the target amount of reference pixels we want to maintain. First, we enumerate each $p \in P$, remove p from P, and reconstruct image based on the subset P/p. Then we find p' corresponding to the reconstructed result most similar (SSIM is utilized to compute the structure similarity) to the original image as the least important pixel to be removed and update P as P/p'. The above operations are iterated until a total of $(|P| - N)$ pixels are removed. Although precise, the algorithm is inefficient: to estimate the priority of all pixels, we have to decode $(|P| + 1)|P|/2$ times.

To speed up, we propose a heuristic algorithm. Intuitively, removing a pixel has little effect on the removal of pixels far away. Therefore, after sorting pixels based on SSIM, we could roughly remove multiple pixels that are far from each other (distance greater than parameter r) rather than removing only one optimal pixel. Number of pixels to be removed is determined by the current total pixel number: $1/k$ of the total pixels are to be removed simultaneously. Meanwhile, we observe that the singal fidelity becomes more sensitive to the changes of pixels when there are few pixels remaining. Therefore, we set a threshold n, under which we still remove one pixel per iteration for accuracy. The proposed key reference pixel selection algorithm is summarized in Algorithm 1.

Fig. 3 illustrates the decoded results in terms of SSIM given different target number N in our testing set. For 3000 testing

Algorithm 1: Key Reference Pixel Selection

```
Input: Image $I$, reference pixel set $P$, structure code $E$
  generator $G$, target number $N$, removing ratio $k = 8$
  sparse radius $r = 10$, min removing number $n_0 = 10$
  threshold number $n = 40$ to keep precise
Output: selected reference pixel set $P^*$
initialize $P^* = P$
while $|P^*| > N$ do
  $\triangle$ enumerate one reference pixel to remove
  for all $p \in P^*$ do
    generate $I_G$ via $G$ based on $I$, $E$ and $P^*/p$
    compute image quality $q_p = \text{SSIM}(I_G, I)$
  $\triangle$ sort reference pixels based on $q_p$
  $\tilde{P} = \text{sort}(P^*)$
  if $|P^*| < n$ then
  $\triangle$ precisely remove a pixel when few pixels remain
    update $P^* \leftarrow P^*/\tilde{P}(1)$
  else
    $\triangle$ roughly remove multiple sparse pixels
    compute $\Delta n = \min(\max(n_0, |P^*|/k), N - |P^*|)$
    initialize removed set $P_r = \varnothing$
    for $i = 1 \to |\tilde{P}|$ do
      if $|\tilde{P}(i) - p| > r$ for all $p \in P_r$ then
        update $P^* \leftarrow P^*/\tilde{P}(i), P_r \leftarrow P_r \cup \{\tilde{P}(i)\}$
          if $|P_r|$ equals to $\Delta n$ then
            break
```

images, the average number of reference pixels are 224, which is however not the optimal. By removing less important or noisy reference pixels until only 122 pixels left, SSIM reaches its peak. Finally, when $N <= 40$, SSIM drops rapidly with less color cues. Therefore, we set $n = 40$ in our experiment.

To compose the bit-stream that supports scalable parameterization, we first signal the binary flags in the file header to indicate which reference pixels are used. After that, the compressed vectorized edges are transmitted, followed by the corresponding RGB values represented in raw binaries. Compared with our previous work, we require additional 224-bit space for flags, but save around $(224 - 122) \times 24 = 2448$ bits for RGB values, adding up to about 0.034 bpp (bit per pixel) savings.

V. Multi-Task Generative Face Image Decoding

A. Image-to-Image Translation Via GAN

In this section, we briefly review the basic conception of GAN for image-to-image translation, which constitutes our decoder. Image-to-image translation [30] is first systematically raised by Isola *et al.*, aiming to transform input images to their closely related output images, such as grey photos to color photos. In [30], their mappings are modelled by a generator $G : X \to Y$, where X are conditional images in the source domain and Y are images in the target domain. For a pair $\{x, y\} \in \{X, Y\}$, we want the generated image $G(x)$ to approach to target y, which is usually formulated as a reconstruction loss $\mathcal{L}_r$. In addition, GAN introduces a discriminator D to discriminate the generated image $G(x)$ from the real image y and whether it matches the condition x. Meanwhile, G tries to confuse D. The two networks compete with each other, so that the generated result gets closer to the real images. This adversarial training is formulated as an adversarial loss $\mathcal{L}_a$.

In the following, we will give the details of the data and loss functions to accomplish our image decoding task.

B. Adversarial-Based Image Reconstruction

In the decoder side, as illustrated in Fig. 1, our decoder G aims to reconstruct an image as close to the original image as possible based on its compact structure (and color) representation. The main idea is to leverage GAN to learn robust data distribution, which maps our sparse representation back to the original image spaces and benefits both human visual quality and machine visual tasks.

To be specific, we formulate our decoding task as an image-to-image translation by converting our compact representation back to the image domain as input. For the structure representation, it is rendered into a normal bitmap E. Meanwhile, for the color representation, we render the sparsely sampled pixels as a one-channel image mask M where sampled pixels are filled with 1 and others with 0. And finally, another three-channel RGB image C is provided with the color values of the sampled pixels at the corresponding locations. The remaining unknown pixels are set to 0. Then, we are able to solve the image-to-image translation, where x and y in Section V-A are a concatenation of E, M and C, and the original image I, respectively. It can be also viewed as a standard machine vision task of image inpainting augmented with extra edge information. C can be viewed as a masked image I with unknown regions specified by M, *i.e.*, $C = I \odot M$ where $\odot$ is the element-wise multiplication operator.

Exploiting the significant progress of image-to-image translation research, we build our decoder following [30], [40]. The decoder, or generator G, contains fully convolutional layers, where the low-level information is conveyed to the outputs via skip connections from shallow layers to deep layers to enforce the structure and color constraints from the inputs. The discriminator D follows PatchGAN [30] to discriminate the realism of local patches. Specifically, G maps the input of E, M and C to a reconstructed image $I_G = G(E, M, C)$ to approach I in both color, structure and perception senses through a reconstruction loss:

$$\mathcal{L}_r = \mathbb{E}\left[\lambda_1 \|I_G - I\|_1 + \lambda_2 \text{SSIM}(I_G, I) + perc(I_G, I)\right], \quad (2)$$

where the first $\mathcal{L}_1$ term measures the pixel-level discrepancy between the reconstructed image and I, SSIM [41] emphasizes the structural similarity, and perceptual loss [40] further enhances the machine-perceptual quality, weighted by λ_1 and λ_2, respectively. Perceptual loss is computed as

$$perc(I_G, I) = \sum_i \mu_i \left(\|\Phi_i(I_G) - \Phi_i(I)\|_2^2\right), \quad (3)$$

where $\Phi_i(I)$ is the feature map of I in the i-th layer of VGG19 [42] and μ_i is the layer weight.

Finally, we use hinge loss [43] as our adversarial objective function to learn the data distribution:

$$\mathcal{L}_a = \mathcal{L}_G + \mathcal{L}_D, \tag{4}$$

$$\mathcal{L}_G = -\mathbb{E}[D(I_G, E, M)], \tag{5}$$

$$\begin{aligned}\mathcal{L}_D = {} & \mathbb{E}[\text{ReLU}(\tau + D(I_G, E, M))] \\ & + \mathbb{E}[\text{ReLU}(\tau - D(I, E, M))],\end{aligned} \tag{6}$$

where τ is a margin parameter. $\mathcal{L}_G$ and $\mathcal{L}_D$ are adversarial losses for G and D, respectively. Here we use channel-wise concatenation to feed multiple inputs into G and D.

C. Scalable and Controllable Image Reconstruction

Scalability. In Section IV-D, we proposed a novel key reference pixel selection algorithm to calculate a scalable enhanced layer. This section introduces our training scheme to enable our coding framework to accept such scalable layer to further save bit-rate. One intuitive solution in our previous work [17] is to train two separate Gs to reconstruct images with and without $\{M, C\}$, respectively. However, it is storage inefficient and provides only two extreme choices of full or no color information without trade-off.

To further enhance the scalability of our model, we propose a simple yet effective training scheme by randomly discarding reference pixels to simulate various quantities of color cues. Specifically, we generate masks m to mask out reference pixels in M and C. For each data, a patch with random size and location is generated and rendered as a one-channel image mask m where 0 corresponds to the patch region and 1 vice versa. Then we use the updated $M' = M \odot m$ and $C' = C \odot m$ to train the network. Beyond that, other settings are the same as our aforementioned reconstruction process in Section V-B. To this end, our single model could accept scalable enhanced layer and establish a smooth transition between the two extremes of human vision and machine vision. An example is given in Fig. 4, where decoded images without color cues still look realistic. Color cues further supplement the image information, such as the white 'B' in the background.

Controllability. In some image reconstruction tasks such as image super-resolution [44], the designed loss terms emphasize different imagery effects. Generally, the reconstruction loss based on low-level vision similarity creates clean and smooth images for *fidelity*, while the adversarial loss based on high-level vision similarity yields complex and textured images for *realism*. Besides human perception, imagery effects also play an important role in machine vision tasks. For example, the related problem of cartoon-texture image decomposition [45] has been raised to extract the textureless image for sake of better shape analyses. Thus even for the same enhanced layer, it is valuable to investigate an imagery effect controllable model adaptive to different tasks.

To further improve the controllability of our model, we propose to incorporate style-based label control into our generator. Specifically, inspired by adaptive instance normalization (AdaIN) [46] to render images with various styles (or imagery effects) in image style transfer [46], image-to-image translation [47], [48] and image generation domain [49], after each convolution layers except the first and the last ones, we add AdaIN layers to control the imagery effects. A control label $\ell \in [0, 1]$ is introduced, which is mapped to the style parameters for AdaIN via a four-layer multilayer perceptron. During training, the weight λ_2 of SSIM loss is set to $\lambda_2 = f(\ell)$ with $f(\cdot)$ a monotonically increasing function. Therefore, a high ℓ will emphasize the reconstruction loss and force the generator to produce more clean and smooth images as shown in the right column of Fig. 4, while users can input a low ℓ to obtain more realistic images with abundant details and textures as in the left column of Fig. 4.

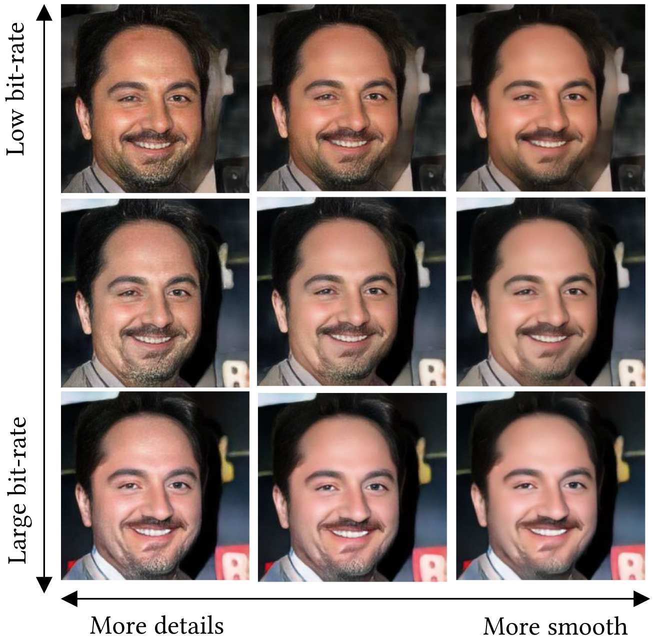
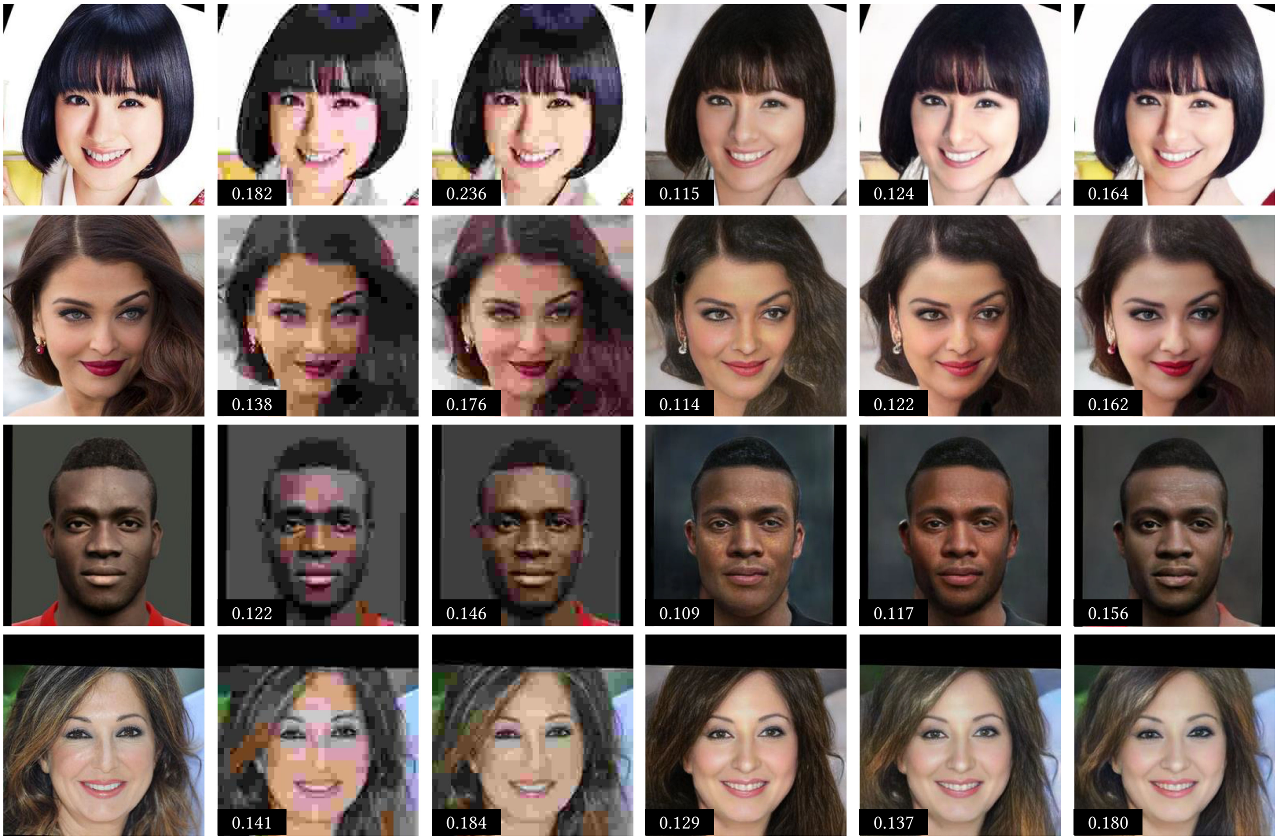


Fig. 4. Illustration of the proposed scalable and controllable image reconstruction. Scalability: from top to bottom, the number of reference pixels used is 0, 15 and 252, respectively. Controllability: from left to right, the control level $\ell = 0.0, 0.5, 1.0$, respectively. Note that all nine results are generated by one single model.

VI. Experimental Results

In this section, we present the experimental results of the proposed method for both the task of human vision and machine vision. We first evaluate our method with respect to human visual quality both qualitatively and quantitatively in Section VI-B. Then we test our method on machine tasks in Section VI-C. Finally, we perform thorough analyses on the proposed scalable color representation extraction and key reference pixel selection in Section VI-D. In addition to the examples included in this paper, more results can be found in our project website[1] and the supplementary material.

[1] [Online]. Available: https://williamyang1991.github.io/projects/VCM-Facev2



(a) input (b) JPEG (qp=4) (c) JPEG (qp=6) (d) our (*N*=0) (e) our (*N*=15) (f) our (*N*=122)

Fig. 5. Visual comparison with JPEG compression. (a) Input image. (b)-(c) Images compressed by JPEG using quality parameter of 4 and 6, respectively. (d) Our decoded images using the encoded edge representations. (e)(f) Our decoded images using both the encoded edge representation and color representation under $N = 15$ and 122, respectively. For each reconstructed image, its bit-rate (bit per pixel, bpp) is shown in the lower left black box.

A. Implementation Details

Dataset. We choose the VGGFace2 [50] dataset for evaluation. We filter the images in VGGFace2 that have small resolution and low quality, and finally use 39 122 images from the training set to train our reconstruction network and 3000 images from the testing set for performance evaluation.

Network architecture. Our generator G utilizes the fully convolutional Encoder-ResBlocks-Decoder architecture as in [40]. Specifically, G is made up of four encoding convolution layers, seven resblocks and four decoding convolution layers. Skip connections [30] are added between the Encoder and the Decoder to preserve the low-level color information. Each convolutional layer is followed by AdaIN layer [46] except the first and the last layer. Meanwhile, the discriminator D follows PatchGAN architecture as in [30] with seven convolution layers and we add Spectral Normalization layers [51] for stable and fast training.

Parameter setting and network training. For key reference pixel selection, we set removing ratio $k = 8$, sparse radius $r = 10$, min removing number $n_0 = 10$, and threshold number $n = 40$. To train our network, we set $\lambda_1 = 100$, and $\tau = 10$. To compute perceptual loss, we use the conv2_1 and conv3_1 layers of the VGG19 [52] trained on ImageNet dataset [42] with $\mu_1 = 1.0$ and $\mu_2 = 0.5$, respectively. The hyper-parameter λ_2 to determine the signal fidelity is controlled by the input parameter ℓ through $\lambda_2 = 1000\ell^3 + 50$. Note that we use exponential function rather than linear function because we found that λ_2 has diminishing marginal effects on the generated image. Exponential function makes the image change more linearly with respect to ℓ. During training, we first train our model with a fixed $\ell = 0.0$ for 10 epoches. Then we train our model with ℓ uniformly sampled from [0,1] for another 10 epoches. Finally, we finetune our model with a fixed $\ell = 0.0$ for 1 epoches, which we found could effectively eliminate the artifacts created by GAN. For reference pixel discarding, we use all, partial (using mask m) and no color information with probabilities of 0.6, 0,1 and 0.3, respectively. We report the best result from the results of $\ell \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$ in the following experiments.

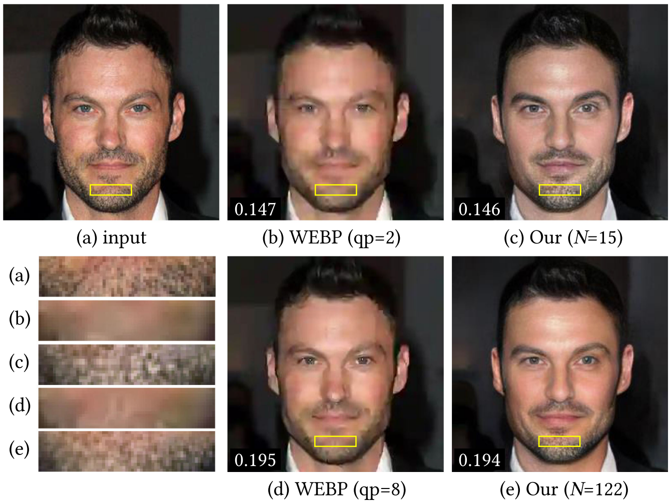
B. Human Vision: Visual Quality Evaluation

Qualitative evaluations. In Figs. 5-6, we present a visual comparison of the proposed method with JPEG and WEBP compression under different quality parameters (qp), which are selected to matches the bit-rate of our method for a fair comparison. Specifically, as summarized in Table I, the bit-rate of JPEG compression with $qp = 4$ corresponds to that of our compression under $N = 15$, while $qp = 6$ corresponds to $N = 122$ (optimal N

TABLE I

QUANTITATIVE COMPARISON WITH JPEG AND WEBP ON BOTH HUMAN VISION TASK AND MACHINE VISION TASK. WE SET $\ell = 0.0$ FOR COMPUTING PERCEPTUALH, NME AND MEMORABILITY. WE SET $\ell = 0.25$ FOR LPIPS, DISTS, AND FID. WE SET $\ell = 0.5$ FOR ACCURACY OF GENDER CLASSIFICATION. WE SET $\ell = 1.0$ FOR COMPUTING SSIM, PSNR, AND PERCEPTUALL. WE INDICATE FOR EACH METRIC WHETHER HIGHER (⇑) OR LOWER (⇓) VALUES ARE MORE DESIRABLE. BEST SCORES ARE HIGHLIGHTED IN BOLD

Method	Bit-Rate	Human Vision							Machine Vision		
	bpp ⇓	SSIM ⇑	PSNR ⇑	perceptualL ⇓	perceptualH ⇓	LPIPS ⇓	DISTS ⇓	FID ⇓	NME ⇓	gender ⇑	memorability ⇑
JPEG ($qp = 4$)	0.153	0.642	23.646	81.313	36.592	0.419	0.363	102.02	36.473	0.762	0.828
WEBP ($qp = 0.4$)	0.154	**0.798**	**28.216**	**42.688**	20.884	0.215	0.219	47.859	4.392	0.902	0.880
our ($N = 15$)	**0.152**	0.706	21.574	57.812	**17.677**	**0.208**	**0.179**	**25.949**	**3.042**	**0.917**	**0.900**
JPEG ($qp = 6$)	0.197	0.735	25.726	64.975	29.913	0.317	0.330	57.194	6.062	0.851	0.867
WEBP ($qp = 3$)	0.197	**0.827**	**29.352**	**36.789**	16.719	0.174	0.193	40.928	3.430	**0.932**	0.886
our ($N = 122$)	**0.191**	0.759	23.034	49.267	**15.233**	**0.165**	**0.161**	**19.729**	**2.748**	0.924	**0.898**



(a) input (b) WEBP (qp=2) (c) Our (*N*=15)

(d) WEBP (qp=8) (e) Our (*N*=122)

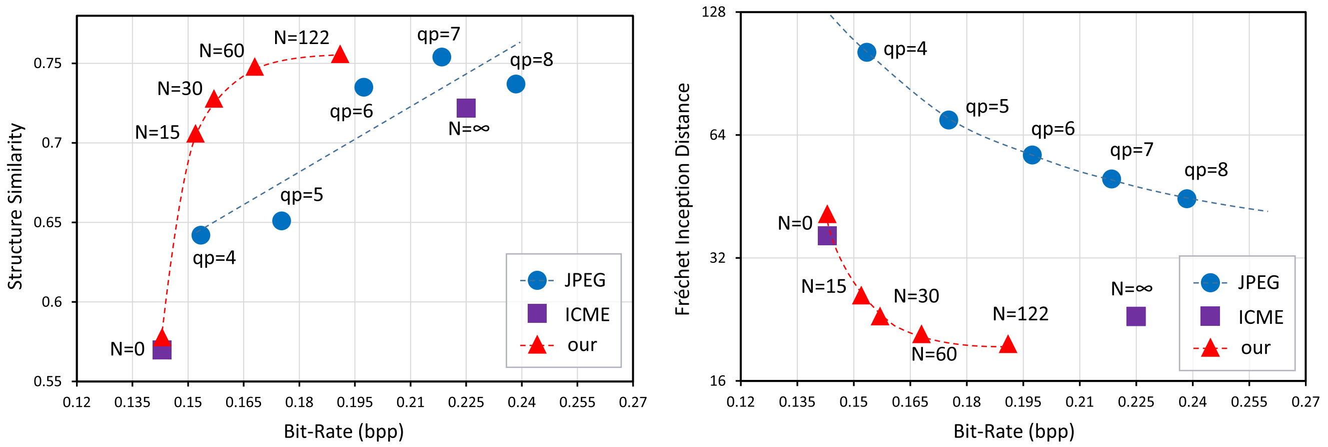
Fig. 6. Visual comparison with WEBP compression. (a) Input image. (b)(d) Images compressed by WEBP using quality parameter of 2 and 8, respectively. (c)(e) Our decoded images under $N = 15$ and 122, respectively. For each reconstructed image, its bit-rate (bit per pixel, bpp) is shown in the lower left black box. The local regions in the yellow box are enlarged below the input image for better visual comparison.

in terms of SSIM as illustrated in Fig. 3). It can be observed that JPEG compression yields distinct block artifacts, which greatly decrease visual quality. Furthermore, the quantization caused severe color distortion, making a lot of pink areas appear on the face in Fig. 5(b). By comparison, our method produces more natural results. We have also included our reconstructed images decoded without any color cues for extreme compression, and our model successfully renders plausible colors on all images. WEBP [53] is a modern image format that provides both lossless and lossy compression for images on the web. In Fig. 6, we set $qp = 2$ and $qp = 8$ to match the bit-rate of our results. It can be seen that our method generates much more vivid details of the beard, while WEBP produces very blurry images.

Quantitative evaluations. To quantitative evaluate the visual quality of the proposed method, we compare with JPEG and WEBP compression in terms of fidelity and realism. The signal fidelity is measured in terms of SSIM, PSNR, perceptual loss, LPIPS [54] and DISTS [55]. First, SSIM considers the luminance, contrast and structure similarities between the decoded image and the original image. The SSIM results are reported in Table I and Fig. 7(a). Under similar bit-rates, the proposed method achieves an improvement of 0.064 and 0.021 on SSIM over JPEG compression under $qp = 4$ and $qp = 6$, respectively. Compared to our previous work [17] (denoted as ICME), the scalability enables us to add very limited 15 reference pixels to bring significant gains. Also, by selecting the most important 122 reference pixels rather than the whole pixels ($N = \infty$), our method fulfills both higher SSIM and lower bit-rates. PSNR is based on the mean squared error (MSE) between the decoded image and the original image. Not surprisingly, JPEG compression are designed in favor of MSE and achieves higher PSNR. The proposed method suffers from the edge misalignment caused by edge vectorization. PSNR is very sensitive to such errors. On the other hand, the advanced WEBP compression surpasses our method in both SSIM and PSNR for it can better preserve the low frequency information, whose results are, however, very blurry in Fig. 6. The discrepancy between SSIM/PSNR and human eyes suggests the design of robust evaluation metrics for VCM, which we give a brief discussion in Sec. VII.

Some attempts have been made to design robust metrics such as perceptual loss, LPIPS [54] and DISTS [55]. Perceptual loss [40] computes the mean squared error in the VGG feature spaces, where the VGG network is trained on ImageNet dataset [42] for natural image classification with 1000 classes. The diverse classes make VGG features highly generalized, and perceptual loss is known to better reflect human perception. In Table I, we report the perceptual loss of shallow conv2_2 layer (denoted as “perceptualL”) and deep conv5_1 layer (denoted as “perceptualH”), respectively. Interestingly, different layers give very different ranks. Shallow layers prefer low-level pixel-wise similarities while deep layers incline towards high-level perceptual similarities. Therefore, our method in favor of high-level machine vision tasks obtains lower perceptual errors in perceptualH. LPIPS [54] and DISTS [55] are two state-of-the-art full-reference image quality assessment metrics designed to match the human perception. It can be clearly seen that the proposed method excels all other comparison methods on LPIPS and DISTS under similar bit-rates, indicating perceptually better image reconstruction.

Meanwhile, we use Fréchet inception distance (FID) [56] to evaluate the realism of the decoded images. In image generation



(a) Structure similarity

(b) Fréchet inception distance

Fig. 7. Illustration of the SSIM, FID and bit-rate of JPEG compression, our previous work [17] and the proposed method.

TABLE II
THE PREFERENCE RATIO ON REALISM AND FIDELITY OF DIFFERENT METHODS AT DIFFERENT BIT-RATE

Method	Bit-Rate (bpp)	Realism	Fidelity
JPEG ($qp=4$)	0.158	0.003	0.197
our ($N=15$)	**0.145**	**0.997**	**0.803**
JPEG ($qp=6$)	0.204	0.007	0.143
our ($N=122$)	**0.185**	**0.993**	**0.857**

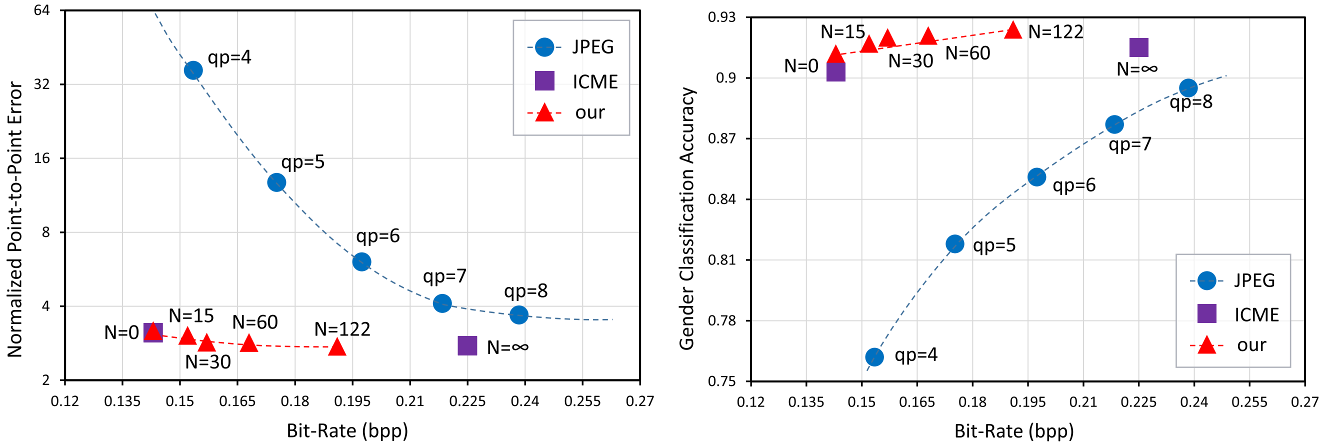
domain, FID compares the statistics of generated samples to real samples, and lower FID corresponds to more realistic and diversified generated samples. The FID results are reported in Table I and Fig. 7(b), where the advantages of the proposed method is more distinct. The annoying block artifacts in JEPG images and the blurriness in WEBP images drastically harm the realism. Compared to our previous work [17], under extreme case where no enhanced layer is used ($N=0$), our method has a little disadvantage. The reason might be that we train separate networks for $N=0$ and $N=\infty$ in [17], which better adapts to either task, while our improved network has to deal with scalable inputs. However, in common cases where the enhanced layer is available, our method is still superior.

To better understand the performance of the compared methods, we perform user studies. We uniformly sample 25 images from our testing set for evaluation (their serial numbers are $60+120i$, $i=0,1,\ldots,24$). We compare our results under $N=15$ with JPEG compression under $qp=4$, as well as our results under $N=122$ with JPEG compression under $qp=6$. Each subject is asked to select one from the two results that looks the most realistic (realism) and best matches the original image (fidelity). A total of 12 subjects participate in this study and a total of 1200 selections are tallied. The preference ratio is used as the evaluation metric. It is calculated as the ratio of a method selected in all comparisons with this method. As shown in Table II, the proposed method uses lower bit-rates than JPEG compression on these 25 testing images. In general, user scores well match the perceptualH, LPIPS, DISTS and FID scores. Similar to FID, the proposed method has a distinct advantage in decoding realistic images, obtaining the best average preference ratio of 0.993 and 0.997 under high and low bit-rates, respectively. In terms of fidelity, the proposed method mostly outperforms JPEG compression under the similar bit-rate. The advantage is not as overwhelming as realism. The reason might be that approximating edges into straight lines and Bézier curves will change subtle facial structures, to which humans are sensitive. It gives us a direction for future work. Overall, the user study quantitatively verifies the superiority of our method.

C. Machine Vision: Landmark Detection

The machine vision performance of our method is verified on three high-level tasks: facial landmark detection, gender classification, and image memorability prediction.

We perform facial landmark detection [57] on the original VGGFace2 [50] dataset, the reconstructed dataset by JPEG, our previous work [17] and our method. Detection results on the original data are served as ground truth. We then calculate the normalized point-to-point error (NME) [58] between the detection results on the compressed data and the ground truth. Table I illustrates the averaged NME and the bit-rate of the compared methods and our method. It can be clearly seen that our method achieves much fewer errors at the similar bit-rate compared to JPEG and WEBP compression. Specifically, NME of our method with few color cues ($N=15$) is only 3.042%, which is 33.43% lower than JPEG under $qp=4$ and 1.35% lower than WEBP under $qp=0.4$. Meanwhile, with more color cues ($N=122$), our method achieves merely 2.748% NME, 3.31% lower than JPEG under $qp=6$ and 0.68% lower than WEBP under $qp=3$. In addition, we further save 0.034 bpp compared to our previous work [17]. Another observation is the stable performance under various bit-rate in Fig. 8(a), indicating that the facial landmark detection is color-robust and favors our color-scalable framework. Fig. 9 further shows the cumulative error distribution, where more than 90% of the images reconstructed by the proposed method have tiny errors less than 5%,



(a) Normalized point-to-point error (NME) on facial landmark detection

(b) Gender classification accuracy

Fig. 8. Illustration of the NME, gender classification accuracy and bit-rate of JPEG compression, our previous work [17] and the proposed method.

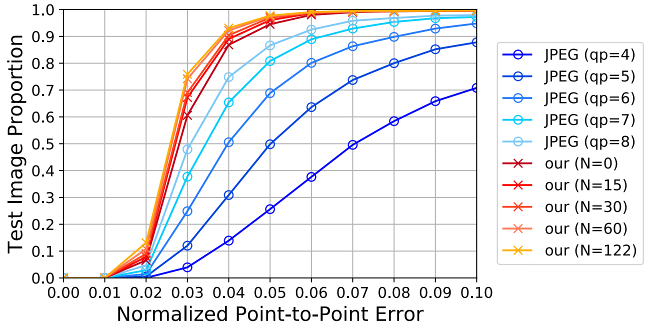
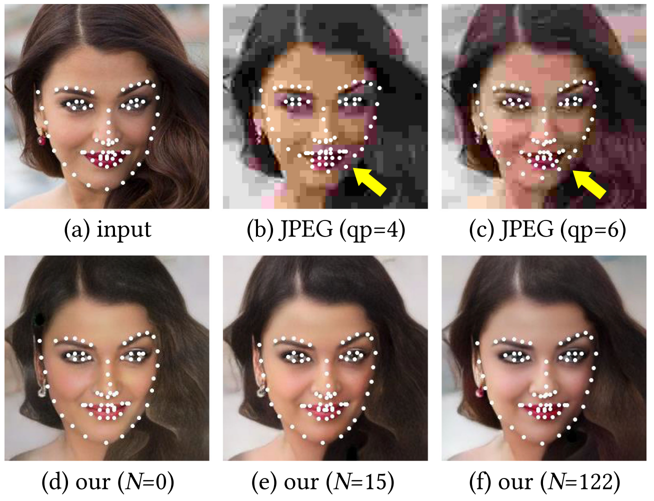


Fig. 9. Cumulative error distribution of JPEG compression and the proposed method on facial landmark detection.



(a) input

(b) JPEG (qp=4)

(c) JPEG (qp=6)

(d) our (*N*=0)

(e) our (*N*=15)

(f) our (*N*=122)

Fig. 10. Visual comparison with JPEG compression on facial landmark detection. The detected landmarks are shown as white circles.

showing great robustness. In Fig. 10, we give an example of landmark detection results. It can be clearly observed that the block artifacts in JPEG compression lead to insufficient distinction between chin and neck, making the algorithm fail to estimate the contour of the faces as indicated by the yellow arrows. On the other hand, the proposed method generates more natural results, thus facilitating the landmark detection.

We exploit deepface [59] to conduct gender classification on the original VGGFace2 [50] dataset, the reconstructed dataset by JPEG, our previous work [17] and our method. We first use deepface to detect the face in the image and then perform gender classification over the face region. Original images with no face detected are discarded. Classification results on the remaining original images are served as ground truth. The corresponding reconstructed images with no face detected are directly regarded as a failure classification. Finally, the classification accuracy on the testing set is reported in Table I and Fig. 8(b). Our method achieves much higher accuracy at the similar bit-rate compared to JPEG compression. Under similar bit-rate, we increase the classification accuracy by 15.5% and 7.3%, compared to JPEG compression at $qp = 4$ and $qp = 6$, respectively. It can be also observed that the proposed method is less affected by the bit-rate limit compared to JPEG compression on this task, *i.e.*, our classification accuracy drops only a little bit under less color information. Our method is inferior at $qp = 3$ but surpassed WEBP at $qp = 0.4$. It implies that our method has more obvious advantages in color-robust machine vision tasks.

Finally, we conduct visual memorability prediction based on MemNet [60] to estimate whether the reconstructed images are memorable. Table I suggests that our results with rich facial details preserved are more memorable than JPEG and WEBP compressed images.

D. Performance Analysis

In this section, we experimentally analyze the performance of the proposed sparse reference pixel extraction and scalable color representation extraction.

Sparse reference pixel extraction. Section IV-C introduces how we find candidate reference pixels for color information based on their relative position to the extracted edges. It enables the decoder to infer their positions as the encoder does, thus saving the bit-rate to record pixel coordinates. To verify such

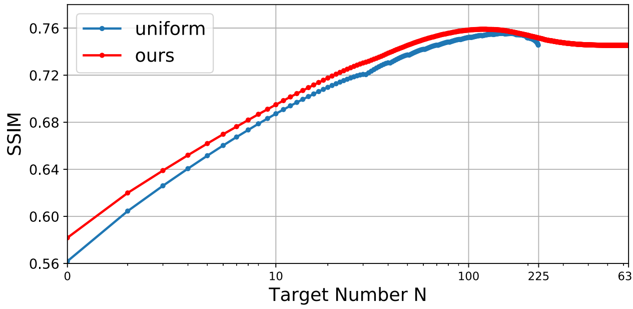


Fig. 11. Compare with reference pixels extracted by uniform sampling. The x-axis uses a logarithmic axis for better visual comparison.

TABLE III
COMPARE WITH REFERENCE PIXELS EXTRACTED BY UNIFORM SAMPLING ON SUB TESTING SET AND FULL TESTING SET

Dataset	SSIM ($N = \infty$)		optimal N		SSIM (optimal N)	
	uniform	ours	uniform	ours	uniform	ours
Subset1	**0.742**	0.740	148	**121**	0.754	**0.757**
Subset2	**0.746**	0.745	147	**116**	0.757	**0.761**
Subset3	**0.744**	0.743	148	**124**	0.755	**0.759**
Full set	**0.744**	0.743	148	**122**	0.756	**0.759**

pixel selection adaptive to the edge maps provides more representative and informative color cues, we conduct a comparative experiment. For a fair comparison, we choose the uniform reference pixel sampling, which do not need to record pixel coordinates, either. Specifically, for 256×256 images, considering the average number of reference pixels extracted by our method is $224 \approx 15 \times 15 = 225$, we uniformly sample 225 pixels with their coordinates $(16i, 16j)$ where $i, j = 1, 2, \ldots, 15$. Then, a separate GAN is trained on the new enhanced layer derived from these pixels to adapt to such extraction strategy. Next, we utilize the proposed key reference pixel selection algorithm to calculate the priority of the reference pixels extracted by our method and uniform sampling and plot the changes in SSIM with color cues gradually reduced in Fig. 11. It can be clearly observed that after removing redundant pixels, our remaining pixels are more informative to preserve SSIM, especially when $N < 100$. In Table III, we divide 3000 testing images in order into three subsets, each containing 1000 images and report the SSIM with and without pixel removal on both full testing set and its subsets. Without pixel removal, uniform sampling and the proposed extraction method have comparable performance in SSIM and bit-rate. Clearly, both of two pixel extraction strategies benefit from redundant pixel removal. However, the optimal number of reference pixels needed to achieve highest SSIM by our method is much fewer than that by uniform sampling. Not to mention that under such condition of fewer bit-rate, our results even have a slightly higher SSIM. The results come to a conclusion that our sparse reference pixel extraction based on edge maps can find more informative and important color cues.

Comparing three subsets and the full testing set in Table III, we find that the optimal N is quite stable regardless of the diversity of the images. Therefore, we could expect $N = 122$ as a default number of reference pixels to work fine for images beyond our testing set.

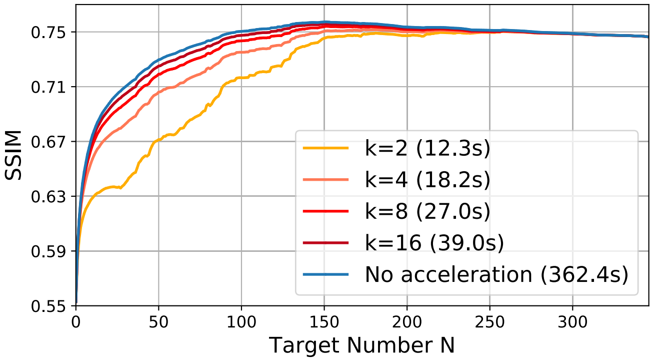


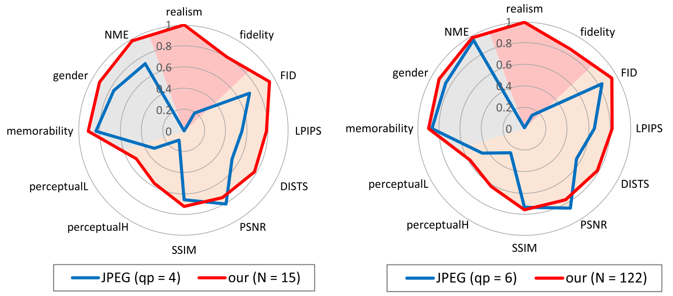
Fig. 12. Acceleration performance of key reference pixel selection algorithm.

Scalable color representation extraction. In Section IV-D, we proposed a key reference pixel selection algorithm where a removing ratio parameter k is used to determine how many pixels to be roughly removed for acceleration. A small k means more pixels are roughly removed each time, thus speeding up the algorithm. In Fig. 12, we study how k compromises between accuracy and running time on 10 uniformly sampled images (their serial numbers are $150 + 300i$, $i = 0, 1, \ldots, 9$) in the testing set. The decoder is tested on a GeForce GTX 1080 Ti GPU. Our baseline without acceleration requires 362.4 s per image to estimate all of its reference pixel priorities. By setting $k = 16$, the proposed method achieves a speedup of about 10 times. The acceptable value for k is within [8,16] and the performance will drop if k is smaller than 8. In this paper, we set a fixed $k = 8$, with average running time of 27.0 s, which occupies the main amount of computational complexity. In terms of other components in our framework, the average running time of edge extraction, edge vectorization, bit-stream generation is about 0.01 s, 0.05 s and 0.03 s per image. In the decoder side, the network requires about 0.01 s and 2.44 s per image on GeForce GTX 1080 Ti GPU and Intel Xeon E5-2650 CPU, respectively. There is still some acceleration space for future work.

Another parameter, sparse radius r to prevent reference pixels in a local region are all removed at once, is set to 10 in this paper. We experimentally find that for $r = 0$ and $r = 10$, the optimal number of reference pixels N are all 122. The corresponding SSIM without sparse removal requirement is 0.758, which is slightly lower than 0.759 using $r = 10$.

VII. DISCUSSION AND FUTURE WORK

Dataset for VCM. In VCM, there is still a lack of multi-task datasets that support both human vision and machine vision to evaluate the performance of image coding and feature coding methods. Specifically, the common image coding datasets are mainly designed for human vision tasks and lack high-level vision label; while previous attempts towards VCM focus on feature coding for machine vision tasks and could not reconstruct the images. This paper presents the first attempt towards VCM to support machine vision and human vision simultaneously, and



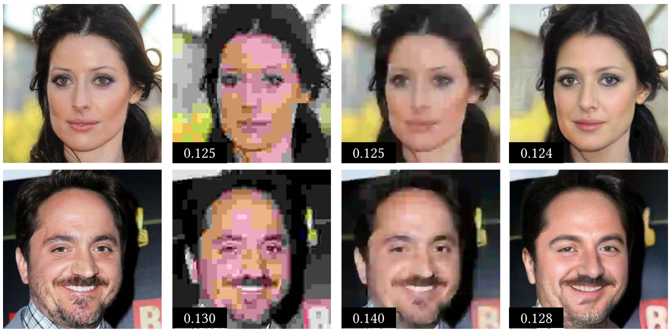
(a) Compare with JPEG ($qp = 4$) (b) Compare with JPEG ($qp = 6$)

Fig. 13. Radar charts visualizing Table I and II. Values have been normalized to the unit range, and axes inverted so that higher value is always better. The red, orange and gray regions represent the human vision subjective scores, human vision objective scores and machine vision scores, respectively.

TABLE IV
PERFORMANCE OF OUR METHOD USING DIFFERENT BACKBONES. WE USE THE SAME HYPER-PARAMETER SETTINGS AS IN TABLE I

Method	Bit-Rate	Human Vision		Machine Vision	
	bpp ⇓	LPIPS ⇓	FID ⇓	NME ⇓	gender ⇑
JPEG ($qp = 1$)	0.123	0.505	187.33	174.20	0.654
WEBP ($qp = 0$)	0.124	0.256	56.984	5.783	0.858
our ($N = 60$)	0.168 / **0.112***	**0.174**	**20.863**	**2.848**	**0.921**
JPEG ($qp = 3$)	**0.133**	0.478	159.81	109.42	0.691
WEBP ($qp = 0.1$)	0.141	0.232	51.143	4.645	0.893
our ($N = 122$)	0.191 / 0.135*	**0.165**	**19.729**	**2.748**	**0.924**

* "bpp1 / bpp2" are our bit-rates using PPM and Brotli backbones, respectively.



(a) input (b) JPEG (qp=1) (c) WEBP (qp=0) (d) our (*N*=60)

Fig. 14. Visual comparison between JPEG, WEBP and our method using Brotli backbone. The lower left black box shows the corresponding bit-rate.

provides a preliminary and new exploration direction under the VCM standard, which places higher demands on data. In this paper, we choose the face dataset for two reasons. First, human faces have an important research value in image and video coding. More importantly, human faces have been well studied in computer vision community, and there are rich related classification and detection models that can provide labels for high-level machine vision tasks. Nevertheless, due to the lack of suitable datasets, we compromise to use the label predicted from the original images as the ground truth, which might contain noisy labels. It is our future research direction to provide a more comprehensive dataset with accurate labels on more diversified tasks such as age estimation, emotion classification, face parsing, and eye tracking.

Quality evaluation. In addition to multi-task datasets, it is also an important and challenging unresolved problem to evaluate the performance of image coding and feature coding methods under the VCM paradigm. In our experimental results, we found a great discrepancy between the subjective scores of the human eyes in the user study and the traditional standard evaluation metrics such as PSNR. Our method obtains lower PSNR but much higher user preferences compared to JPEG compression. It shows that some traditional metrics cannot well reflect the human visual perception, and how to define a fair quality evaluation for VCM should be considered. In this paper, we have conducted a preliminary exploration towards quality evaluation by combining diversified human vision and machine vision metrics. For human vision, we select FID, perceptual loss, LPIPS and DISTS. User study in terms of realism and fidelity is further included. For machine vision, we report results on three different high-level tasks.

However, a large number of scores makes it less intuitive to evaluate the overall performance of a method. Here we suggest a potentially fair representation for quality evaluation: the radar chart. In Fig. 13, we normalize the scores in Table I and II to the unit range and visualize them in the radar chart. Regions are colored according to the category of vision tasks. Fig. 13 intuitively displays the comprehensive performance of the methods under multiple tasks in a single figure, where our method generally surpasses JPEG compression especially in human vision subjective scores. We believe that such a form will inspire the design of subsequent quality evaluation for VCM. More details of the radar chart can be found in the supplementary material.

Advanced submodules. This paper provides a new VCM framework using some basic modules, and verifies its effectiveness on the face data. Each of our modules can be replaced with more advanced ones. For example, the PPM scheme can be replaced with more advanced Brotli [61] to losslessly compress edge vectors. Specifically, we separate each vector in the SVG descriptions into operation markers (*i.e.*, Move, Line, and Curve) and the corresponding numerical parameters. We compress the markers using Huffman coding. We then utilize Brotli [61] to compress the parameters. The two components are concatenated to form the edge bit-stream. In Table IV and Fig. 14, we show visual and quantitative results of our method using the Brotli backbone. This improvement brings a bit-rate save of 0.056 bpp. Thus, our method ($N \leq 60$) supports extreme compression that is beyond the capability of JPEG ($qp = 1$) and WEBP ($qp = 0$), which better meets the limited bit-rate requirement of VCM. Likewise, other submodules such as our GAN-based decoder can benefit from more advanced backbones, which implies a huge room for performance improvement.

VIII. CONCLUSION

In this paper, we present a new image coding framework to facilitate both human vision and machine vision. The input image is first analyzed and compressed as the compact structure and scalable color representations. Leveraging the advanced generative model in machine vision, we train a network to faithfully reconstruct images from the compact representations in a scalable and controllable manner. Experimental results demonstrate the superiority of the proposed method in both human vision tasks in terms of visual quality and fidelity and machine vision tasks of facial landmark detection, gender classification and memorability prediction. This paper presents the first attempt towards VCM with respective to image coding via scalable feature-based compression.

References

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

[2] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. xviii-xxxiv, Feb. 1992.

[3] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consum. Electron.*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.

[4] J. Balle, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," in *Proc. Conf. Learn. Representations*, 2017, pp. 1–12.

[5] J. Balle, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *Proc. Conf. Learn. Representations*, 2018, pp. 1–13.

[6] S. Xia, K. Liang, W. Yang, L.-Y. Duan, and J. Liu, "An emerging coding paradigm VCM: A scalable coding approach beyond feature and signal," in *Proc. IEEE Conf. Multimedia Expo*, 2020, pp. 1–6.

[7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Computer Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[8] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 4263–4270.

[9] L.-Y. Duan *et al.*, "Overview of the MPEG-CDVS standard," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 179–194, Jan. 2016.

[10] L.-Y. Duan *et al.*, "Compact descriptors for video analysis: The emerging MPEG standard," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 44–54, Apr./Jun. 2019.

[11] L.-Y. Duan, J. Liu, W. Yang, T. Huang, and W. Gao, "Video coding for machines: A paradigm of collaborative compression and intelligent analytics," *IEEE Trans. Image Process.*, to be Published, doi: 10.1109/TIP.2020.3016485.

[12] B. Xiong, X. Fan, C. Zhu, X. Jing, and Q. Peng, "Face region based conversational video coding," *IEEE Trans. Circuits Syst. Vid. Technol.*, vol. 21, no. 7, pp. 917–931, Jul. 2011.

[13] M. Elad, R. Goldenberg, and R. Kimmel, "Low bit-rate compression of facial images," *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2379–2383, Sep. 2007.

[14] B. Erol and F. Kossentini, "Automatic key video object plane selection using the shape information in the MPEG-4 compressed domain," *IEEE Trans. Multimedia*, vol. 2, no. 2, pp. 129–138, Jun. 2000.

[15] T. Meier and K. N. Ngan, "Automatic segmentation of moving objects for video object plane generation," *IEEE Trans. Circuits Syst. Vid. Technol.*, vol. 8, no. 5, pp. 525–538, Sept. 1998.

[16] S. McPhee. (2020, December) *NVIDIA Announces cloud-AI Video-Streaming Platform to Better Connect Millions Working and Studying Remotely*. [Online]. Available: https://nvidianews.nvidia.com/news/nvidia-announces-cloud-ai-video-streaming-platform-to-better-connect-millions-working-and-studying-remotely

[17] Y. Hu, S. Yang, W. Yang, L.-Y. Duan, and J. Liu, "Towards coding for human and machine vision: A scalable image coding approach," in *Proc. IEEE Conf. Multimedia Expo.*, 2020, pp. 1–6.

[18] K. Gregor, F. Besse, D. J. Rezende, I. Danihelka, and D. Wierstra, "Towards conceptual compression," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3549–3557.

[19] L. Theis, W. Shi, A. Cunningham, and F. Huszar, "Lossy image compression with compressive autoencoders," in *Proc. Conf. Learn. Representations*, 2017, pp. 1–12.

[20] J. Chang *et al.*, "Layered conceptual image compression via deep semantic synthesis," in *Proc. IEEE Conf. Image Process.*, 2019, pp. 694–698.

[21] R. Torfason, *et al.*, "Towards image understanding from deep compression without decoding," in *Proc. Conf. Learn. Representation*, 2018, pp. 1–12.

[22] S. Wang, *et al.*, "Scalable facial image compression with deep feature reconstruction," in *Proc. IEEE Conf. Image Process.*, 2019, pp. 2691–2695.

[23] H. Choi and I. V. Bajic, "High efficiency compression for object detection," in *Proc. IEEE Conf. Acoust., Speech, Signal Process.*, 2018, pp. 1792–1796.

[24] T. He, S. Sun, Z. Guo, and Z. Chen, "Beyond coding: Detection-driven image compression with semantically structured bit-stream," in *Proc. Picture Coding Symp.*, 2019, pp. 1–5.

[25] L. Bragilevsky and I. V. Baji, "Tensor completion methods for collaborative intelligence," *IEEE Access*, vol. 8, pp. 41162–41174, 2020.

[26] R. A. Cohen, H. Choi, and I. V. Baji, "Lightweight compression of neural network feature tensors for collaborative intelligence," in *Proc. IEEE Conf. Multimedia Expo.*, 2020, pp. 1–6.

[27] S. R. Alvar and I. V. Baji, "Bit allocation for multi-task collaborative intelligence," in *Proc. IEEE Conf. Acoust., Speech, Signal Process.*, 2020, pp. 4342–4346.

[28] S. R. Alvar and I. V. Baji, "Multi-task learning with compressible features for collaborative intelligence," in *Proc. IEEE Conf. Image Process.*, 2019, pp. 1705–1709.

[29] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[30] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1125–1134.

[31] J.-Y. Zhu *et al.*, "Toward multimodal image-to-image translation," in *Adv. Neural Inf. Process. Syst.*, 2017, pp. 465–476.

[32] T. Dekel, C. Gan, D. Krishnan, C. Liu, and W. T. Freeman, "Sparse, smart contours to represent and edit images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3511–3520.

[33] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2536–2544.

[34] S. Xu, D. Liu, and Z. Xiong, "E2I: Generative inpainting from edge to image," *IEEE Trans. Circuits Syst. Vid. Technol.*, to be Published, doi: 10.1109/TCSVT.2020.3001267.

[35] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, "EdgeConnect: Structure guided image inpainting using edge prediction," in *Proc. Conf. Comput. Vis. Workshops*, 2019, pp. 1–10.

[36] P. Dollar and C. L. Zitnick, "Structured forests for fast edge detection," in *Proc. Conf. Comput. Vis.*, 2013, pp. 1841–1848.

[37] J. Xu, R. Joshi, and R. A. Cohen, "Overview of the emerging hevc screen content coding extension," *IEEE Trans. Circuits Syst. Vid. Technol.*, vol. 26, no. 1, pp. 50–62, Jan. 2016.

[38] M. Weber. (2004, Jan.) *AutoTrace: A Program for Converting Bitmap to Vector Graphic*. [Online]. Available: http://autotrace.sourceforge.net/

[39] J. Cleary and I. Witten, "Data compression using adaptive coding and partial string matching," *IEEE Trans. Commun.*, vol. 32, no. 4, pp. 396–402, Apr. 1984.

[40] J. Johnson, A. Alahi, and F. F. Li, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.

[41] Z. Wang, A. C. Bovik, H. R. Sheikh, 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.

[42] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[43] J. Yu, *et al.*, "Free-form image inpainting with gated convolution," in *Proc. Conf. Comput. Vis.*, 2019, pp. 4471–4480.

[44] X. Wang, K. Yu, C. Dong, X. Tang, and C. C. Loy, "Deep network interpolation for continuous imagery effect transition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1692–1701.

[45] S. Ono, T. Miyata, and I. Yamada, "Cartoon-texture image decomposition using blockwise low-rank texture characterization," *IEEE Trans. Image Process.*, vol. 23, no. 3, pp. 1128–1142, Mar. 2014.

[46] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," *in Proc. Conf. Comput. Vis.*, 2017, pp. 1501–1510.

[47] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 172–189.

[48] S. Yang, Z. Wang, J. Liu, and Z. Guo, “Deep plastic surgery: Robust and controllable image editing with human-drawn sketches,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 1–17.

[49] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4401–4410.

[50] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, “VGGFace2: A dataset for recognising faces across pose and age,” in *Proc. IEEE Conf. Autom. Face Gesture Recognit.*, 2018, pp. 67–74.

[51] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, “Spectral normalization for generative adversarial networks,” *in Proc. Conf. Learn. Representations*, 2018, pp. 1–12.

[52] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *Proc. Conf. Learn. Representations*, 2015, pp. 1–10.

[53] R. Rabbat. (2010, October) *WebP, a New Image Format for the Web*. [Online]. Available: https://developers.google.com/speed/webp

[54] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 586–595.

[55] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli. (2020, April) *Image Quality Assessment: Unifying Structure and Texture Similarity*. [Online]. Available: https://arxiv.org/abs/2004.07728

[56] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in *Adv. Neural Inf. Process. Syst.*, 2017, pp. 6626–6637.

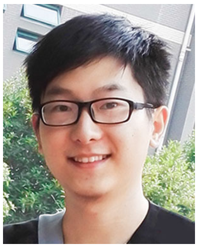
[57] A. Bulat and G. Tzimiropoulos, “How far are we from solving the 2D & 3D face alignment problem?(and a dataset of 230000 3D facial landmarks),” in *Proc. Conf. Comput. Vis.*, 2017, pp. 1021–1030.

[58] D. Cristinacce and T. F. Cootes, “Feature detection and tracking with constrained local models.” in *Proc. Brit. Mach. Vis. Conf.*, 2006, pp. 1–10.

[59] S. I. Serengil and A. Ozpinar, “Lightface: A hybrid deep face recognition framework,” in *Proc. IEEE Innovations Intell. Syst. Appl. Conf.*, 2020, pp. 1–5.

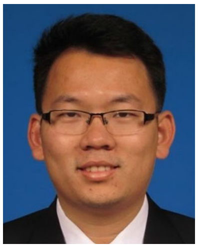
[60] A. Khosla, A. S. Raju, A. Torralba, and A. Oliva, “Understanding and predicting image memorability at a large scale,” in *Proc. Conf. Comput. Vis.*, 2015, pp. 2390–2398.

[61] J. Alakuijala *et al.*, “Brotli: A general-purpose data compressor,” *ACM Trans. Inf. Syst.*, vol. 37, no. 1, pp. 1–30, 2018.



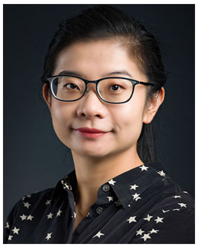
Shuai Yang (Member, IEEE) received the B.S. and Ph.D. degrees (Hons.) in computer science from Peking University, Beijing, China, in 2015 and 2020, respectively. He is currently a Postdoctoral Research Fellow with NTU AI Corporate Laboratory, Nanyang Technological University, Singapore. From 2018 to 2019, he was a Visiting Scholar with the Texas A&M University, College Station, TX, USA. From March 2017 to August 2017, he was a Visiting Student with the National Institute of Informatics, Tokyo, Japan. His current research interests include image stylization and image inpainting. He was the recipient of the IEEE ICME 2020 best paper awards and IEEE MMSP 2015 Top10% paper awards.

Yueyu Hu (Student Member, IEEE) received the B.S. degree in 2018 in computer science from Peking University, Beijing, China, where he is currently working toward the master’s degree with the Wangxuan Institute of Computer Technology. His current research interests include video and image compression, and enhancement with machine learning.



Wenhan Yang (Member, IEEE) received the B.S. degree and Ph.D. degree (Hons.) in computer science from Peking University, Beijing, China, in 2012 and 2018, respectively. He is currently a Postdoctoral Research Fellow with the Department of Computer Science, City University of Hong Kong, Hong Kong. From 2015 to 2016 and from September 2018 to November 2018, he was a Visiting Scholar with the National University of Singapore, Singapore. His current research interests include deep-learning based image processing, bad weather restoration, and related applications and theories.

Ling-Yu Duan (Member, IEEE) received the Ph.D. degree in information technology from The University of Newcastle, Callaghan, NSW, Australia, in 2008. He is currently a Full Professor with the National Engineering Laboratory of Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing, China, and since 2012, he has been the Associate Director of the Rapid-Rich Object Search Laboratory, a joint lab between Nanyang Technological University, Singapore, and Peking University. Since 2019, he has been with Peng Cheng Laboratory, Shenzhen, China. He has authored or coauthored about 200 research papers. His research interests include multimedia indexing, search, and retrieval, mobile visual search, visual feature coding, and video analytics. He was the Co-Editor of the MPEG Compact Descriptor for Visual Search Standard (ISO/IEC 15938-13) and MPEG Compact Descriptor for Video Analytics standard (ISO/IEC 15938-15). He is currently an Associate Editor for the IEEE TRANSACTIONS ON MULTIMEDIA, *ACM Transactions on Intelligent Systems and Technology* and *ACM transactions on Multimedia Computing, Communications, and Applications*, and the Area Chair of the ACM MM and IEEE ICME. He is a Member of the MSA Technical Committee in IEEE-CAS Society. He was the recipient of the IEEE ICME best paper awards in 2020 and 2019, the IEEE VCIP best paper award in 2019, EURASIP *Journal on Image and Video Processing* Best Paper Award in 2015, the Ministry of Education Technology Invention Award (First Prize) in 2016, the National Technology Invention Award (Second Prize) in 2017, the China Patent Award for Excellence in 2017, and the National Information Technology Standardization Technical Committee Standardization Work Outstanding Person Award in 2015.



Jiaying Liu (Senior Member, IEEE) received the Ph.D. (Hons.) degree in computer science from Peking University, Beijing, China, in 2010. She is currently an Associate Professor and a Peking University Boya Young Fellow with the Wangxuan Institute of Computer Technology, Peking University. From 2007 to 2008, she was a Visiting Scholar with the University of Southern California, Los Angeles, CA, USA. She was a Visiting Researcher with the Microsoft Research Asia in 2015 supported by the Star Track Young Faculties Award. She has authored more than 100 technical articles in refereed journals and proceedings, and holds 50 granted patents. Her current research interests include multimedia signal processing, compression, and computer vision. She is a Senior Member of CSIG and CCF. She was a member of Multimedia Systems & Applications Technical Committee, and Visual Signal Processing and Communications Technical Committee in IEEE Circuits and Systems Society. She was also the Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and *Elsevier Journal of Visual Communication and Image Representation*, the Technical Program Chair of IEEE ICME-2021 and ACM ICMR-2021, the Publicity Chair of IEEE ICME-2020 and ICIP-2019, and the Area Chair of CVPR-2021, ECCV-2020, and ICCV-2019. From 2016 to 2017, she was the APSIPA Distinguished Lecturer. She was the recipient of the IEEE ICME-2020 best paper awards and IEEE MMSP-2015 Top10% paper awards.