Taming Reversible Halftoning
via Predictive Luminance

Cheuk-Kit Lau, Menghan Xia, Tien-Tsin Wong

Abstract—Traditional halftoning usually drops colors when dithering images with binary dots, which makes it difficult to recover the original color information. We proposed a novel halftoning technique that converts a color image into a binary halftone with full restorability to its original version. Our novel base halftoning technique consists of two convolutional neural networks (CNNs) to produce the reversible halftone patterns, and a noise incentive block (NIB) to mitigate the flatness degradation issue of CNNs. Furthermore, to tackle the conflicts between the blue-noise quality and restoration accuracy in our novel base method, we proposed a predictor-embedded approach to offload predictable information from the network, which in our case is the luminance information resembling from the halftone pattern. Such an approach allows the network to gain more flexibility to produce halftones with better blue-noise quality without compromising the restoration quality. Detailed studies on the multiple-stage training method and loss weightings have been conducted. We have compared our predictor-embedded method and our novel method regarding spectrum analysis on halftone, halftone accuracy, restoration accuracy, and the data embedding studies. Our entropy evaluation evidences our halftone contains less encoding information than our novel base method. The experiments show our predictor-embedded method gains more flexibility to improve the blue-noise quality of halftones and maintains a comparable restoration quality with a higher tolerance for disturbances.

Index Terms—Reversible halftoning, deep learning, blue-noise.

✦

1 INTRODUCTION

HALFTONING is commonly used in the printing industry [1] to reproduce tone with limited colors, e.g. black and white, due to the cost consideration. The original image's color and fine details are inevitably lost during this process. This makes the originals nearly impossible to be recovered from these degraded halftones. Even the state-of-the-art inverse halftoning methods [2], [3] can only recover an approximate grayscale version since the color is usually dropped before halftoning. Apparently, resolving this dilemma requires a fore-looking halftoning technique that retains the necessary information for restoration. In this paper, we conducted a thorough study to explore this problem.

Traditional halftoning methods distribute halftone dots mainly for tone reproduction. We observe that this target still permits certain perturbation in terms of the desired binary pattern, as evidenced in Fig. 1. It indicates the possibility of utilizing such a degree of freedom for additional usage, i.e., embedding the potentially missing color information and fine details. Formally, this brings out a new concept, i.e., reversible halftoning, which converts a color image to a halftone that possesses restoration ability to the original color version. Inspired by invertible grayscale [4], we adopt the invertible generative model to formulate our problem. However, generating quality halftones is more challenging than decolorization. The challenges lie

Fig. 1. Observation: the halftone variants of (a) (b) (c) present similar visual quality but with different binary patterns, as the overlaid RGB image visualized in (d). It shows the possibility of modulating the patterns for additional usage.

in the flatness degradation of CNNs in halftoning and the difficulty in achieving vivid visual simulation and accurate information embedding with 1-bit pixels. To address flatness degradation, we propose a Noise Incentive Block (NIB) that introduces spatial variation to the feature space while reserving the information intactness. To achieve the binary halftone, we propose a binary gate that takes gradient propagation tricks to allow training with quantization.

Anyhow, as reported in our preliminary study [5], the binary encoding space is limited and causes sacrifice for the blue-noise property against the restoration accuracy. Inspired by the predictive coding concept [6], we promote

- Cheuk-Kit Lau and Tien-Tsin Wong are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. E-mail:{cklau21,ttwong}@cse.cuhk.edu.hk
- Menghan Xia is with Tencent AI Lab. E-mail:menghanxyz@gmail.com

the encoding framework by exploiting the predictive power from the inverse halftone module. The intuition is that most luminance information could be inferred from the halftone, and removing luminance information from the encoding stage offers more capacity for blue-noise realization. The model is trained end-to-end with highly mixed objectives, formulated as three loss terms: halftone loss, restoration loss, and luminance loss. Particularly, we propose a guiding-ware training scheme to circumvent the tricky converging issue of multi-objective optimization.

Extensive evaluation and ablation study demonstrate that the proposed predictive encoding model allows a good balance among the visual simulation, blue-noise profile, and restoration accuracy for reversible halftoning. The trained model achieves very competitive performance against traditional halftoning algorithms in halftoning quality while still maintaining decent restoration accuracy of the original color image.

The preliminary version of this manuscript presented two distinct contributions. Firstly, we introduced a novel method for reversible halftoning that enhances the functionality of existing halftoning applications. This method circumvents the ill-posed inverse halftoning problem at its source. Secondly, we proposed a model-agnostic plug-in, the noise incentive block, which effectively addresses the flatness degradation of CNN.

In this manuscript, our primary focus is to promote the invertible generation framework with a predictive coding concept. Our objective is to demonstrate the efficacy of this framework in reducing the encoding burden and improving the embedded halftone quality.

2 BACKGROUND

2.1 Image Halftoning

Digital halftoning has been widely studied over the past decades. The goal is to render images in only two levels of pixel values, black and white. It creates an illusion of the continuous tone of the original image through the spatial distances between black and white dots. Traditional deterministic approaches include ordered dithering [7], [8], [9], error diffusion [10], [11], [12], dot diffusion [13], and direct binary search [14]. They aim to produce halftone images that preserve the local tone of the original image while with minimal artifacts.

Since humans are perceptually more aware of artifacts in low-frequency areas, an ideal halftone image should contain the blue-noise property. The blue-noise property corresponds to visually pleasing [1] and minimal low-frequency components [15]. There are several works to achieve this, such as using perturbed error diffusion [1], blue-noise mask [15], [16], diffusion parameter set optimization [11], [17], and tile-based methods [18].

Although focusing on blue-noise rendering can produce a smooth and evenly distributed surface, fine details such as edges and complex structures will be blurred. Many proposed works aim to improve the halftone images using edge enhancement [19], [20], [21], [22]. Pang et al. [23] first introduced structural similarity and tonal similarity into the optimization function, followed by Chang et al.

[24] optimized the error diffusion algorithm with structural similarity.

Some neural-network-based approaches [25], [26] aim to produce halftone images in a deterministic manner.

2.2 Inverse Halftoning

In the early printing industry, many images in newspapers, magazines, and books are halftone printings. "Inverse halftone" dedicates to restoring the continuous tone of images from the halftone images. It is an ill-posed problem because the fine details have been lost in the halftoning process. The simplest method is to process the halftone image with a low-pass filter [27], [28], [29]. However, such a method will also remove edge information. Kite et al. [30] proposed a kernel function built from local gradients to preserve high-frequency details. Xiong et al. [31] proposed to extract edge information and discard background noise via wavelet decomposition. Some works reformulate the continuous-tone restoration problem as a projection onto convex sets (POCS) [32], [33]. Ting and Riskin [34] proposed using a look-up table (LUT) to obtain a temporary grayscale image. Mese and Vaidyanathan [35] further proposed restoring the grayscale image using LUT without any linear filtering techniques. Both approaches improve the efficiency of restoring continuous-tone images. Therefore, many dictionary learning-based approaches have been proposed since then [36], [37], [38], [39], [40], [41].

Yue and Chen [42] proposed using Hopfield neural network [43] based optimization model to inverse halftoning. Huang et al. [44] proposed using a radial basis function neural network to restore the continuous tone from the halftone input. However, the quality of inverse halftoning is highly dependent on the starting halftone method.

Recently, deep learning approaches have been explored by authors. Xiao et al. [45] and Gao et al. [46] proposed inverse halftone via U-Net structure with convolution layers. Xia and Wong [2] improved the restoration quality by introducing residual learning layers to predict enhanced details further. Kim and Park [3] proposed a generative adversarial network (GAN) with object categories prediction and edge information extraction. Besides restoring grayscale images, restoring color from halftone images is harder. It is because more information is needed to fill-ins instead of luminance only. Yen et al. [47] restored color images by concatenating the inverse halftone and colorization stages. Such a method requires extra information to hint at the network to predict color from the intermediate grayscale image.

2.3 Reversible Generation

The reversible generation topic has been widely studied in the data hiding field. Major tasks applications include hiding watermarks or copyright declarations in images [48], [49], [50]. Also, authors have explored methods to hide the color information in the grayscale version image. Queiroz and Braun [51] proposed hiding chrominance channels into subbands from wavelet transform. Xu and Chan [52] proposed hiding the chrominance channels specifically in high-frequency areas of the grayscale version via error diffusion techniques.

Recently, CNNs have gained massive success in image processing tasks. By considering the grayscale image as a latent representation of the color image, Xia et al. [4] proposed an encoding-and-decoding framework to generate reversible grayscale images that can be reversed back to their color version. Ye et al. [53] further proposed using the dual features extractions to improve the restoration quality. A similar framework is adopted in other tasks, such as image resampling [54], [55] and image retargeting [56]. Another approach, invertible neural networks (INNs) [57], [58], [59], [60], [61], [62], [63], [64], generates latent representation without loss of information; however, it relies on explicitly structured network architecture. Such constraint generally makes the training tricky and unstable. In our preliminary study [5], we adopted the invertible generation model as [4], and the limited encoding space of binary pattern causes the trade-off between the blue-noise quality and the restoration quality. In this paper, we promote the encoding framework with a predictive coding concept, i.e., removing the luminance information from the encoding stage and inferring it from the halftone pattern, which facilitates making a practically better balance between the visual quality and data embedding accuracy.

3 REVERSIBLE HALFTONING

We aim to learn reversible binary patterns toward halftoning color images, which is required to offer visual pleasantness and embed restoration-necessary information in the meantime. The key idea is to encode the color information into the halftone image and restore the color image by decoding the halftone image. We first adopted the autoencoder design, where the latent feature is represented in the halftone patterns, to approach the problem. However, the halftone patterns have to fulfill certain objectives: 1) the distribution of dots should resemble the continuous tone of its grayscale version perceptually; 2) the distribution of dots should maintain high blue-noise quality; and 3) the color information should be embedded into the distribution of dots. This poses a challenge to the novel autoencoder approach because the latent feature is not just a representation of the embedded information, but for fulfilling all three objectives simultaneously.

3.1 Embedding Framework with Predictive Luminance

Concept of Predictive Coding. The concept of predictive coding had been described in different areas. In neuroscience, "predictive coding" suggests that the brain solves inverse problems via an internal model of the world [65], [66]. It provides an explanation of how our brain receives and reduces redundant signals. Such an idea was also established in the signal-processing domain. The key idea is to compress data with discarded information and restore the data by predicting the discarded information back. Predictable information shall be excluded from the compressed data. The compressed data should only include the residual error between the predicted and the actual values. Such an approach significantly increases the compression ratio. Predictive coding appears in various applications, such as image compression [67], temporal video compression [68] and representation learning [69], [70].

Our problem is similar to the data compression settings, where information is compressed (encoded) and restored (decoded). Our novel autoencoder method suffers the drawback of encoding information into the halftone pattern. Since we train the network to encode and decode information in RGB space, the encoder will encode all information in RGB as it can. However, due to the binary level of pixels and the halftone image having to resemble the continuous tone of its input, the encoding space available for encoding is further limited. In our base method, blue-noise quality has been sacrificed. If we remove some information from the limited encoding space but put it back in the restoration stage, the network should have more freedom to produce halftone patterns while maintaining its restoration ability. On the other hand, we know the work of inverse halftone has been long studied and well-developed. State-of-the-art work [2] can predict the continuous tone from halftone images with fine details. We can offload the luminance information from the encoding-decoding pipeline, thus constraining the network to sample the subspace of chrominance only. In the restoration stage, we extend the network with a predictor module, an inverse halftone module, to restore the offloaded luminance information. In this manuscript, we aim to improve the blue-noise quality of our halftone image through this spared encoding space.

We extend the design established in $\text{Ours}/_{base}$ [5]. Our network consists of three main components(Fig. 2):

- An encoder that encodes color information into the generated halftone image;
- A predictor that predicts the luminance channel from the encoded halftone image;
- A decoder that restores the chrominance channels from the encoded halftone image.

Given an RGB image I_c, we construct a reversible halftone image O_h by the encoder E and the binary gate B:

$$\tilde{O}_h = \text{E}(\text{ Nib}(I_c)) \qquad (1)$$

$$O_h = \text{B}(\tilde{O}_h) \qquad (2)$$

The details of the noise incentive block $\text{Nib}(\cdot)$ is discussed in section 3.2.1. The encoder generates a *pseudo halftone image* $\tilde{O}_h$, in which each pixel are real value ranging from 0 to 1. The binary gate quantizes the pixels in the pseudo halftone image from real value to either 0 or 1. Then we feed O_h from (2) into two networks, a decoder network D to restore the chrominance channels O_c^{ch}, and a predictor network P to predict the luminance channel O_c^l.

$$O_c^{ch} = \text{D}(O_h) \qquad (3)$$

$$O_c^l = \text{P}(O_h) \qquad (4)$$

Finally, we obtain the restored color image O_c by concatenating those three channels and convert to RGB color space. Our color space conversion function follows the standard specified in [71].

3.2 Network Architecture

We adopt the U-shaped architecture for both the encoder and decoder networks. Both networks share a similar structure, containing three downscale blocks, three upscale
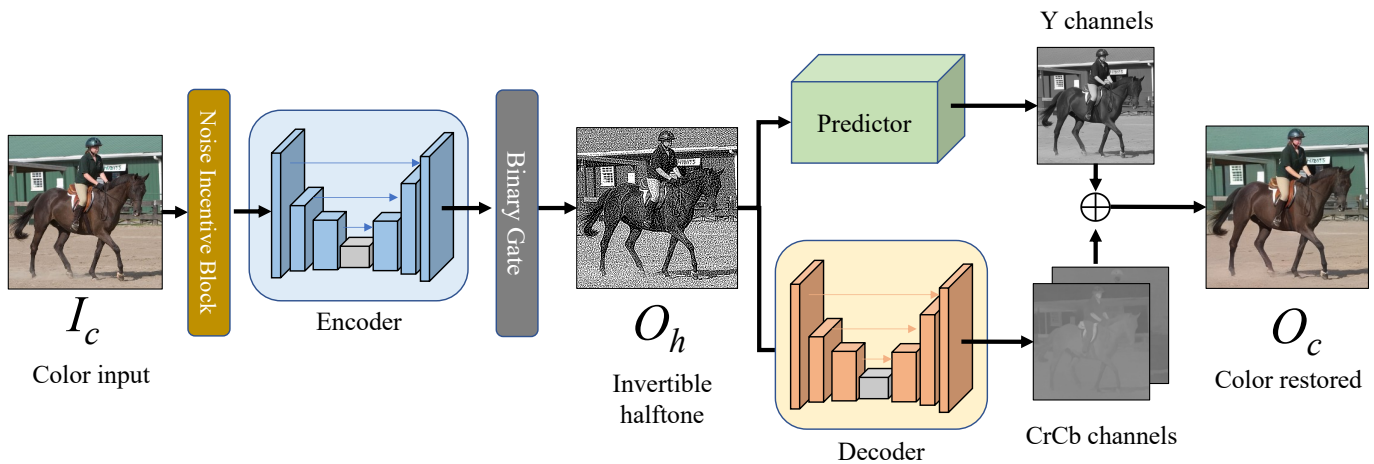
Fig. 2. Overview of our network architecture with embedded luminance predictor. ⊕ denotes the concatenation operation.

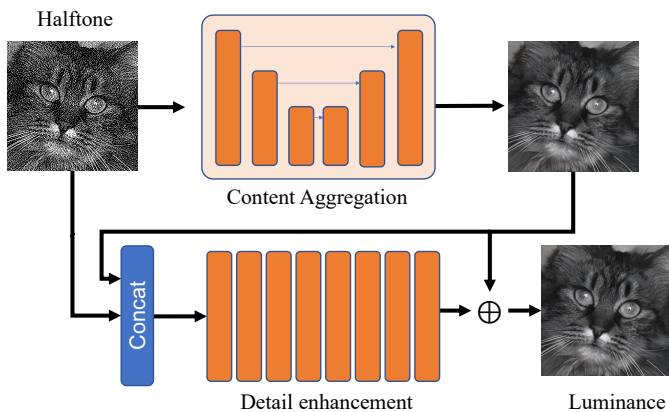Fig. 3. Overview of the predictor architecture [2]. ⊕ denotes the addition operation.

blocks, four residual blocks, and two convolution blocks. We adopt U-Net as the network backbone because of its enlarged receptive field, and other qualified CNN architectures may also work. We adopted the [2] model as our predictor module. Any other inverse halftone module may also work. Additionally, we propose two special designs within this network: the noise incentive block to mitigate the flatness degradation introduced by CNN; and the binary gate to encourage the network to generate near-binary pixels. The base architecture, which does not include the predictor module, is denoted as Ours/$_{base}$ in this manuscript.

3.2.1 Noise Incentive Block

We uncovered a phenomenon that we refer to as "flatness degradation," which arises from the convolutional paradigm with spatially shared kernels when presented with flat inputs. This phenomenon leads to a scaling operation that applies the same parameters across the input and produces a constant signal, thereby impeding the ability of CNNs to dither a constant grayness. This, in turn, hinders the formulation of the blue-noise profile, which is primarily measured over the constant grayness. To address this

issue, we propose the Noise Incentive Block (NIB), which introduces spatial variation to the feature representation while preserving the original input. By preprocessing the color image before passing it to the encoder, our dithering network is able to generate binary halftone in flat regions. The NIB also enables us to formulate the blue-noise profile through low-frequency constraints on dithered constant-grayness. The example of NIB-equipped results is located in Fig. 16 in the supplementary.

3.2.2 Binary Gate

Another special design for the dithering network is the binary gate $\mathrm{B}(\cdot)$ that quantizes the network output $\tilde{O}_h$ to be a strict binary image $O_h = \mathrm{B}(\tilde{O}_h)$. We explicitly adopt a binary gate because the soft non-binary penalty is insensitive to tiny deviations, i.e., near-0 or near-1 valued pixels, which is vulnerable to quantization when stored as a 1-bit bitmap and thus hurts the restoration accuracy. However, one obstacle should be noted: the binary gate is non-differentiable. To enable the joint training, we use Straight-Through Estimator [72] on the binarization when calculating the gradients.

3.2.3 Predictor

Fig. 3 shows an overview of the predictor module. We notice that halftone patterns inherently convey luminance and structural information, regardless of whether they are encoded with color information or not. As a result, we have implemented Xia's [2] inverse halftone module to predict continuous luminance information, thereby allowing us to concentrate on encoding chrominance information. The predictor consists of two key components: the content aggregation block, which incorporates three downscale blocks, three upscale blocks, and four residual blocks; and the detail enhancement block, which employs eight residual blocks to improve the predicted luminance details.

3.3 Loss Function

We trained our network with the following loss functions: the halftone loss $\mathcal{L}_{half}$; the restoration loss $\mathcal{L}_{restore}$; and the

luminance loss $\mathcal{L}_{lumin}$. We trained our network in multiple stages; the detailed combination of loss functions and their corresponding coefficients are discussed in Section 3.4.

3.3.1 Halftoning

We adopted the halftone loss $\mathcal{L}_{half}$ to train the network to generate the desired reversible halftone image. Our halftone loss is formulated as:

$$\mathcal{L}_{half} = \alpha \cdot \mathcal{L}_{bin} + \beta \cdot \mathcal{L}_{tone} + \gamma \cdot \mathcal{L}_{blue} \qquad (5)$$

where $\mathcal{L}_{bin}$ denotes the binary loss; $\mathcal{L}_{tone}$ denotes the tone loss; and $\mathcal{L}_{blue}$ denotes the blue-noise loss.

Let $\tilde{O}_h$ be the *pseudo halftone image* generated by the encoder E but before the quantization layer Q. Since quantization on $\tilde{O}_h$ cannot be differentiated, binarization loss takes a crucial role in encouraging the network to produce binary intensity values on the halftone image. It is formulated as

$$\mathcal{L}_{bin} = ||\mathrm{B}(\tilde{O}_h) - \tilde{O}_h||_1 \qquad (6)$$

where $|| \cdot ||_1$ denotes the L_1 norm. $\mathrm{B}(\cdot)$ denotes the binary gate.

Based on the tone similarity concept, which was proposed by [23], we applied tone loss $\mathcal{L}_{tone}$ to encourage the halftone image O_h to resemble the tone of the input image. It is formulated as follows:

$$\mathcal{L}_{tone} = ||G(I_{gray}) - G(O_h)||_2 \qquad (7)$$

where $G(\cdot)$ denotes a Gaussian filter with kernel size 11×11 and sigma 2.0; I_{gray} denotes the grayscale version of color input I_c; $|| \cdot ||_2$ denotes the L_2 norm.

To train the network to produce halftone with blue-noise property, we adopted the blue-noise loss $\mathcal{L}_{blue}$ suggested by [5]. Its basic idea is to restrict the network to generate minimal low-frequency components because they are more noticeable to the human eye. Therefore, we prepared a set of plain-color images $\mathcal{P}$. For each training iteration, after the color image from the dataset has been passed to the network, we randomly draw a plain color image $p \in \mathcal{P}$. A halftone image z_p is obtained by passing p into the network. The blue-noise loss is formulated as

$$\mathcal{L}_{blue} = ||[DCT(z_p) - DCT(p_{gray})] \odot M||_2 \qquad (8)$$

where p_{gray} denotes the grayscale version of p. $DCT(\cdot)$ denotes the discrete cosine transformation function. M denotes the binary mask. We set M to only allow the first 3.8% of low-frequency DCT coefficients to pass through.

Compared to our preliminary version [5] of this manuscript, we dropped the structure loss suggested by [23] since it has no significant effect on our training outcome.

3.3.2 Restoration

We constructed the restoration loss as

$$\mathcal{L}_{restore} = \zeta \cdot \mathcal{L}_{chromin} + \eta \cdot \mathcal{L}_{percep} \qquad (9)$$

where $\mathcal{L}_{chromin}$ denotes the chrominance loss; and $\mathcal{L}_{percep}$ denotes the perceptual loss. The chrominance loss trains the decoder to extract chrominance information from the encoded halftone image. Given a restored chrominance channels O_c^{ch}, the chrominance loss are formulated as

$$\mathcal{L}_{chromin} = ||I_c^{ch} - O_c^{ch}||_2 \qquad (10)$$

The perceptual loss trains the network to resemble color signals at the perceptual level. We adopted the perceptual loss $\mathcal{L}_{percep}$ suggested by [5], which is formulated as

$$\mathcal{L}_{percep} = ||\Psi(I_c) - \Psi(O_c)||_2 \qquad (11)$$

where we denote $\Psi(\cdot)$ as the latent feature extracted from the *conv4_4* layer of the pre-trained VGG-19 module [73].

The luminance loss trains the predictor to generate a continuous luminance channel from the halftone image. Since we adopted the inverse halftone module from [2]. We take the full loss function of [2] as our luminance loss, $\mathcal{L}_{lumin}$.

$$\mathcal{L}_{content} = ||\hat{O}_c^l - I_{gray}||_2 \qquad (12)$$

$$\mathcal{L}_{full} = w_a||\Psi(O_c^l) - \Psi(I_{gray})||_2 + ||O_c^l - I_{gray}||_1 \qquad (13)$$

$$\mathcal{L}_{lumin} = \mathcal{L}_{content} + w_b\mathcal{L}_{full} \qquad (14)$$

where $\hat{O}_c^l$ denotes the initial predicted grayscale image from the content aggregation module in [2]. We set the coefficients to the default value stated in [2], where $w_a = 2.0 \times 10^{-6}$, $w_b = 1.5$.

3.4 Training Strategy

Training the whole model from scratch is vulnerable to a local minimum because of the challenging optimization target. To circumvent this problem, we propose to adopt a warm-up training scheme. In the first stage, we aim to warm up the dithering network alone, so that it can generate visually pleasant halftone images. To stabilize the training, the binary gate is temporally removed. Unfortunately, this relaxation still fails to guarantee satisfactory halftones in Fig. 4(b), and it is even associated with slow convergence, as shown in Fig. 20(green curve) in the supplementary. To boost the training, we propose explicitly providing a reference halftone image I_h to guide the training. For simplicity, the classical error diffusion [11] is employed as the reference. However, directly measuring the pixel-wise difference between the predicted halftone and the reference does not work, since per-pixel inspection can never capture the intrinsic feature of binary halftone patterns.

Halftone Pattern Measurement. Inspired by perceptual loss [74], we propose to measure the halftone pattern difference in the continuous feature domain. We pretrained an inverse halftoning network F, a U-shaped architecture with three downscale blocks, four residual blocks, and three upscale blocks, to capture the halftone patterns in the continuous feature domain. Accordingly, we formulate the guidance loss $\mathcal{L}_G$ as

$$\mathcal{L}_G = ||\mathrm{F}(O_h) - \mathrm{F}(I_h)||_2 \qquad (15)$$

Then, we perform the warm-up training on the dithering network with the combined loss:

$$\mathcal{L}_{stage1} = \mathcal{L}_{half} + \mathcal{L}_G \qquad (16)$$

where we set $\alpha = 0.1$, $\beta = 0.6$, $\gamma = 0.3$. The red curve in Fig. 20 shows the high training efficiency. With only 28
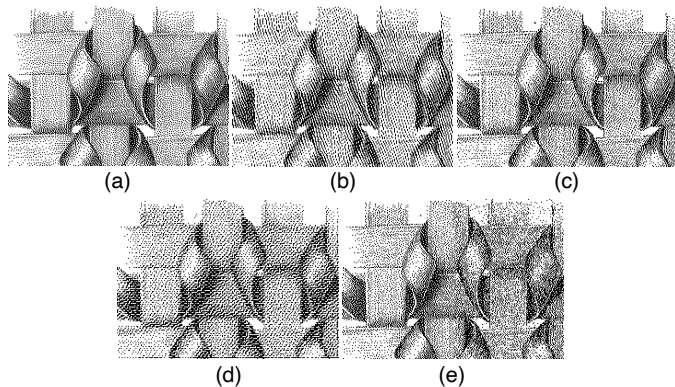
Fig. 4. Halftone generated by models trained with and without guidance loss in different stages. (a) Error diffusion; (b) warm-up training for 130 epochs w/o guidance loss; (c) warm-up training for 28 epochs with guidance loss; (d) our stage two w/o guidance loss; (e) our stage two with guidance loss.

epochs, it is able to generate decent visual results, as shown in Fig. 4(c).

In the second stage, we froze the predictor module; and trained the encoder and decoder networks to learn the desired halftone pattern. The whole model was trained under the following combination of loss functions until the loss converged

$$\mathcal{L}_{stage2} = \mathcal{L}_{half} + \mathcal{L}_{restore} + \epsilon \cdot \mathcal{L}_G \qquad (17)$$

By isolating the predictor module in training, we ensure that the learning of the encoder does not involve luminance information; only chrominance information is encoded into the halftone. we set $\alpha = 0.4$, $\beta = 0.6$, $\gamma = 0.9$, $\epsilon = 0.3$, $\zeta = 1$ and $\eta = 0.00002$ empirically. It is worth noting that we set γ as 0.9 instead of 0.3 as Ours/$_{base}$. The detailed analysis and reasoning are discussed in Section 4.5. We still have to use guidance loss $\mathcal{L}_G$ here; as we experimented, if we dropped this loss, the halftone loses its structures and becomes over-smoothed. Fig. 4(d) shows an example of training without guidance loss in stage two.

At the final stage, we fine-tuned the predictor module. We adopted the inverse halftone module from [2] as our predictor, and it was trained by the loss function specified in (14) only. The encoder and decoder are frozen in this stage.

The approach that separates stage two and stage three ensures that the encoder only encodes chrominance information into O_h. The decoder only outputs two channels, compared to Ours/$_{base}$, which is three. Hence, the restoration burden on luminance has been shifted to the predictor. Such modification allows the encoder to generate O_h with better blue-noise quality instead of sacrificing it. Therefore, our proposed method generates halftones with better tone resemblance and blue-noise quality, while maintaining its restoration quality compared to our base design Ours/$_{base}$.

4 EXPERIMENTAL RESULTS

We trained the warm-up stage with 28 epochs until the model generated decent visual results, then we trained the second stage and final stage until both corresponding losses converged. Each stage takes 87 epochs and 50 epochs respectively. The whole training takes a total of 165 epochs

TABLE 1
Quantitative evaluation on halftone images in terms of the mean PSNR and SSIM values. Higher PSNR/SSIM indicate better quality.

Methods	PSNR	SSIM
Ostromoukhov method	41.728	0.1007
Structure-aware halftoning	21.803	0.0340
Ours	34.444	0.1094

to complete. It is obvious that our predictor embedded method contains more parameters on the restoration side than our base method Ours/$_{base}$. Therefore, to further justify the effectiveness of the predictor module, we compare our method with different variations of Ours/$_{base}$ in this section. The color error maps, in this paper and the supplementary, are generated by normalizing the pixel from [0,255] to [0,1] and computing the L1 distance between the images in the RGB color space.

4.1 Dataset

We evaluated our method on the VOC2012 dataset [75]. It contains 17,125 color images. We cropped and resized all images into 256×256. We randomly split the image set into: 13,758 images as training set; and 3,367 images as validation set.

4.2 Comparison with traditional halftoning

Following the practice in [23], the tone consistency is measured by PSNR between the Gaussian-filtered halftone and the Gaussian-filtered luminance channel of the input, and the structure consistency is measured by SSIM between the halftone and the luminance channel of the input. We experimented with 3,367 grayscale images (decolorized from our validation set), as existing halftoning methods can only dither grayscale images. Two classical halftoning methods that generate high-quality halftones are selected as our competitors, Ostromoukhov's method [11] and the structure-aware halftoning method [23]. In our experiment, the structure-aware halftoning method is used with default parameters for quantitative evaluation while the case-by-case tuned result is provided for visual comparison. The statistics are tabulated in Table 1. Among all, our method achieves the best comprehensive performance of tone similarity (PSRN) and structure similarity (SSIM). Fig. 5 shows examples on a gray ramp and images with structures. Our halftone resembles the continuous tone but is not as smooth as the traditional methods. It is because we traded off the blue-noise quality for encoded color information. However, our halftone visual quality is comparable with traditional methods in images with structures. Our method achieves better structure than the error-diffusion method [11], and less rigid patterns compared to the structure-aware method [23]. We further compared our method with some state-of-the-art halftoning methods [17], [76]. Fig. 6 shows that our method produce less "worm effects" than [11], [23] but still produce checkboard patterns compared to those improved methods [17], [76].

To analyze the blue-noise quality of our halftone images, we adopted the common analysis methods as in [1]. We
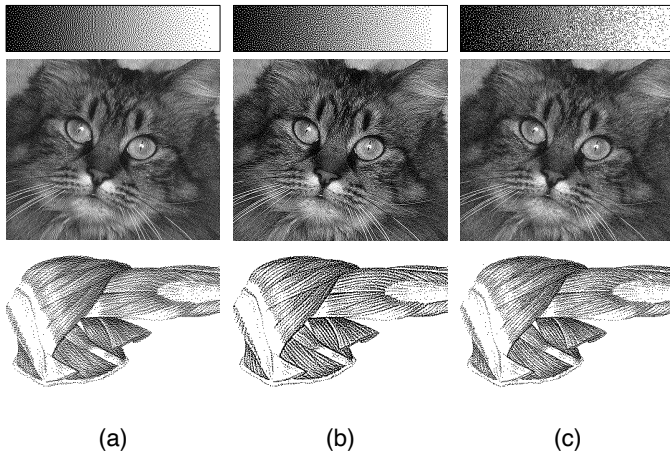
Fig. 5. Qualitative comparison of halftone images with intensive structures. (a) Ostromoukhov; (b) Structure-aware; and (c) Ours.
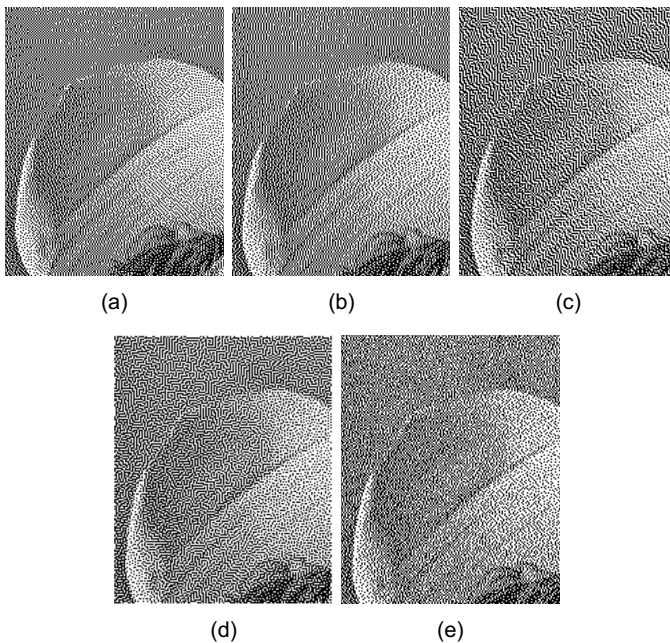

Fig. 6. Qualitative comparison of halftone patterns on "Lena". The example of TDED$_{BS}$ is directly from [76]. (a) Floyd-Steinberg; (b) Ostromoukhov; (c) Zhou and Fang [17]; (d) TDED$_{BS}$ [76]; and (e) Ours.
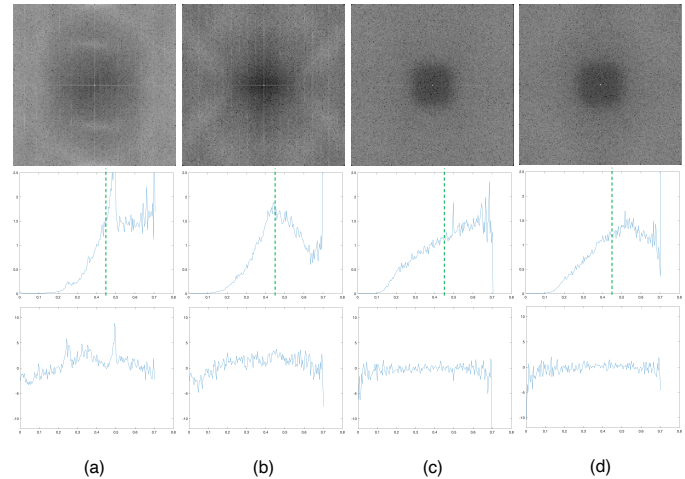

Fig. 7. Spectrum analysis on various halftone results from constant-grayness 0.8. From top to bottom: the power spectrum, radially averaged power spectrum density and anisotropy. The green dashed line indicates the principle frequency. (a) Floyd-Steinberg; (b) Ostromoukhov; (c) Ours/base; and (d) Ours

selected the classical error diffusion methods [10], [11] as our competitors. We analyzed halftone images obtained from constant-grayness images in terms of its *power spectrum* and its *radially averaged power spectrum*. The grayness is set to 0.8. The power spectrum indicates the frequency amplitude in 2-D. Since the amplitude of frequency in halftone is supposed to be radially symmetry. The radially averaged power spectrum visualizes the 2-D power spectrum in 1-D space. According to [1], for a good halftone with blue-noise property, the radial frequency graph should have 1) low amplitude in low-frequency areas; 2) a peak transition region on principle frequency; 3) a flat high-frequency region. We adopted the principle frequency defined in [77]. Fig. 7 illustrates the power spectrum and radial averaged power spectrum of the converted halftone image. Our method produces low amplitude in low-frequency regions

similar to [10], [11]. Also, we observed our peak is closer to the principle frequency, and the shape of the curve resembles the shape in the classical method [11]. The frequency analysis among different gray levels is located in Fig. 15 in the supplementary.

4.3 Evaluation on Reversible Halftoning

Blue-noise quality We evaluate the blue-noise quality on the halftone generated by our method, which includes the predictor module, and our base method Ours/$_{base}$ over color input images. Fig. 8 (top-row) shows halftone examples produced by Ours/$_{base}$ and our method. Both methods preserve the structural details. Our halftone patterns produce smoother surfaces and less "grid-like" structures in low-variance areas. This indicates a better blue-noise property on our halftones. The improvement of blue-noise quality is much more evident on Fig. 9(a). Our halftone dissolves the "grid-like" patterns and is visually smoother than Ours/$_{base}$ with comparable restoration quality. More examples of the color ramp are provided in Fig. 21 in the supplementary.

By observing the spectrum analysis results on Ours/$_{base}$ and our method in Fig. 7, we can see that our halftone resembles the transition peak closer to [11] than Ours/$_{base}$ in power spectrum. Although the peak region is not as wide as [11]; and shifted right from the principle frequency, it approaches the principle frequency closer than our base methods. Hence, our method with a predictor module extends the model's ability to produce halftone images with better blue-noise quality.

Restoration quality We compare our method in grayscale and color image inputs. We take two state-of-the-art methods as our competitors: the PRL-Net [2] as our baseline grayscale candidates and the ColTran [78] as baseline color candidates. The PRL-Net [2] generates grayscale from the error-diffused halftone, and ColTran [78] colorize the grayscale from [2] to obtain the color version. Since PRL-Net can only restore grayscale images, we prepared 3,367

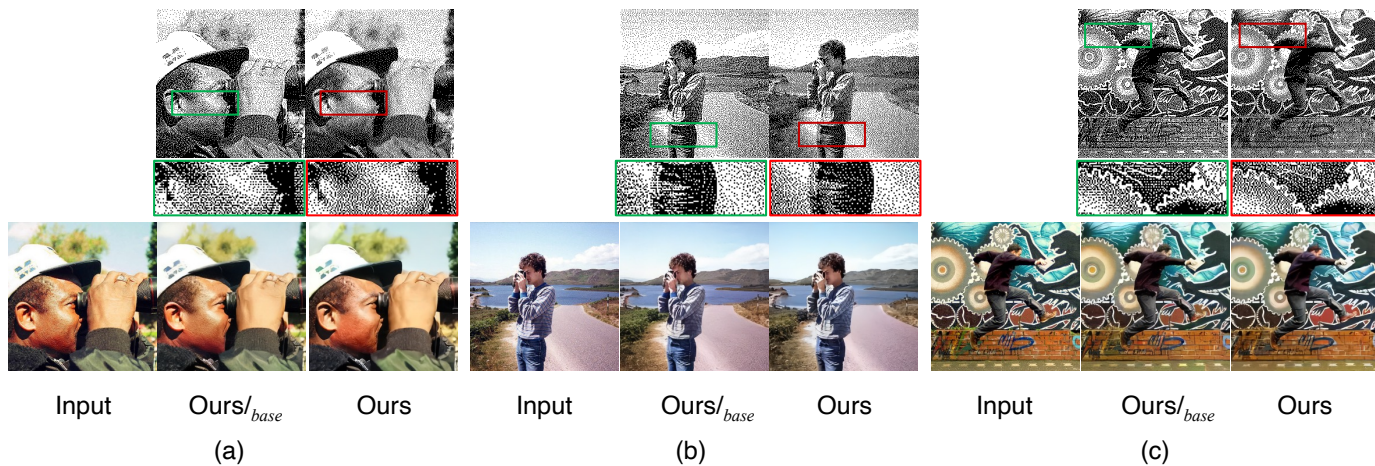| Input	Ours/$_{base}$	Ours	Input	Ours/$_{base}$	Ours	Input	Ours/$_{base}$	Ours
	(a)			(b)			(c)	

Fig. 8. Qualitative comparison on halftone image and restored image together.

TABLE 2
Quantitative evaluation on halftone and restored color images in terms of the mean PSNR and SSIM values. Higher PSNR/SSIM indicate better quality.

Methods	Halftone		Restoration	
	PSNR	SSIM	PSNR	SSIM
PRL-net [2] + ColTran [78]	41.728	0.1007	19.543	0.4612
Ours/$_{base}$	30.624	0.1440	28.130	0.8592
Ours	32.283	0.1136	27.292	0.6060

Input	Ours/$_{base}$	Ours	Input	Ours/$_{base}$	Ours
	(a)			(b)	

Fig. 9. Qualitative comparison on halftone image and restored image together on color ramp. (a) Y=0.8; and (b) Y=0.5.

grayscale images (decolorized from our testing dataset) for grayscale comparison. Table 2 presents the statistics of both PSNR and SSIM. Our superiority lies in the restoration of the color domain. Our method avoids the ill-posed problem of color choice in areas and improves color segmentation with encoded color information. ColTran [78] experiences the drop in PSNR due to differences in color choice from the ground truth. Fig. 10 shows an example of ColTran [78] failing to segment color section properly. Our method is able to produce the pink and green color at corresponding areas which ColTran [78] failed. In fact, the ability of ColTran [78] to guess color lies in its training batches while our method retrieves the color information from the halftone patterns.

Furthermore, we compare our method with Ours/$_{base}$ to evaluate the effectiveness of the predictor module. The example in Fig. 8 (bottom-row) demonstrates our improved encoded halftone with comparable restoration ability with Ours/$_{base}$. By adopting the predictor module, we achieve the same level of restoration quality while improving the blue-noise quality in halftone. It is because, as our halftone becomes smoother with less encoded information, the predictor module fills in the missing luminance information by "guessing". Therefore, our restoration power maintains a comparable level with Ours/$_{base}$ when the "guess" is correct. We notice the restoration artifacts in extreme dark luminance, such as Y=1 in Fig. 21. We believe it is caused by the inverse halftone module being trained on images with structural complexity, rather than plain colors. Nonetheless, our restoration quality is comparable on average and applicable in real-world cases.
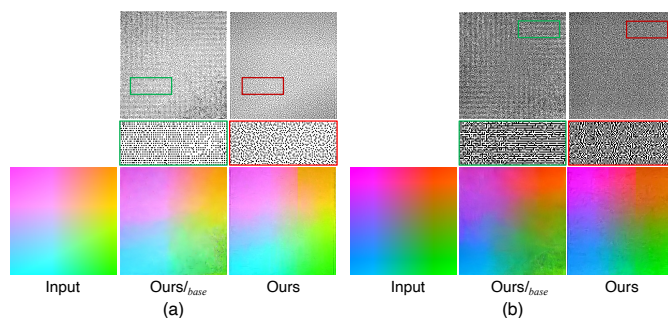

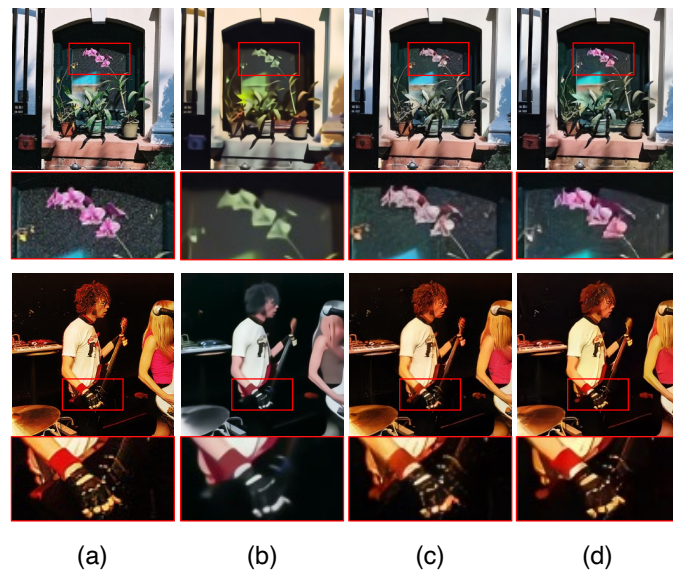| (a)	(b)	(c)	(d)

Fig. 10. Colorization vs. our methods. (a) Input; (b) ColTran [78]; (c) Ours/$_{base}$; and (d) Ours

4.4 Data embedding study

We adopt the concept of entropy in information theory to estimate encoded information in our halftone patterns. The information of a source produced can be measured in

TABLE 3
Entropy estimations via lossless universal zip compression.

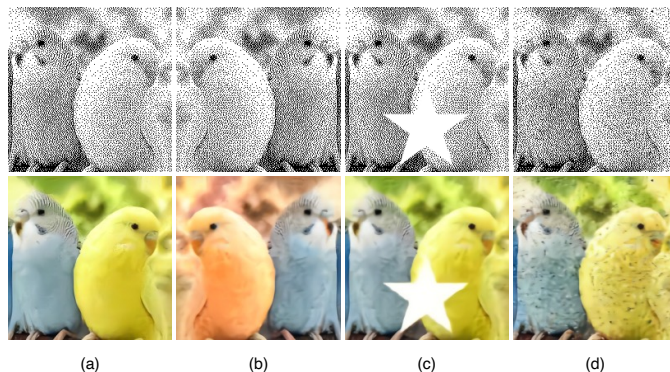| Methods	Compression rate
Ostromoukhov's [11]	87.0%
Ours/$_{base}$	86.2%
Ours/$_{base}^{\gamma+}$	86.3%
Ours/$_{base}^{L}$	86.6%
Ours/$_{base}^{L\gamma+}$	86.6%
Ours	86.8%

Fig. 11. Robustness study of reversible halftones. The color image (bottom row) is restored from the reversible halftones (top row). (a) No operation; (b) flipped; (c) partial masked; and (d) random noise with 10% impulse noise, which is more destructive than Gaussian noise.
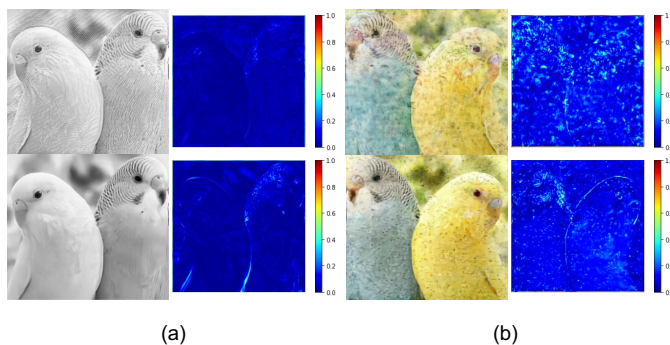

Fig. 12. Robustness comparison between flipping and random noise. The top is Ours/$_{base}$ and the bottom is Ours. (a) Flipped (grayscale); and (b) random noise.

TABLE 4
Ablation study on various training methods.

Methods	Halftone		Restoration	
	PSNR	SSIM	PSNR	SSIM
Ours/$_{base}$	30.625	0.1459	27.822	0.6462
Ours/$_{p-froze}$	32.360	0.1148	24.115	0.4835
Ours/$_{end-to-end}$	30.854	0.1467	28.995	0.6850
Ours	32.680	0.1152	27.686	0.6565

terms of entropy [79]. We claim that our method encoded less information than Ours/$_{base}$. One way to evaluate the estimated entropy is to compare signals via a lossless compression. In lossless compression, redundant signals are replaced by shorter code words [80]. Therefore, sources with less information, hence less "surprise" in signals, should obtain a higher compression rate [79], [81].

Since compressing images with different spatial arrangements yield variances in compression rates, We rotated and flipped all 3,367 images in all four directions before evaluating their compression rates. By expanding the image set in this way, we could ensure that the compression rates we compared were representative of each image's general compression characteristics. Finally, we compressed all the halftone images into a single ZIP archive using the universal zip library [82] for comparison purposes. Table 3 shows the respective compression rates. The classical error-diffusion method obtains the highest compression ratio, while all variations of our base methods obtain lower compression rates. Our predictor-embedded method sits between the error-diffusion method and our base methods. This experiment further proves our claim.

Furthermore, we analyze how the encoding information is embedded and its robustness by applying several typical disturbances to the generated halftones, including flipping, partial removal, and random impulse noises. Fig. 11 illustrates the restored color examples from augmented halftones. Fig. 11(c) shows when under a regional mask, the color in the unmasked regions is restored similarly to the original. This indicates our color data are encoded local-wise instead of global-wise. However, the restored structure is also blurred. We believe this is brought by the prediction accuracy of the predictor. The incorrect color restored from the flipped halftone in Fig. 11(b) indicated the encoded information is directionally sensitive. Although both Ours/$_{base}$ and Ours cannot restore a correct color from the flipped halftones, our restored version contains fewer structural diagonal artifacts. Fig. 12(a) shows a comparison of the grayscale version of the restored image. Fig. 12(b) further shows that our method increases the tolerance to noise against our base method, which indicates the good potential to be used in real-world applications. Since most of the structural information is constructed from the luminance and we offload such work to the predictor, the encoded information only affects the color correctness. Therefore, our restored color in flipped halftones contains fewer structural artifacts than Ours/$_{base}$. Our method also shows higher tolerance to random noises than Ours/$_{base}$. Ground-truth

images and detail comparisons are located in Fig. 22 in the supplementary.

4.5 Ablation Study

In this section, we demonstrate the necessity of the fine-tuning stage of the predictor. In the training iteration, we froze the predictor so it does not participate in the gradient backward propagation. Ours/$_{p-froze}$ in Table 4 shows a huge drop in restoration accuracy. Fig. 19 in the supplementary shows an example of this case. Since we adopted the pretrained model from [2], which trained on classic error-diffusion halftone images, as our predictor, the pretrained model cannot recognize our halftone pattern and treat them as noise (Fig. 19(c)). Therefore, the predicted luminance contains artifacts.

TABLE 5
Ablation study on variation of $\mathcal{L}_{blue}$'s coefficient γ.

Methods	γ	Halftone		Restoration	
		PSNR	SSIM	PSNR	SSIM
Ours/$_{base}$	0.3	30.678	0.1426	28.234	0.6408
	0.9	30.124	0.1838	27.666	0.6524
Ours/$_{base}^{L}$	0.3	29.927	0.2422	28.276	0.6331
	0.9	28.744	0.1536	21.353	0.4489
Ours	0.3	36.194	0.1081	29.362	0.7025
	0.9	32.282	0.1133	27.451	0.6036

We studied the importance of the isolation training strategy. We released the predictor and trained all three modules end-to-end in the second stage, skipping stage three. The result is denoted as Ours/$_{end-to-end}$ in Table 4. We can see that there is an improvement in color restoration. However, the halftone accuracy remains the same level as Ours/$_{base}$. It is because, in the backward propagation stage, both the predictor and decoder act as the learning factors for the encoder. With the increased parameters on the restoration side, an improvement in color restoration is expected. Therefore, isolating the predictor when training the encoder becomes a necessary step. This indicates the significance of our two-stage strategy.

Finally, we compare the effectiveness of blue-noise manipulation between our base and predictor-embedded methods. For a fair comparison, we trained our base method with doubled layers in the decoder module, denoted as Ours/$_{base}^{L}$, to match the parameter size with the predictor-embedded approach. It is worth noting that we propose setting the coefficient $\gamma = 0.3$ in our base method because we found that $\mathcal{L}_{blue}$ and $\mathcal{L}_{restore}$ are conflicting each other. However, with the predictor-embedded approach, we can push the value γ to 0.9. Table 5 shows the detailed comparison between the variations of γ in training. We can see that if we increase the weight of blue-noise loss, Ours/$_{base}$ results in the same level of quality regarding halftone accuracy and restoration accuracy. Fig. 13 shows the spectrum analysis with models trained with increased γ. Even with a larger parameter size in the decoder and higher blue-noise loss weight on Ours/$_{base}$, the anisotropy was suppressed, but the model still struggles to produce a transition peak around the principle frequency in the power spectrum. We can see that both Fig. 13(a) and Fig. 13(b) produce high intensity in the high-frequency area. It is because without changing the encoding content and the suppression of low frequency introduced by the blue-noise loss $\mathcal{L}_{blue}$, the model is forced to encode information into the high-frequency area. Ours/$_{base}$ reaches its limits to improve the blue-noise property.

With the predictor approach, our model quickly raises the bar of halftone tone consistency and restoration accuracy with the default weighting of 0.3. It is because, with lesser information that needs to be encoded, the model tends to improve the tone of halftone patterns when we set the default $\mathcal{L}_{tone}$'s coefficient $\beta = 0.6$. Since we aim to improve the blue-noise quality in our halftone images,
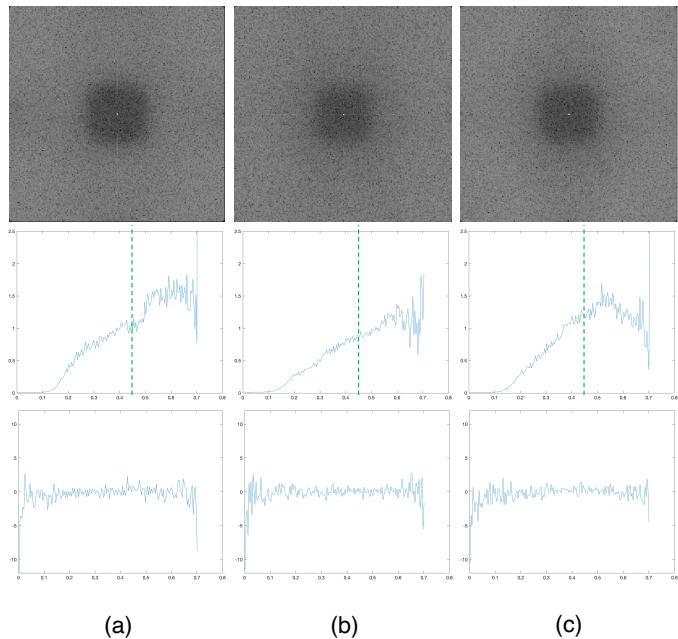

Fig. 13. Spectrum analysis between Ours/$_{base}$ and Ours with increased blue-noise loss weight. (a) Ours/$_{base}$, $\gamma = 0.9$; (b) Ours/$_{base}^{L}$, $\gamma = 0.9$; and (c) Ours.
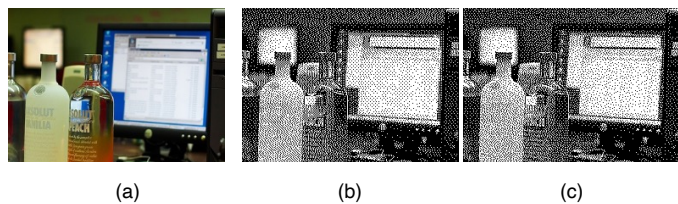

Fig. 14. Example of halftone generated via different tone and blue-noise loss weights. (a) Input (b) $\beta = 0.6, \gamma = 0.3$; and (c) $\beta = 0.6, \gamma = 0.9$

we take $\gamma = 0.9$ as our final proposed version. Fig. 14 shows an example of halftone images between heavier tone consistency vs. heavier blue-noise weights.

The ablation study of the NIB block can be found in the supplementary.

5 CONCLUSION

We propose a novel reversible halftoning technique with high restoration ability and state-of-the-art visual quality. Our approach is a strong alternative to traditional halftoning methods and eliminates the need to tackle the ill-posed inverse halftoning problem. To extend the ability of the reversible model, we introduce a predictive module that offloads the encoding burden between the blue-noise property and the hidden color information. Our formulation of the blue-noise loss as a low-frequency constraint on constant-grayness guarantees the visual pleasantness of halftone patterns. We also propose a method to modulate the priorities of different loss terms in three stages to handle the tricky optimization landscape. Our experiments demonstrate the advantages of our approach and highlight the improvement achieved by the predictor strategy. We believe our contributions to reversible halftoning and the predictor approach will inspire future work in this field.

REFERENCES

[1] R. A. Ulichney, "Dithering with blue noise," *Proceedings of the IEEE*, vol. 76, no. 1, pp. 56–79, 1988.

[2] M. Xia and T.-T. Wong, "Deep inverse halftoning via progressively residual learning," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 523–539.

[3] T.-H. Kim and S. I. Park, "Deep context-aware descreening and rescreening of halftone images," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–12, 2018.

[4] M. Xia, X. Liu, and T.-T. Wong, "Invertible grayscale," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, pp. 1–10, 2018.

[5] M. Xia, W. Hu, X. Liu, and T.-T. Wong, "Deep halftoning with reversible binary pattern," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 000–14 009.

[6] A. Said and W. A. Pearlman, "Reversible image compression via multiresolution representation and predictive coding," in *Visual Communications and Image Processing '93, Cambridge, MA, USA, 7-12 November 1993*, vol. 2094, 1993.

[7] J. Limb, "Design of dither waveforms for quantized visual signals," *The Bell System Technical Journal*, vol. 48, no. 7, pp. 2555–2582, 1969.

[8] B. Lippel and M. Kurland, "The effect of dither on luminance quantization of pictures," *IEEE Transactions on Communication Technology*, vol. 19, no. 6, pp. 879–888, 1971.

[9] B. E. Bayer, "An optimum method for two-level rendition of continuous tone pictures," in *IEEE International Conference on Communications, June, 1973*, vol. 26, 1973.

[10] R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial gray-scale," in *Proceedings of the Society of Information Display*, vol. 17, 1976, pp. 75–77.

[11] V. Ostromoukhov, "A simple and efficient error-diffusion algorithm," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, pp. 567–572.

[12] V. Ostromoukhov, C. Donohue, and P.-M. Jodoin, "Fast hierarchical importance sampling with blue noise properties," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 488–495, 2004.

[13] D. E. Knuth, "Digital halftones by dot diffusion," *ACM Transactions on Graphics (TOG)*, vol. 6, no. 4, pp. 245–273, 1987.

[14] K. Chandu, M. Stanich, C. W. Wu, and B. Trager, "Direct binary search (dbs) algorithm with constraints," in *Color Imaging XVIII: Displaying, Processing, Hardcopy, and Applications*, vol. 8652. International Society for Optics and Photonics, 2013, p. 86520K.

[15] T. Mitsa and K. J. Parker, "Digital halftoning technique using a blue-noise mask," *JOSA A*, vol. 9, no. 11, pp. 1920–1929, 1992.

[16] Q. Yu and K. J. Parker, "Adaptive color halftoning for minimum perceived error using the blue noise mask," in *Color Imaging: Device-Independent Color, Color Hard Copy, and Graphic Arts II*, vol. 3018. SPIE, 1997, pp. 272–276.

[17] B. Zhou and X. Fang, "Improving mid-tone quality of variable-coefficient error diffusion using threshold modulation," in *ACM SIGGRAPH 2003 Papers*, 2003, pp. 437–444.

[18] A. Lagae, C. Kaplan, C.-W. Fu, V. Ostromoukhov, J. Kopf, and O. Deussen, "Tile-based methods for interactive applications," *SIGGRAPH Classes*, pp. 93–1, 2008.

[19] P. G. Roetling, "Halftone method with edge enhancement and moiré suppression," *JOSA*, vol. 66, no. 10, pp. 985–989, 1976.

[20] R. Eschbach and K. T. Knox, "Error-diffusion algorithm with edge enhancement," *JOSA A*, vol. 8, no. 12, pp. 1844–1850, 1991.

[21] J. Z. Lai and C.-C. Chen, "Algorithms of halftoning color images with edge enhancement," *Journal of Visual Communication and Image Representation*, vol. 14, no. 4, pp. 389–404, 2003.

[22] X. Li, "Edge-directed error diffusion halftoning," *IEEE Signal Processing Letters*, vol. 13, no. 11, pp. 688–690, 2006.

[23] W.-M. Pang, Y. Qu, T.-T. Wong, D. Cohen-Or, and P.-A. Heng, "Structure-aware halftoning," in *ACM SIGGRAPH 2008 papers*, 2008, pp. 1–8.

[24] J. Chang, B. Alain, and V. Ostromoukhov, "Structure-aware error diffusion," in *ACM SIGGRAPH Asia 2009 papers*, 2009, pp. 1–8.

[25] D. Anastassiou, "Neural net based digital halftoning of images," in *1988., IEEE International Symposium on Circuits and Systems*. IEEE, 1988, pp. 507–510.

[26] K. R. Crounse, T. Roska, and L. Chua, "Image halftoning with cellular neural networks," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 40, no. 4, pp. 267–283, 1993.

[27] Y.-T. Kim, G. R. Arce, and N. Grabowski, "Inverse halftoning using binary permutation filters," *IEEE transactions on image processing*, vol. 4, no. 9, pp. 1296–1311, 1995.

[28] P. W. Wong, "Inverse halftoning and kernel estimation for error diffusion," *IEEE Transactions on Image Processing*, vol. 4, no. 4, pp. 486–498, 1995.

[29] L.-M. Chen and H.-M. Hang, "An adaptive inverse halftoning algorithm," *IEEE transactions on image processing*, vol. 6, no. 8, pp. 1202–1209, 1997.

[30] T. D. Kite, N. Damera-Venkata, B. L. Evans, and A. C. Bovik, "A high quality, fast inverse halftoning algorithm for error diffused halftones," in *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No. 98CB36269)*, vol. 2. IEEE, 1998, pp. 59–63.

[31] Z. Xiong, M. T. Orchard, and K. Ramchandran, "Inverse halftoning using wavelets," *IEEE transactions on image processing*, vol. 8, no. 10, pp. 1479–1483, 1999.

[32] S. Hein and A. Zakhor, "Halftone to continuous-tone conversion of error-diffusion coded images," in *Sigma Delta Modulators*. Springer, 1993, pp. 133–154.

[33] G. B. Unal and A. E. Çetin, "Restoration of error-diffused images using projection onto convex sets," *IEEE transactions on image processing*, vol. 10, no. 12, pp. 1836–1841, 2001.

[34] M. Y. Ting and E. A. Riskin, "Error-diffused image compression using a binary-to-gray-scale decoder and predictive pruned tree-structured vector quantization," *IEEE transactions on image processing*, vol. 3, no. 6, pp. 854–858, 1994.

[35] M. Mese and P. P. Vaidyanathan, "Look-up table (lut) method for inverse halftoning," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1566–1578, 2001.

[36] J.-H. Lee, H.-J. Wu, and M.-Y. Wu, "A reversible data hiding scheme to inverse halftoning." in *SIGMAP*, 2009, pp. 86–89.

[37] J.-H. Lee, M.-Y. Wu, and H.-J. Wu, "A new inverse halftoning method using reversible data hiding for halftone images," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 1–13, 2010.

[38] C.-H. Son, "Inverse halftoning based on sparse representation," *Optics letters*, vol. 37, no. 12, pp. 2352–2354, 2012.

[39] C.-H. Son and H. Choo, "Local learned dictionaries optimized to edge orientation for inverse halftoning," *IEEE Transactions on Image Processing*, vol. 23, no. 6, pp. 2542–2556, 2014.

[40] P. G. Freitas, M. C. Farias, and A. P. Araújo, "Enhancing inverse halftoning via coupled dictionary training," *Signal Processing: Image Communication*, vol. 49, pp. 1–8, 2016.

[41] Y. Zhang, E. Zhang, W. Chen, Y. Chen, and J. Duan, "Sparsity-based inverse halftoning via semi-coupled multi-dictionary learning and structural clustering," *Engineering Applications of Artificial Intelligence*, vol. 72, pp. 43–53, 2018.

[42] T.-W. Yue and G.-T. Chen, "An auto-invertible neural network for image halftoning and restoration," in *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 3. IEEE, 1995, pp. 1450–1455.

[43] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.

[44] W.-B. Huang, A. W. Su, and Y.-H. Kuo, "Neural network based method for image halftoning and inverse halftoning," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2491–2501, 2008.

[45] Y. Xiao, C. Pan, X. Zhu, H. Jiang, and Y. Zheng, "Deep neural inverse halftoning," in *2017 International Conference on Virtual Reality and Visualization (ICVRV)*. IEEE, 2017, pp. 213–218.

[46] Q. Gao, X. Shu, and X. Wu, "Deep restoration of vintage photographs from scanned halftone prints," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4120–4129.

[47] Y.-T. Yen, C.-C. Cheng, and W.-C. Chiu, "Inverse halftone colorization: Making halftone prints color photos," in *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2021, pp. 1734–1738.

[48] T. Luo, G. Jiang, M. Yu, F. Shao, and Z. Peng, "Disparity based stereo image reversible data hiding," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 5492–5496.

[49] W.-C. Yang and L.-H. Chen, "Reversible dct-based data hiding in stereo images," *Multimedia Tools and Applications*, vol. 74, no. 17, pp. 7181–7193, 2015.

This article has been accepted for publication in IEEE Transactions on Visualization and Computer Graphics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2023.3278691

12

[50] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "Hidden: Hiding data with deep networks," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 657–672.

[51] R. L. de Queiroz and K. M. Braun, "Color to gray and back: color embedding into textured gray images," *IEEE Transactions on Image Processing*, vol. 15, no. 6, pp. 1464–1470, 2006.

[52] Z.-X. Xu and Y.-H. Chan, "Improving reversible color-to-grayscale conversion with halftoning," *Signal Processing: Image Communication*, vol. 52, pp. 111–123, 2017.

[53] T. Ye, Y. Du, J. Deng, and S. He, "Invertible grayscale via dual features ensemble," *IEEE Access*, vol. 8, pp. 89 670–89 679, 2020.

[54] Y. Li, D. Liu, H. Li, L. Li, Z. Li, and F. Wu, "Learning a convolutional neural network for image compact-resolution," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1092–1107, 2018.

[55] J. Xing, W. Hu, and T.-T. Wong, "Scale-arbitrary invertible image downscaling," *arXiv preprint arXiv:2201.12576*, 2022.

[56] W. Tan, B. Yan, C. Lin, and X. Niu, "Cycle-ir: Deep cyclic image retargeting," *IEEE Transactions on Multimedia*, vol. 22, no. 7, pp. 1730–1743, 2019.

[57] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[58] J.-H. Jacobsen, A. Smeulders, and E. Oyallon, "i-revnet: Deep invertible networks," *arXiv preprint arXiv:1802.07088*, 2018.

[59] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," *Advances in neural information processing systems*, vol. 31, 2018.

[60] J. Behrmann, W. Grathwohl, R. T. Chen, D. Duvenaud, and J.-H. Jacobsen, "Invertible residual networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 573–582.

[61] L. Ardizzone, C. Lüth, J. Kruse, C. Rother, and U. Köthe, "Guided image generation with conditional invertible neural networks," *arXiv preprint arXiv:1907.02392*, 2019.

[62] M. Xiao, S. Zheng, C. Liu, Y. Wang, D. He, G. Ke, J. Bian, Z. Lin, and T.-Y. Liu, "Invertible image rescaling," in *European Conference on Computer Vision*. Springer, 2020, pp. 126–144.

[63] Y. Liu, Z. Qin, S. Anwar, P. Ji, D. Kim, S. Caldwell, and T. Gedeon, "Invertible denoising network: A light solution for real noise removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 365–13 374.

[64] R. Zhao, T. Liu, J. Xiao, D. P. Lun, and K.-M. Lam, "Invertible image decolorization," *IEEE Transactions on Image Processing*, vol. 30, pp. 6081–6095, 2021.

[65] M. W. Spratling, "A review of predictive coding algorithms," *Brain and cognition*, vol. 112, pp. 92–97, 2017.

[66] ——, "Predictive coding as a model of biased competition in visual attention," *Vision research*, vol. 48, no. 12, pp. 1391–1408, 2008.

[67] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The loco-i lossless image compression algorithm: Principles and standardization into jpeg-ls," *IEEE Transactions on Image processing*, vol. 9, no. 8, pp. 1309–1324, 2000.

[68] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the h. 264/avc video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, pp. 560–576, 2003.

[69] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[70] T. Han, W. Xie, and A. Zisserman, "Video representation learning by dense predictive coding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[71] R. I.-R. BT *et al.*, "Studio encoding parameters of digital television for standard 4: 3 and wide-screen 16: 9 aspect ratios," *Int. Radio Consultative Committee Int. Telecommun. Union, Switzerland, CCIR Rep*, pp. 624–4, 2011.

[72] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," *arXiv preprint arXiv:1308.3432*, 2013.

[73] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[74] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[75] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results," http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[76] Y.-H. Fung and Y.-H. Chan, "Tone-dependent error diffusion based on an updated blue-noise model," *Journal of Electronic Imaging*, vol. 25, no. 1, pp. 013 013–013 013, 2016.

[77] R. Ulichney, *Digital halftoning*. MIT press, 1987.

[78] M. Kumar, D. Weissenborn, and N. Kalchbrenner, "Colorization transformer," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=5NA1PinlGFu

[79] A. Baronchelli, E. Caglioti, and V. Loreto, "Measuring complexity with zippers," *European journal of physics*, vol. 26, no. 5, p. S69, 2005.

[80] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[81] R. Avinery, M. Kornreich, and R. Beck, "Universal and accessible entropy estimation using a compression algorithm," *Physical review letters*, vol. 123, no. 17, p. 178102, 2019.

[82] J.-l. Gailly and M. Adler, "Zlib compression library," 2004.
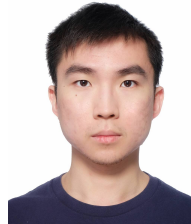
Cheuk-Kit Lau received a B.Sc degree in Computer Science from The Chinese University of Hong Kong in 2017. He begins to pursue an M.Phil degree with the Department of Computer Science & Engineering of the Chinese University of Hong Kong in 2021. His research interests include image processing, video generation, deep learning and computer vision.

Menghan Xia received a B.Eng. degree in Remote Sensing Science and Techniques in 2014 and a Master degree in Pattern Recognition and Intelligent Systems in 2017 from Wuhan University. In 2021, he obatained his Ph.D. degree in Computer Science from The Chinese University of Hong Kong. He is currently a senior researcher in Tencent AI Lab. His research interests include image & video generation, facial animation, cross-modal translation and deep learning.

Tien-Tsin Wong graduated from the Chinese University of Hong Kong in 1992 with a B.Sc. degree in Computer Science. He obtained his M.Phil. and Ph.D. degrees in Computer Science from the same university in 1994 and 1998, respectively. In August 1999, he joined the Computer Science & Engineering Department of the Chinese University of Hong Kong. He is currently a professor. He is a core member of the Virtual Reality, Visualization and Imaging Research Centre in The Chinese University of Hong Kong. His main research interests include computer graphics, computational manga, precomputed lighting, image-based rendering, GPU techniques, medical visualization, multimedia compression, and computer vision.