Deep Halftoning with Reversible Binary Pattern

Menghan Xia[1,3] Wenbo Hu[1] Xueting Liu[2] Tien-Tsin Wong[1,3*]
[1] The Chinese University of Hong Kong [2] Caritas Institute of Higher Education
[3] Guangdong-Hong Kong-Macao Joint Laboratory of Human-Machine Intelligence-Synergy Systems, SIAT, Chinese Academy of Science
{mhxia, wbhu, ttwong}@cse.cuhk.edu.hk tliu@cihe.edu.hk

Abstract

Existing halftoning algorithms usually drop colors and fine details when dithering color images with binary dot patterns, which makes it extremely difficult to recover the original information. To dispense the recovery trouble in future, we propose a novel halftoning technique that converts a color image into binary halftone with full restorability to the original version. The key idea is to implicitly embed those previously dropped information into the halftone patterns. So, the halftone pattern not only serves to reproduce the image tone, maintain the blue-noise randomness, but also represents the color information and fine details. To this end, we exploit two collaborative convolutional neural networks (CNNs) to learn the dithering scheme, under a non-trivial self-supervision formulation. To tackle the flatness degradation issue of CNNs, we propose a novel noise incentive block (NIB) that can serve as a generic CNN plug-in for performance promotion. At last, we tailor a guiding-aware training scheme that secures the convergence direction as regulated. We evaluate the invertible halftones in multiple aspects, which evidences the effectiveness of our method.

1. Introduction

Halftoning is commonly used in the printing industry [44] to reproduce tone with limited colors, e.g. black and white, due to the cost consideration. During this process, both the color and fine details of the original image are inevitably lost. This makes the originals nearly impossible to be recovered from these degraded halftones. Even the state-of-the-art inverse halftoning methods [48, 16] 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. This paper makes the first attempt to explore this novel problem.

Traditional halftoning methods distribute halftone dots mainly for tone reproduction, and we observe that this target still permits certain perturbation in term of the desired binary pattern, as evidenced in Figure 1. It indicates the possibility of utilizing such 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 restorability to the original color version. Inspired by invertible grayscale [47], we adopt the invertible generative model to formulate our problem. However, generating quality halftones is much more challenging than decolorization. First, convolutional neural networks (CNN) that work with spatially shared kernels is not native for halftoning, which suffer from flatness degradation (as detailed in Section 3.1). Figure 3 illustrates an example that CNNs fail to introduce spatial variation in those flat regions. Second, it is non-trivial to achieve both complex visual simulation and accurate information embedding via optimization over 1-bit pixels. Furthermore, the discrete binary pattern poses challenge to capturing its properties via general pixel-wise metrics.

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

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

To address flatness degradation, we propose a *Noise Incentive Block* (NIB) that introduces spatial variation to the feature space but still reserves the information intactness through training along with the CNNs. In fact, NIB is a model-agnostic plug-in and hence applicable to other relevant applications (see Section 4.3). We find that the equipping the dithering network with NIB breaks the obstacle of flatness degradation and make it feasible to dither

*Corresponding author.

constant-valued images. Importantly, this feature enables us to formulate the blue-noise profile through low-frequency penalization on constant-grayness halftone. To achieve the binary halftone, we append the dithering network with a binary gate that takes gradient propagation tricks to allow training with quantization. The model is trained end-to-end with highly mixed objectives, which is formulated as four loss terms: binarization loss, blue-noise loss, halftone conformity loss, and invertibility loss. Indeed, these partially conflicting loss terms complicate the training, especially in the case of inaccurate proxy gradient from the binary gate. These challenges are circumvented by our guiding-aware training scheme.

Comparative evaluation and ablation study illustrate the advantages of our proposed method, and application exploration tells the generic usability of our proposed noise incentive block. The paper contributes in:

- The innovative idea of *reversible halftoning*, which offers a brand new functionality to existing halftoning applications. It saves the ill-posed inverse halftoning problem at the source.
- A model-agnostic plug-in, noise incentive block that addresses the flatness degradation of CNNs. It finds general applicability in image synthesis tasks.
- An effective measurement for discrete halftoning patterns, which may inspire further exploration in relevant direction, e.g. manga screentone processing.

2. Related Works

2.1. Digital Halftoning

Many digital halftoning techniques have been proposed in the past half century, including ordered dithering [3, 44], error diffusion [10], dot diffusion [21], and direct binary search (DBS) [37]. The primary goal of the above classic approaches is to retain the local tone of the original image with least visual artifacts introduced. To avoid extra pattern being introduced, it is desirable for the halftone images to own the blue-noise property [32]. To achieve this, several techniques are further proposed, such as blue-noise mask [33], patterns optimization [12, 26, 2], variable threshold [34, 55], and recursive tiling strategy [22]. Unfortunately, blue-noise algorithms usually over-blur fine details. To better preserve the fine structures while pursuing the blue-noise property, several methods have been proposed to generate halftone patterns guided by edge enhancement techniques [8, 14, 23, 27]. Different from the edge enhancement techniques, Pang et al. [35] further proposed to optimize both structural similarity and tonal similarity to capture human-vision-sensitive structures. However, while digital halftoning has long been a widely-explored topic for researchers, none of the existing methods considers the invertibility, i.e. the ability to recover the original version.

2.2. Inverse Halftoning

Inverse halftoning has been studied in the past three decades, mainly due to the need to recover the pictures from the legacy printed media. A straightforward approach is to filter the halftone image with customized filters [17, 46]. In order to better restore high-frequency details, Kite et al. [20] proposed to exploit the gradient-based space-varying filtering on the error-diffused images. To relieve the prior information requirement, some works propose to formulate the grayscale reconstruction problem as projection onto a convex set (POCS). Yue and Chen [53] proposed to inverse halftoning via a hopfield neural network [13] based optimization model. Xiong et al. [50] proposed to separate halftoning noises from original image through edge detection, and reconstruct the original image based on overcomplete wavelet expansions. Mese and Vaidyanathan [31] further proposed to utilize a precomputed look-up table (LUT) for grayscale restoration, which greatly improves both efficiency and effectiveness. Multiple dictionary learning based variants have been proposed since then [24, 25, 39, 40, 11]. Most recently, deep learning methods [16, 48] have been proposed to solve the inverse halftoning, and achieve the state-of-the-art performance. However, inverse halftoning is ill-posed by nature and can only "guess" a rough grayscale version. In contrast, our invertible halftone enables a deterministic restoration process, which thus can achieve higher accuracy.

2.3. Invertible Generation

The idea of invertible generation is mainly studied in the data hiding field, e.g. hiding copyright text or watermarks in images [29, 52, 56]. Later, researchers attempted to hide an image into anther image, such as hiding chromatic channels into its grayscale version [7, 51], or hiding one view of a stereoscopic image into the other view [41, 42]. Recently, a significant improvement has been made by using the deep convolutional neural networks. In particular, Xia et el. [47] proposed to convert color images to invertible grayscale images that can be later inverted back to its color version via an encoding-and-decoding framework. A similar framework is adopted to generate invertible low-resolution images from high-resolution input with details compactly encoded [28]. Our reversible halftoning can be classified into this stream of work, addressing a more challenging problem with novel technical designs. As another line of such technique, invertible neural networks (INNs) [15, 19, 4, 49] formulate the network architecture with explicit invertible operations. However, compared to the encoding-and-decoding based invertible model [47], such strong constraint inevitably restricts the model capability and makes the training tricky.

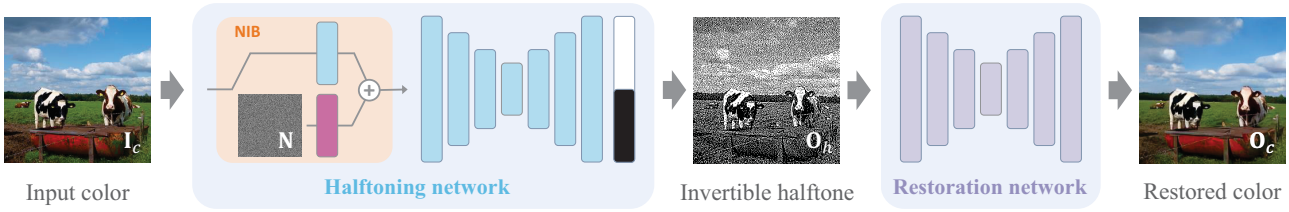


Figure 2. System overview. Given a color image input $\mathbf{I}_c$, the dithering network generates a bitonal halftone image $\mathbf{O}_h$, which could be inverted back to the original color version $\mathbf{O}_c$ by the restoration network. Particularly, the dithering network is equipped with a noise incentive block (NIB) and a binary gate, in order to enable the binary halftoning for arbitrary input.

3. Reversible Halftoning Pattern

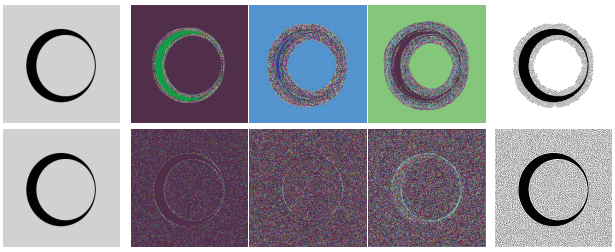
We aims to learn reversible binary patterns toward halftoning color images, which is required to offer visual pleasantness and embed restoration-necessary information in the meantime. Figure 2 shows the overview diagram.

3.1. Network Architecture

We adopt the U-shaped architecture for both the dithering networks and the restoration network. Both networks share a similar structure, containing three downscale blocks, three upscale blocks, four residual blocks, and two convolution blocks. The detailed architecture parameters are provided in the supplementary material. Note that, we adopt U-Net as network backbone just because its enlarged receptive field, and other qualified CNN architectures may also works. Additionally, we propose two key designs for the dithering network, i.e. noise incentive block and binary gate, which enables the CNN to model halftoning properly.

Noise Incentive Block. We find that typical CNNs, consisting of convolutional layers with bias terms and activation functions, are unable to introduce spatial variation to the output when fed with a flat input. We call this phenomena *flatness degradation*, which is caused by the convolution paradigm with spatially shared kernels. Formally, the convolution of a constant signal $s(x) \equiv c$ and an arbitrary kernel function $k(x)$ is definitely anther constant signal $\tilde{s}(y) = c \cdot \mu(k(x))$ where $\mu(\cdot)$ takes the mean value. So, given a flat input $\mathbf{X}$, the manipulation of CNN degrades to a scaling operation $\mathbf{Y} = \alpha\mathbf{X}$ regardless of the CNN parameters. Figure 3 illustrates such an example. As a result, flatness degradation hinders the CNN to dither a constant-grayness, which can disable the formulation of blue-noise profile (introduced in Section 3.2) since the blue-noise profile is mostly measured over the constant-grayness [43, 34].

To circumvent the flatness degradation, we propose the Noise Incentive Block (NIB) that can be used as a model-agnostic plug-in for CNNs. The key idea is to introduce spatial variation to the feature representation but never contaminate the intactness of the original input information. For simplicity, a Gaussian noise map is exploited as additive



(a) Input (b) Intermediate Feature maps (c) halftone

Figure 3. Visualization of CNN halftoning. Due to the flatness degradation, typical CNNs fail to generate spatial variation in flat regions (up row); The NIB equipped CNNs can address the limitation effectively (bottom row).

variation proxy in the learned feature space: $\mathbf{f}_1(\mathbf{I}_c)+\mathbf{f}_2(\mathbf{N})$, where both $\mathbf{f}_1$ and $\mathbf{f}_2$ are single convolution layer and $\mathbf{N}$ is a dynamically sampled Gaussian noise map. Equipped with NIB, our dithering network is free of flatness degradation and thus can generate binary halftone in flat regions, as the visualized results shown in Figure 3. In practice, the NIB brings advantages to our dithering model in two aspects: (i) It allows us to formulate the blue-noise profile through low-frequency constraint on dithered constant-grayness; (ii) It even promotes the performance in general cases (see Table 3) since the noise randomness favors the dithering processing to focus on pattern distribution instead of individual pixel values. Importantly, as a generic solution to tackle flatness degradation, our proposed NIB also finds noticeable advantages in other relevant applications, which is explored in Section 4.3.

Binary Gate. Another special design for the dithering network is the binary gate $\mathbf{B}(\cdot)$ that quantizes the network output $\tilde{\mathbf{O}}_h$ to be a strict binary image $\mathbf{O}_h = \mathbf{B}(\tilde{\mathbf{O}}_h)$. 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 the quantization when stored as 1-bit bitmap and thus hurts the restoration accuracy. However, there is one obstacle should be noted that the binary gate is non-differentiable. To enable the joint training, we use Straight-Through Estimator [5] on the binarization when calculating the gradients.

3.2. Loss Function

Our network is trained with the loss function defined in Eq. 1, consisting of halftone conformity loss $\mathcal{L}_C$, blue-noise loss $\mathcal{L}_N$, binarization loss $\mathcal{L}_B$, and invertibility loss $\mathcal{L}_V$.

$$\mathcal{L} = \omega_1 \mathcal{L}_C + \omega_2 \mathcal{L}_N + \omega_3 \mathcal{L}_B + \mathcal{L}_V, \tag{1}$$

where the hyper-parameters $\omega_1 = 0.6$, $\omega_2 = 0.3$, and $\omega_3 = 0.1$ are set empirically.

3.2.1 Halftone Conformity Loss

To ensure that the generated halftone is visually similar to the input, we optimize the tone and structure similarities, as suggested by [35]. Hence, our halftone conformity loss $\mathcal{L}_C$ is formulated as:

$$\mathcal{L}_C = \ell_T + \sigma \ell_S, \tag{2}$$

where the tone loss ℓ_T and the structure loss ℓ_T is combined with the coefficient $\sigma = 0.02$ empirically.

Specifically, given the color input $\mathbf{I}_c$ and the generated halftone $\mathbf{O}_h$, the tone loss ℓ_T is formulated as:

$$\ell_T = \mathbb{E}_{\mathbf{I}_c \in \mathcal{I}}\{||\mathrm{G}(\mathbf{O}_h) - \mathrm{G}(\mathbf{I}_c^l)\}||_2\}, \tag{3}$$

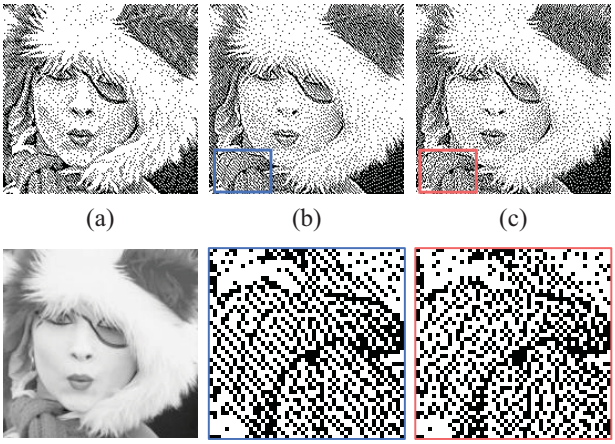
where $\mathbf{I}_c^l$ is the luminance channel of the original input image $\mathbf{I}_c$. $\mathrm{G}(\cdot)$ denotes the Gaussian filter with a 11×11 kernel size. $||\cdot||_2$ denotes the L_2 norm (MSE). $\mathbb{E}_{\mathbf{I}_c \in \mathcal{I}}\{\cdot\}$ denotes the average operator over all input images $\mathbf{I}_c$ in the training dataset $\mathcal{I}$. Accordingly, the structure loss ℓ_S measures the structure similarity index measure (SSIM) [45] between $\mathbf{O}_h$ and the luminance channel of the original image $\mathbf{I}_c^l$:

$$\ell_S = \mathbb{E}_{\mathbf{I}_c \in \mathcal{I}}\{||\mathrm{SSIM}(\mathbf{O}_h, \mathbf{I}_c^l)||_1\}, \tag{4}$$

3.2.2 Blue-Noise Loss

The blue-noise property is typically required in traditional halftoning algorithms [43], so as to avoid injecting extra patterns. Although SSIM has been shown to be useful to achieve blue-noise property [35], we found that adopting SSIM alone as the loss term is insufficient to enforce the blue-noise property. Instead, we design an explicit blue-noise loss in our loss function to suppress the potential pattern artifacts. The basic idea to penalize the low-frequency component on the dithered constant-grayness, since human vision system is more sensitive to low-frequency signal. Accordingly, we prepare a group of constant-valued color images $\mathcal{S}$ and include them in our training dataset. The blue-noise loss $\mathcal{L}_N$ is formulated on the halftone images generated from these constant-valued images, as

$$\mathcal{L}_N = \mathbb{E}_{\mathbf{I}_c \in \mathcal{S}}\{||(\mathrm{DCT}(\mathbf{O}_h) - \mathrm{DCT}(\mathbf{I}_c)) \odot \mathbf{M}||_1\}, \tag{5}$$



(a) (b) (c)

(d) Input grayscale (e) Blowup of (b) and (c)

Figure 4. Effects of the halftone conformity loss and blue-noise loss. (a) Without conformity loss and blue-noise loss; (b) Without blue-noise loss; (c) With the full loss.

where $\mathrm{DCT}(\cdot)$ denotes the discrete cosine transformation (DCT), $\odot$ means element-wise product, and $\mathbf{M}$ denotes a constant binary mask with the low-frequency components set to 1 and others set to 0. Specifically, we regard the first 3.8% low-frequency DCT coefficients as the target component to minimize, and the result shows it works well. Figure 4(a), (b) exhibit visually annoying stripe patterns, while in comparison, additionally employing the blue-noise loss resolves this problem effectively (c).

3.2.3 Binarization Loss

Although the binary gate explicitly binarizes the dithering network output, it relies on a rough gradient estimator to enable the gradient propagation. However, the straight-through estimator simply use an identity gradient, which actually makes the binarization operation $\mathbf{B}(\cdot)$ ignored in the backward propagation. To guarantee the training stability, we encourage the input value to $\mathbf{B}(\cdot)$ to be as close to 0 or 1 as possible, which is formulated as the binarization loss $\mathcal{L}_B$:

$$\mathcal{L}_B = \mathbb{E}_{\mathbf{I}_c \in \mathcal{I}}\{||\min_{d=\{0,1\}}\{|\tilde{\mathbf{O}}_h - \mathbf{C}_d|\}||_1\}, \tag{6}$$

where $\tilde{\mathbf{O}}_h$ is the *pseudo halftone image* obtained before the binary gate. $\mathbf{C}_d$ is a constant-value matrix with the same size as $\tilde{\mathbf{O}}_h$ in which all elements are equal to d, $|\cdot|$ is the element-wise absolute operator, $\min\{\cdot\}$ is the element-wise minimum operator, and $||\cdot||_1$ denotes the L_1 norm. In experiment, dropping $\mathcal{L}_B$ would collapse the training.

3.2.4 Invertibility Loss

While the three losses above regulate the visual quality of the halftone, the invertibility loss $\mathcal{L}_L$ ensures the restored

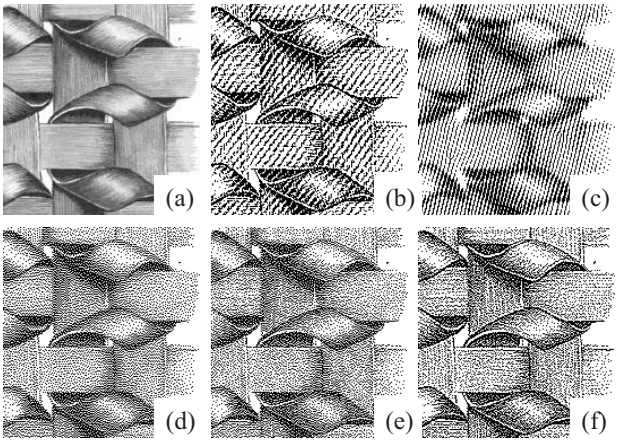


Figure 5. Halftone output under different training schemes. (a) Input; (b) Joint training from scratch; (c) Warm-up training for 130 epochs w/o guidance loss; (d) Error diffusion; (e) Warm-up training for 28 epochs with guidance loss. (f) Two-stage training.

image to be as similar to the input as possible, formulated in both pixel level (measured by pixel-wise MSE) and perceptual level (measured by MSE of VGG features):

$$\mathcal{L}_V = \mathbb{E}_{\mathbf{I}_c \in \mathcal{I}}\{||\mathbf{O}_c - \mathbf{I}_c||_2 + \lambda||\Psi(\mathbf{O}_c) - \Psi(\mathbf{I}_c)||_2\}, \quad (7)$$

where $\Psi(\cdot)$ denotes the feature map from the *conv4_4* layer of a pretrained VGG-19 network [38], which empirically represents the perception feature of images. Weight $\lambda = 2.0 \times 10^{-4}$ is empirically set to balance the magnitudes between the two domains. Under joint training, the invertibility loss propagates gradient to both dithering network $\mathbf{E}$ and restoration network $\mathbf{D}$.

3.3. Training Scheme

To learning desired halftone patterns, the dithering and restoration networks are trained jointly by minimizing the loss function in Eq. 1. However, training the whole model from scratch is vulnerable to local minimal because of the challenging optimization target. Figure 5(b) shows the failure of halftones. To circumvent this problem, we propose to adopt the two-stage 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 fail to guarantee satisfactory halftones (Figure 5(d)), and it is even associated with slow convergence, as shown in Figure 6 (green curve). To boost the training, we propose to explicitly provide a reference halftone image $\mathbf{I}_h$ to guide the training. For simplicity, the classical error diffusion [34] 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.

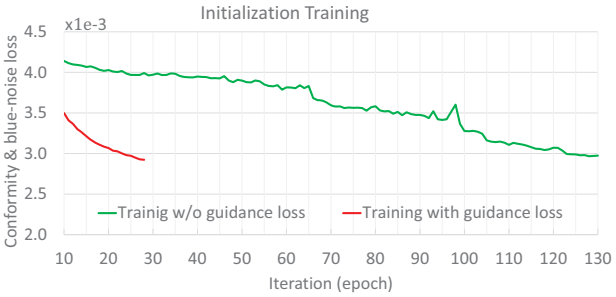


Figure 6. Halftone visual loss against iteration of warm-up training with and w/o the *guidance loss* $\mathcal{L}_G$.

Halftone Pattern Measurement. Inspired by perceptual loss [54], we propose to measure the halftone pattern difference in continuous feature domain. Specifically, it is modeled by a pretrained inverse halftoning network $\mathbf{F}$ (detailed in supplementary material), which shows decent sensitivity to capture the halftone patterns. Accordingly, we formulate the guidance loss $\mathcal{L}_G$ as

$$\mathcal{L}_G = \mathbb{E}_{\mathbf{I}_c \in \mathcal{I}}\{||\mathbf{F}(\mathbf{O}_h) - \mathbf{F}(\mathbf{I}_h)||_2\}, \quad (8)$$

Then, we train the dithering network with the combined loss: $\mathcal{L}_G + \omega_1\mathcal{L}_C + \omega_2\mathcal{L}_B + \omega_2\mathcal{L}_N$. The red curve in Figure 6 shows the high training efficiency. With only 28 epochs, it is able to generate visual decent results, as shown in Figure 5(e). Anyhow, there is no need to train it until convergence since visually good halftone is just part of the terminal target.

In the second stage, we drop the guidance loss and jointly train the whole model under the full loss in Eq. 1 for another 115 epochs. Figure 5(f) shows the final result. With the ADAM solver [18], it takes 143 epochs to complete the whole training. In both stages, we leverage the ReduceLRonPlateau learning rate scheduler to manage the learning rate, which is set to 0.0001 initially and then halved when the training loss stops to decrease.

4. Experimental Results

Dataset. Our training dataset is collected from the publicly available VOC2012 dataset [9]. There are $17,125$ color images in the dataset. We randomly select $13,758$ of them for training, and reserve the rest $3,367$ images for quantitative evaluation as testing dataset. All these images has the size of 256×256 by cropping and resizing.

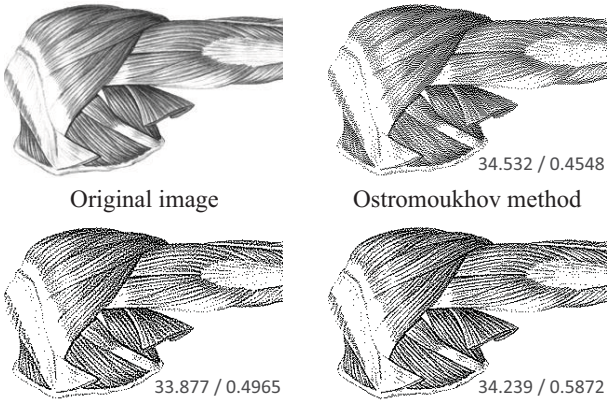
The evaluation conducted on the halftone quality and its restorable accuracy, for which we use the same model trained with default hyper-parameters (defined in Eq. 1). The source code and trained model are available at: https://github.com/MenghanXia/ReversibleHalftoning

4.1. Halftone Quality

Visual Conformity. We measure the visual conformity of the halftones to the input in terms of tone and structure.

Table 1. Evaluation on halftone images in terms of PSNR, and SSIM. Higher PSNR/SSIM indicate better quality.

Method	PSNR		SSIM	
	Mean	Stddev	Mean	Stddev
Ostromoukhov method	41.728	1.1235	0.1007	0.0690
Structure-aware halftoning	21.803	2.2570	0.0340	0.0500
Ours (grayscale input)	33.262	0.6206	0.1594	0.0888
Ours (color input)	32.861	0.7899	0.1573	0.0877

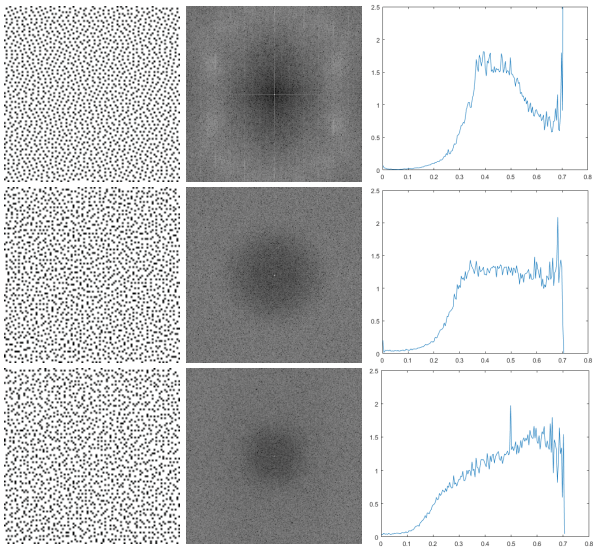


Original image | Ostromoukhov method | Structure-aware halftoning | Our method

Figure 7. Arm with legible structures. The PSNR/SSIM of each halftone image is annotated for reference.

Following the practice in [35], 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 perform the experiment over $3,367$ grayscale images (decolorized from our testing dataset), as existing halftoning methods can only dither grayscale images. Two classical halftoning methods that generates high-quality halftones are selected as our competitors, Ostromoukhov method [34] and the structure-aware halftoning method [35]. In our experiment, the structure-aware halftoning method is used with default parameters for quantitative evaluation while 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). Figure 7 illustrates an example for visual comparison. Besides, our method obtained as good result from color image dithering.

Blue-Noise Profile. As usually, the blue-noise property of halftoning methods is analyzed on their generated halftones of constant-grayness images. Specifically, we compute the Fourier amplitude spectrum and radially averaged power spectra [35]. The Fourier amplitude spectrum indicates the amplitude of the frequency components where lower-frequency components are expected to have lower amplitude in blue-noise profiles. The radially averaged power spectrum visualizes the blue-noise property in 1D. As shown in Figure 8, all three results achieves certain degree of blue-noise property. The Ostromoukhov method and structure-aware halftoning performs slightly better than our method since they only target for halftone quality without invertibility consideration. Actually, our method trades the blue-noise property for the restoration accuracy, as both the color information and the blue-noise compete for the distribution of halftone dots.



Halftone | Fourier amplitude | Radially averaged profile

Figure 8. Spectral analysis on the halftone of constant-grayness (grayness=0.8). Ostromoukhov method (top); Structure-aware halftoning (middle); Our method (bottom).

4.2. Restoration Accuracy

We compare our method with the state-of-the-art inverse halftoning method, PRL-Net [48], which recovers grayscale from error diffused halftones. Anyway, this comparison may not be very appropriate since the input to PRL-Net is error-diffused halftones, while the input to our restoration network is the information-encoded invertible halftones. Since PRL-Net can only restore grayscale images, we prepared $3,367$ grayscale images (decolorized from our testing dataset) for comparison, and additionally evaluate our method on color version restoration from the associated invertible halftones. Table 2 presents the statistics of both PSNR and SSIM, and our method outperforms PRL-Net in both metrics. The numerical superiority is not significant because the major difference lies in fine details that are less sensitively captured by PSRN while matters visual quality. Figure 9 illustrates one challenging example that has rich details with low resolution. PRL-Net fails to restore the fine facade details and the text characters. In comparison, our method not only recovers both fine details and text, but

Table 2. Evaluation on restored images by PSNR, and SSIM. Restoring color is only feasible by our method.

Dataset	Method	PSNR		SSIM	
		Mean	Stddev	Mean	Stddev
Grayscale	PRL-Net	29.693	3.4732	0.8796	0.0564
	Ours	30.386	3.0381	0.9077	0.0395
Color	Ours	28.130	2.6507	0.8592	0.0470

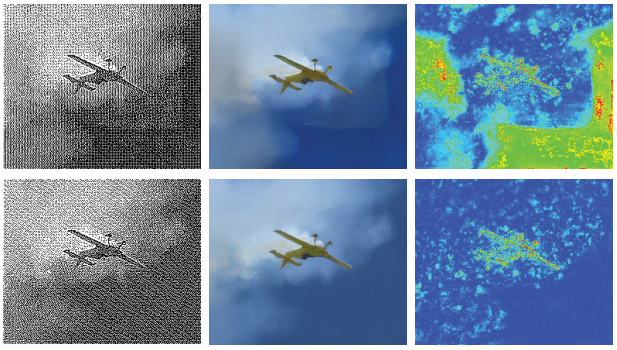
(a) Inverse halftoned (b) Our restored grayscale

(c) Our restored color (d) Ground truth

Figure 9. Inverse halftoning vs. ours. The red arrows point to the problematic reconstruction.

Table 3. Ablation analysis on noise incentive block (NIB). Statistic over the color testing dataset.

Category	Variant	PSNR		SSIM	
		Mean	Stddev	Mean	Stddev
Halftoning	Ours/NIB	31.915	1.8185	0.1514	0.0827
	Ours	33.734	0.6078	0.1702	0.0906
Restoration	Ours/NIB	27.743	2.2795	0.8667	0.0420
	Ours	29.112	2.9705	0.8826	0.0430



Invertible halftone Restored image Error map

Figure 10. Performance comparison between the model without (top row) and with NIB equipped (bottom row). The color-coded error maps visualize the deviation w.r.t the ground truth.

also recovers the original colors when the original input is color version. Our superiority comes from the encoded information in the invertible halftone when PRL-Net [48] has to "guess" the missing information.

In fact, inverse halftoning techniques can be used to recover color images from color halftones, i.e. channel by channel. However, as a naively extension of binary halftone, color halftones require three channels to store the patterns and thus have restricted application scenarios, which will not be discussed here.

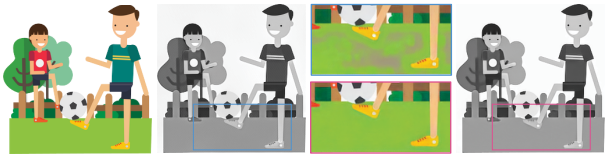
4.3. Utility of Noise Incentive Block

Benefits to Reversible Halftoning. As mentioned in Section 3.1, our proposed noise incentive block (NIB) enables the dithering network to generate binary halftones for constant input. Figure 8 demonstrates its effectiveness in dithering a constant-gray image. To further analyze the effect, we conduct an ablation study on the NIB of our dithering network. Note that, when NIB is not used, the blue-noise loss cannot be applied since it is formulated on the dithered constant-grayness. Regarding this, we intentionally remove the blue-noise loss in all model variants to avoid inducing other factors. The quantitative result over the color testing dataset is tabulated in Table 3. Interestingly, the statistics shows that equipping NIB to the dithering network improves both the halftone generation and color image restoration. It is probably because the randomness introduced by NIB favors the dithering process, i.e. focusing on pattern distribution instead of individual pixel values. In addition, CNNs also partially degrades in smooth regions, which hinders the generation of desired halftone patterns. Figure 10 shows an example to verify this hypothesis.

Application to Generative Model. As a generic solution to address flatness degradation, our noise incentive block (NIB) is evaluated in two relevant applications that are required to generate spatially variation from flat input. First, we apply NIB to color image encoding [47] that encodes a color image into its grayscale version by representing the color information as imperceptible texture patterns. Obviously, if the input image contains some constant-valued regions, the color-encoded texture pattern cannot be generated properly because of flatness degradation and hence affects the color restoration. Figure 11 shows an example that evidences that NIB can address this limitation effectively. Quantitative evaluation on DIV2K dataset [1] is tabulated in Table 4. In addition, we further apply NIB to improve image synthesis from semantic layout [36]. Despite the employed CNN model has a noise input along with the semantic layout, the semantic layout still challenges the convolution layers with flatness degradation because the noise is just used as an initialized result, instead of tackling flatness degradation like NIB. Figure 12 illustrates the comparative results, showing that the NIB-equipped model exhibits notable advantages in synthesizing more realistic textures with sharp details. This is reflected by the significantly decreased FID [30] on benchmark Cityscape [6], as shown in Table 4. It makes no gain in the indirect segmentation accuracy s-

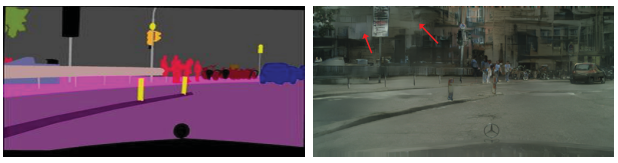
Table 4. Quantitative evaluation on applying NIB to the state-of-the-art models of invertible grayscale and semantic image synthesis respectively.

Method	PSRN	SSIM	
IG [47]	38.411	0.9765	
IG+NIB	39.314	0.9811	
Method	mIOU	Accu	FID
SPADE [36]	62.3%	81.9%	71.8
SPADE+NIB	61.9%	81.8%	54.6



(a) Color input (b) Grayscale by IG (c) Restored color patches (d) Grayscale by IG+NIB

Figure 11. Applying our NIB to IG [47] for color image encoding. The color patches are decoded from the encoded grayscales.



(a) Input semantic layout (b) SPADE

(c) SPADE+NIB (d) Potential ground truth

Figure 12. Applying our NIB to SPADE [36] for semantic image synthesis. The red arrows point to unrealistic blurriness.

ince NIB benefits spatially variant generation while having negligible influences on pixel alignment.

4.4. Discussion

The invertible halftone possesses high restorability to the original color version because the necessary information is embedded via the learned binary pattern. Consequently, outside interferences to the halftone patterns may affect their reversibility potential. To explore the robustness, we apply several typical disturbances to the generated halftones, include flipping, partial removal, and random impulse noises. Figure 13 visualizes the qualitative comparison. Since the information-embedded patterns are sensitive to flipping, the restored color image (b) shows texture artifacts and incorrect colors. Differently, it shows relatively good tolerance to partial removal (b) and random noises (c), which indicates good potential to be used in real-world applications. This is partially confirmed by the decent performance in printing-and-scanning scenario, as provided in the supplementary material.

(a) No operation (b) Flipping (c) Partial removal (d) Random noise

Figure 13. Robustness study of reversible binary patterns. The color images (bottom row) are restored from the halftone variants (top row) with different interferences. The random noise is 10% impulse noise, which is more destructive than Gaussian noise.

In our current formulation, both the information embedding and the blue-noise requirement are competing for the distribution of halftone dots, which means a trade-off has to be taken. So, to improve the overall performance demands a larger solution space. One possible solution to enlarge the representation space, namely to use a large resolution for the generation of invertible halftones, e.g. the halftone resolution can be $\times 2$, $\times 4$ or even larger fold of the input size. This direction offers promising potential to introduce more requirements on the invertible halftones, such as the halftoning styles, which will be an interesting future work.

5. Conclusion

We proposed the conceptually novel reversible halftoning technique, which offers high restorability along with state-of-the-art visual quality. As a stronger alternative, it is directly applicable to traditional halftoning applications but saves the potential trouble of tacking the ill-posed inverse halftoning. To achieve this, we proposed the noise incentive block (NIB) to tackle flatness degradation of CNNs, which not only promotes our dithering performance greatly but also finds impressive utilities in other relevant applications. Besides, the blue-noise loss is formulated as low-frequency constraint on constant-grayness, which effectively guarantees the visual pleasantness of halftone patterns. To handle the tricky optimization landscape, we proposed to modulate the priorities of different loss terms in two stages. Extensive experiments verified the advantages of our method, and we expect both the reversible halftoning approach and key technical designs to inspire follow-up works.

Acknowledgement: This project is supported by Shenzhen Science and Technology Program (No.JCYJ20180507182410327) and CUHK Direct Grant for Research 2020/2021.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.

[2] Farhan A. Baqai and Jan P. Allebach. Halftoning via direct binary search using analytical and stochastic printer models. *IEEE Transactions on Image Processing (TIP)*, 12(1):1–15, 2003.

[3] Bryce E. Bayer. An optimum method for two-level rendition of continuous tone pictures. In *IEEE International Conference on Communications*, 1973.

[4] Jens Behrmann, Will Grathwohl, Ricky T. Q. Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *International Conference on Machine Learning (ICML)*, 2019.

[5] Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *preprint arXiv:1308.3432*, 2013.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[7] Ricardo L. de Queiroz and Karen M. Braun. Color to gray and back: color embedding into textured gray images. *IEEE Transactions on Image Processing (TIP)*, 15(6):1464–1470, 2006.

[8] Reiner Eschbach and Keith T. Knox. Error-diffusion algorithm with edge enhancement. *Journal of the Optical Society of America A*, 8(12):1844–1850, 2017.

[9] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge 2012 (voc2012) results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012.

[10] Robert W. Floyd. An adaptive algorithm for spatial gray-scale. In *Society of Information Display*, 1976.

[11] Pedro Garcia Freitas, Mylène C. Q. Farias, and Aletéia P. F. Araújo. Enhancing inverse halftoning via coupled dictionary training. *Signal Processing: Image Communications*, 49:1–8, 2016.

[12] Robert Geist, Robert Reynolds, and Darrell Suggs. A markovian framework for digital halftoning. *ACM Transactions on Graphics (TOG)*, 12(2):136–159, 1993.

[13] John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. 1982.

[14] Byong-Won Hwang, Tae-Ha Kang, and Tae-Seung Lee. Improved edge enhanced error diffusion based on first-order gradient shaping filter. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, 2004.

[15] Jörn-Henrik Jacobsen, Arnold W. M. Smeulders, and Edouard Oyallon. i-revnet: Deep invertible networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[16] Tae-Hoon Kim and Sang Il Park. Deep context-aware descreening and rescreening of halftone images. *ACM Transactions on Graphics (TOG)*, 37(4):48, 2018.

[17] Yeong-Taeg Kim, R. Arce Gonzalo, and Grabowski Nikolai. Inverse halftoning using binary permutation filters. *IEEE Transactions on Image Processing (TIP)*, 4(9):1296–1311, 1995.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint:1511.06349*, 2014.

[19] Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[20] Thomas D. Kite, Damera-Venkata Niranjan, L. Evans Brian, and C. Bovik Alan. A high quality, fast inverse halftoning algorithm for error diffused halftones. In *IEEE International Conference on Image Processing (ICIP)*, 1998.

[21] Donald E. Knuth. Digital halftones by dot diffusion. *ACM Transactions on Graphics (TOG)*, 6(4):245–273, 1987.

[22] Johannes Kopf, Daniel Cohen-Or, Oliver Deussen, and Dani Lischinski. Recursive wang tiles for real-time blue noise. *ACM Transactions on Graphics (TOG)*, 25(3):509–518, 2006.

[23] Nae-Joung Kwak, Soung-Pil Ryu, and Jae-Hyeong Ahn. Edge-enhanced error diffusion halftoning using human visual properties. In *IEEE International Conference on Hybrid Information Technology*, 2006.

[24] Jia-Hong Lee, Hong-Jie Wu, and Mei-Yi Wu. A reversible data hiding scheme to inverse halftoning. In *International Conference on Signal Processing and Multimedia Applications (SIGMAP)*, 2009.

[25] Jia-Hong Lee, Mei-Yi Wu, and Hong-Jie Wu. A new inverse halftoning method using reversible data hiding for halftone images. *EURASIP Journal on Advances in Signal Processing*, 2010:89, 2010.

[26] Pingshan Li and Jan P. Allebach. Tone-dependent error diffusion. *IEEE Transactions on Image Processing (TIP)*, 13(2):201–215, 2004.

[27] Xin Li. Edge-directed error diffusion halftoning. *IEEE Signal Processing Letter*, 13(11):688–690, 2017.

[28] Yue Li, Dong Liu, Houqiang Li, Li Li, Zhu Li, and Feng Wu. Learning a convolutional neural network for image compact-resolution. *IEEE Transactions on Image Processing (TIP)*, 28(3):1092–1107, 2019.

[29] Ting Luo, Gangyi Jiang, Mei Yu, Feng Shao, and Zongju Peng. Disparity based stereo image reversible data hiding. In *IEEE International Conference on Image Processing (ICIP)*, 2014.

[30] Thomas Unterthiner Bernhard Nessler Martin Heusel, Hubert Ramsauer and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[31] Murat Mese and Palghat P. Vaidyanathan. Look-up table (lut) method for inverse halftoning. *IEEE Transactions on Image Processing (TIP)*, 10(10):1566–1578, 2001.

[32] Don P. Mitchell. Generating antialiased images at low sampling densities. In *ACM Transactions on Graphics (TOG)*, 1987.

[33] Theophano Mitsa and Kevin J. Parker. Digital halftoning technique using a blue-noise mask. *JOSA A*, 9(11):1920–1929, 1992.

[34] Victor Ostromoukhov. A simple and efficient error-diffusion algorithm. In *ACM SIGGRAPH*, 2001.

[35] Wai-Man Pang, Yingge Qu, Tien-Tsin Wong, Daniel Cohen-Or, and Pheng-Ann Heng. Structure-aware halftoning. *ACM Transactions on Graphics (TOG)*, 27(3):89, 2008.

[36] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[37] Michael A. Seldowitz, Jan P. Allebach, and Donald W. Sweeney. Synthesis of digital holograms by direct binary search. *Applied Optics*, 26(14):2788–2798, 1987.

[38] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2014.

[39] Chang-Hwan Son. Inverse halftoning based on sparse representation. *Optics letters*, 37(12):2352–2354, 2012.

[40] Chang-Hwan Son and Hyunseung Choo. Local learned dictionaries optimized to edge orientation for inverse halftoning. *IEEE Transactions on Image Processing (TIP)*, 23(6):2542–2557, 2014.

[41] Xuefeng Tong, Guangce Shen, Guorong Xuan, Shumeng Li, Zhiqiang Yang, Jian Li, and Yun-Qing Shi. Stereo image coding with histogram-pair based reversible data hiding. In *International Workshop on Digital-Forensics and Watermarking*, 2014.

[42] Xuefeng Tong, Xin Wang, Guorong Xuan, Shumeng Li, and Yun Q. Shi. Optimal histogram-pair and prediction-error based reversible data hiding for medical images. In *International Workshop on Digital-Forensics and Watermarking*, 2015.

[43] Robert Ulichney. *Digital halftoning*. MIT press, 1987.

[44] Robert A. Ulichney. Dithering with blue noise. In *Proceedings of the IEEE*, 1988.

[45] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*, 13(4):600–612, 2004.

[46] Ping Wah Wong. Inverse halftoning and kernel estimation for error diffusion. *IEEE Transactions on Image Processing (TIP)*, 4(4):486–498, 1995.

[47] Menghan Xia, Xueting Liu, and Tien-Tsin Wong. Invertible grayscale. *ACM Transactions on Graphics (TOG)*, 37(6):246:1–246:10, 2018.

[48] Menghan Xia and Tien-Tsin Wong. Deep inverse halftoning via progressively residual learning. In *Asian Conference on Computer Vision (ACCV)*, 2018.

[49] Mingqing Xiao, Shuxin Zheng, Chang Liu, Yaolong Wang, and et al. Invertible image rescaling. In *European Conference on Computer Vision (ECCV)*, 2020.

[50] Zixiang Xiong, T. Orchard Michael, and Ramchandran Kannan. Inverse halftoning using wavelets. *IEEE Transactions on Image Processing (TIP)*, 8(10):1479–1483, 1999.

[51] Zi-Xin Xu and Yuk-Hee Chan. Improving reversible color-to-grayscale conversion with halftoning. *Signal Processing: Image Communications*, 52:111–123, 2017.

[52] Wen-Chao Yang and Ling-Hwei Chen. Reversible dct-based data hiding in stereo images. *Multimedia Tools Application*, 74(17):7181–7193, 2015.

[53] Tai-Wen Yue and Guo-Tai Chen. An auto-invertible neural network for image halftoning and restoration. In *IEEE International Conference on Neural Networks (ICNN)*, 1995.

[54] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[55] Bingfeng Zhou and Xifeng Fang. Improving mid-tone quality of variable-coefficient error diffusion using threshold modulatio. *ACM Transactions on Graphics (TOG)*, 22(3):437–444, 2003.

[56] Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. Hidden: Hiding data with deep networks. In *European Conference on Computer Vision (ECCV)*, 2018.