RESEARCH ARTICLE

Super-Resolution of Color Halftone Images Using Convolutional Neural Networks

**GUILHERME APOLINARIO SILVA NOVAES, (Graduate Student Member, IEEE),
AND HAE YONG KIM**

Departamento de Engenharia de Sistemas Eletrônicos, Escola Politécnica, Universidade de São Paulo, São Paulo 05508-010, Brazil

Corresponding author: Guilherme Apolinario Silva Novaes (g.novaes@usp.br)

ABSTRACT In this work, we investigate how to increase the resolution of color halftone images using convolutional neural networks (CNNs). As far as we know, this is the first work that increases resolution of color halftone images using a CNN-based solution. For this task, we first train the well-known Enhanced Deep Super-Resolution (EDSR) network with halftone images to obtain the Halftone-EDSR model. We argue that it is not possible to use conventional data augmentation techniques in this problem, due to the peculiar texture of halftone images. We present cropping as a viable data augmentation technique. Using cropping and image patches as training samples, we substantially speed up the training and get better quality models. We compare the independent channel model (which increases the resolution of each of the CMYK channels independently and then merges them) with the joint channel model (which increases the resolution of all four image channels at once) and conclude that the the latter is superior to the first. We experimentally demonstrate that the proposed Halftone-EDSR is superior to all previous techniques, both for pre-print and post-print halftone images. However, Halftone-EDSR can generate upsampled images with Moiré patterns. To minimize Moiré patterns, we propose a new network model called Halftone-Net. We use the Fast Fourier Transform, followed by a CNN, to detect the strong presence of Moiré patterns in halftone images and demonstrate that Halftone-Net generates fewer images with strong Moiré patterns than Halftone-EDSR.

INDEX TERMS Halftone, convolutional neural networks, deep learning, up-sampling, super-resolution.

I. INTRODUCTION

To print a color image, it is usually converted from the RGB (red, green, blue) color space to the CMYK (cyan, magenta, yellow, black) ink color space before printing. Each of the four image channels is then transformed into a binary halftone image consisting of clusters of dots of different sizes. The goal is to reproduce the original color image when these four binary halftone images are printed with the primary ink colors and are observed from a distance.

In this work, we investigate how to increase the resolution of color halftone images using convolutional neural networks (CNNs). There are several works that increase resolution of continuous-tone color images using CNNs but, to the best of our knowledge, this is the first work that uses CNN to increase the resolution of color halftone images.

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang.

A. RELATED WORKS

The literature on super-resolution of halftone images by machine learning (both grayscale and color) is scarce.

Kim [1] applies windowed zoom decision trees (WZDT) to upscale monochrome binary halftone images, achieving a mean absolute error (MAE) of 1.111% when doubling the resolution of test halftone images generated by HP LaserJet driver's ''coarse dots'' option, using an 8×8 sliding window. However, larger windows, such as those used in CNNs, can yield even smaller errors. Also, this study focuses on pre-print monochrome halftones and does not address post-print or color halftones. Pre-print halftone is a binary image generated by a halftoning algorithm that has not gone through the print-scan process. Post-print halftone is a continuous-tone image obtained by printing and scanning pre-print halftone.

Dong et al. [2] propose a neural network architecture for up-sampling continuous-tone images. They argue that CNNs better preserve details during resampling compared
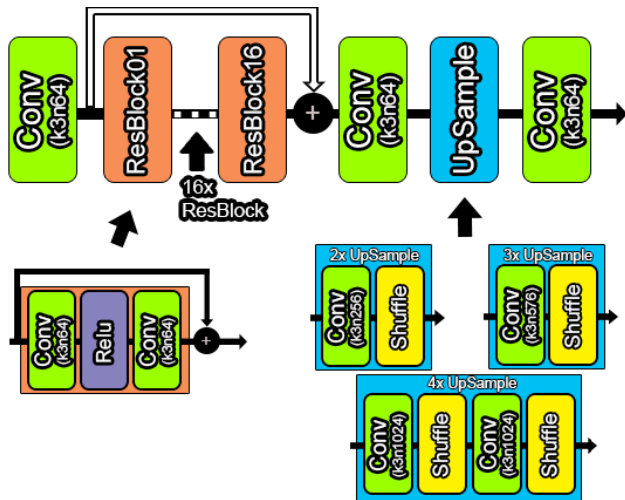
FIGURE 1. EDSR architecture to increase the image resolution. Convolutional layers are annotated with "k" representing the kernel size and "n" representing the number of filters.

to conventional techniques such as bilinear or bicubic interpolations.

Frank et al. [3] improve halftoning for low-resolution print engines with machine learning techniques. Jiang et al. [4] introduce an efficient halftoning technique using deep reinforcement learning that produces high-quality halftones while being computationally efficient. However, these approaches are focused on converting grayscale images into high-quality halftone images rather than enhancing halftone resolution.

Guo et al. [5] propose a self-supervised learning model for classifying scanned dithering halftone images. They address classification without a large set of labeled data, but the goal of this work is not resolution increasing.

Shao et al. [6] used vision transformer for inverse halftoning, producing promising results. Huajian et al. [7] present an inverse halftoning method using an invertible neural network. Li et al. [8] presents a review of the main inverse halftoning methods. However, our work is about increasing halftone resolution and not inverse halftoning.

Lim et al. [9] propose the Enhanced Deep Super-Resolution network (EDSR), a CNN architecture specifically designed to enhance continuous-tone image resolution while preserving fine details. It is based on the ResNet architecture [10] and utilizes residual blocks. EDSR incorporates additional up-sampling blocks using 2D convolutions and the depth to space layer (pixel shuffle). The model employs one of three types of output blocks to increase resolution by factors 2, 3 or 4 (see Fig. 1).

B. CONTRIBUTIONS OF THIS WORK

We published a preliminary version of this work at a conference [11], but the present work presents several aspects that were not described there. In our previous work, we trained the well-known enhanced deep super resolution

(EDSR) model [2] on single-channel halftone images to increase the resolution. Let us call the EDSR trained on halftone images Halftone-EDSR, to distinguish it from the conventional EDSR network trained on continuous-tone images. We consistently observed that Halftone-EDSR generated smaller errors than WZDT. In this article, we expand our previous research to upscale color halftone images. This work has many additional contributions:

1) To the best of our knowledge, this is the first work that increases the resolution of color halftone images using CNNs. We propose to use EDSR trained on halftone images (Halftone-EDSR) for this task.

2) There are two ways to use Halftone-EDSR to upscale color halftone images:

   a) *Independent-channel:* Use 4 monochrome Halftone-EDSR models to upscale each channel independently and then merge the 4 results to get the final upscaled color image;

   b) *Joint-channel:* Train a model that increases the resolution of the 4 channels at the same time.

   We experimentally demonstrate that better results are obtained when all channels are processed jointly by the network.

3) Data augmentation is essential to avoid overfitting and to obtain better performance in image classification [12] and segmentation [13] using CNN. In halftone resolution increasing, most data augmentation image distortions (such as random rotation, translation, resizing, etc.) cannot be used due to the peculiar nature of halftone dot patterns. We demonstrate experimentally that random cropping is an effective way of data augmentation for halftone image upscaling, and the performance of the model increases considerably. We also show that the training time becomes considerably shorter using cropping as data augmentation.

4) The upscaled halftone images may present strong Moiré patterns. To reduce Moiré patterns in the resulting images, we modify Halftone-EDSR to obtain a new convolutional neural network architecture that we call Halftone-Net.

5) We use a technique based on Fast Fourier Transform (FFT), followed by a CNN, to measure the presence of strong Moiré patterns in a halftone image. We use this technique to show that Halftone-Net generates fewer images with a strong Moiré pattern than Halftone-EDSR.

C. STRUCTURE OF THE ARTICLE

This work is organized into the following sections. Section "II. Super-resolution of halftone color images" describes the basic concepts used in this work. Section "III. Experiment settings" explains the details of our experiments. Section "IV. Halftone-EDSR in color images" explains our basic halftone upsampling model. Section "V. Improving Halftone-EDSR" addresses limitations faced by the basic

model, including generation of undesirable Moiré patterns. Section "VI. Halftone-Net" introduces a new model that mitigates generation of Moiré patterns.

Both Halftone-EDSR and Halftone-Net models are available at Google Colab.[1]

II. SUPER-RESOLUTION OF COLOR HALFTONE IMAGES

A. TRADITIONAL HALFTONING

Before digital imaging, special photographic techniques were used to break down grayscale images into halftone dots. One of these was "screening", where a coarse-woven fabric screen was suspended in front of the camera plate, breaking the incoming light into a pattern of dots. In fact, the term "lines per inch" (LPI) seems to derive from the thickness of the fabric screen used in the halftoning process. Color halftone images were generated repeating this process for primary colors.

In this work, we train and test our algorithm on old comic book images that were printed using traditional halftoning techniques, to test its ability to upsample post-print halftones.

B. DIGITAL HALFTONING

Digital halftone images can be categorized into three main types:

- *Clustered-dot dithering* arranges clusters of dots in a regular grid pattern, with varying sizes and patterns to create different tones. This amplitude modulation technique is commonly used in electro-photographic printers (e.g., laser and LED printers) that cannot print isolated dots. It is also widely employed in commercial printing for magazines and newspapers using offset printing.
- *Dispersed-dot dithering* disperses tiny dots in a regular pattern without forming clusters. Through frequency modulation, different dot densities create the illusion of various shades or colors. This method is only applicable to printers capable of printing isolated dots, such as inkjet or matrix printers.
- *Error diffusion algorithms* distribute quantization errors caused by color approximation to neighboring pixels. This frequency modulation technique propagates the quantization error from each pixel to its neighbors, effectively spreading it throughout the image. Similar to dispersed-dot dithering, error diffusion can only be used by printers capable of printing isolated dots. It is very difficult to upsample error diffusion halftone images using machine learning algorithms due to their chaotic nature.

We tested our algorithm on pre-print halftones generated by clustered-dot dithering algorithm (Fig. 2), because this is the most widely used technique.
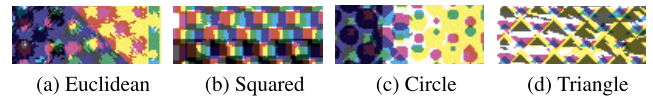

FIGURE 2. Examples of color clustered-dot dithering halftones.

(a) Euclidean (b) Squared (c) Circle (d) Triangle

C. SPATIAL RESOLUTION

The quality of the image observed at a distance is influenced by halftoning parameters. The size and spacing of dots significantly impact the quality of the image [14], are referred to as spatial resolution and encompasses two concepts:

- *Dots per inch* (DPI) represents the density of tiny ink dots that a printer can produce within an inch. These dots can be either on (indicating ink deposition) or off (no ink deposition). DPI is a hardware-based metric used to measure a printer's resolution capability.
- *Lines per inch* (LPI) measures the number of clusters of dots printed within one inch. These halftone dots, which can vary in size and shape, create the illusion of different color shades on paper [14], [15]. Halftone algorithms generate these clusters of dots and arrange them into "lines". LPI is a user-adjustable software metric. Fig. 3 provides a visual comparison of a continuous-tone color image and its corresponding halftone versions at 75, 150 and 300 LPI.

D. PRE- AND POST-PRINT HALFTONE

Halftone images can be categorized into two types: pre-print and post-print (Fig. 4).

- A *pre-print* halftone is a binary image (or a set of binary images) generated by a halftone algorithm before the printing and scanning process. It can be either a monochrome binary image created from a grayscale image or a set of four binary images representing the CMYK channels created from a color image.
- A *post-print* halftone is obtained by printing a pre-print halftone onto paper and scanning it back. Due to limitations in the mechanical and optical processes [16], the resulting post-print halftone is a continuous-tone image, which can be single channel or color.

E. MOIRÉ PATTERN

Modern color printer halftoning techniques aim to accurately reproduce colors while accounting for imperfections in colorants and complex dot interactions [17].

Ensuring high-quality color halftone images also involves preventing unwanted noises on image, such as Moiré patterns, visible at specific viewing distances. Moiré patterns emerge due to interactions between different frequency components in an image. These patterns can arise from overlapping angles of different image channels (Fig. 5a) or during the down-sampling of specific image textures.

To minimize Moiré patterns in clustered dot dithering, we can select different and appropriate dithering angles for each CMYK color channel during halftoning [14], [18].
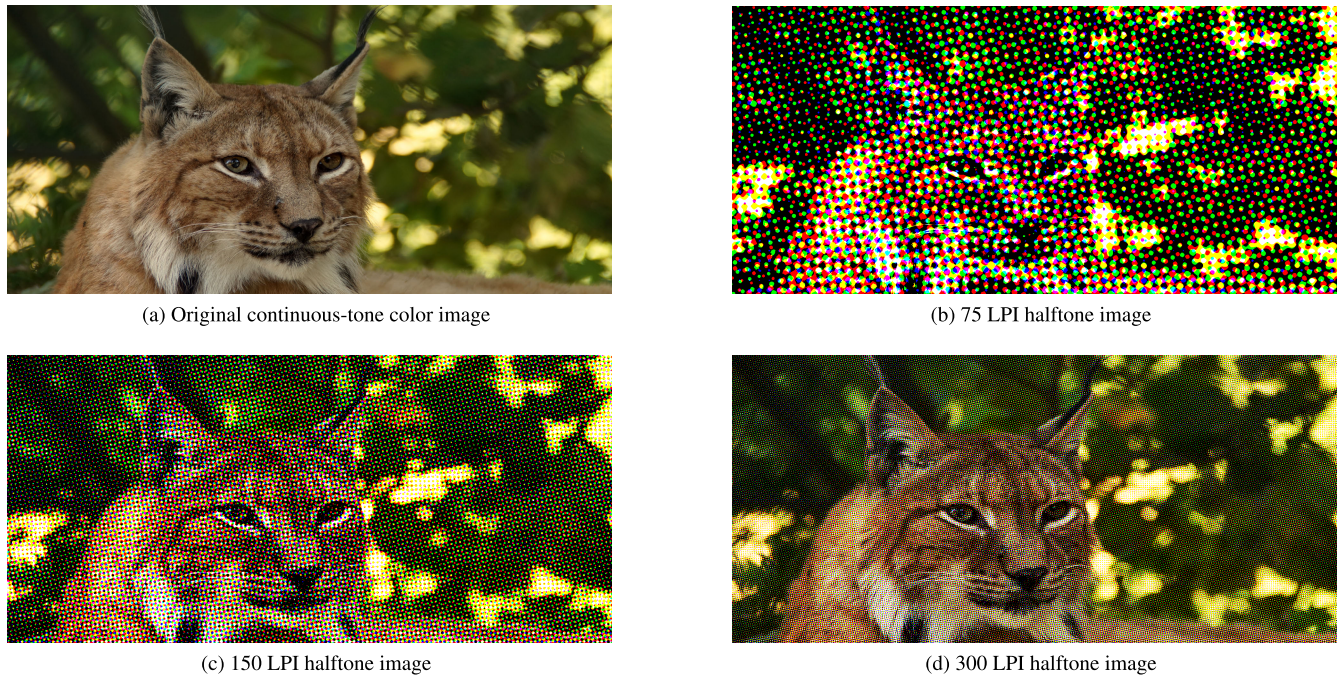
---

[1]https://colab.research.google.com/drive/1EiLuiss3e7Lctf0fFf1oIevAPKqUMeHr

(a) Original continuous-tone color image

(b) 75 LPI halftone image

(c) 150 LPI halftone image

(d) 300 LPI halftone image

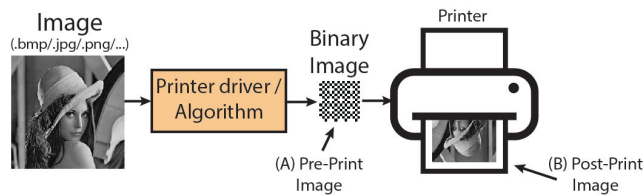**FIGURE 3.** Halftones of a color image in different LPIs.

FIGURE 4. A pre-print halftone (A) is a binary image before print-scan process. A post-print halftone images (B) is the image obtained printing pre-print halftone and scanning it back.

Moiré pattern can occur when an image is captured from a screen (Fig. 5b) or a halftone is scanned [18] (Fig. 5c). Moiré pattern in scanned halftone image is typically caused by interaction between the halftone screen frequency (LPI) of the printed image and the scanning resolution (DPI) of the scanner [19]. Moiré patterns can also be generated when a document viewing program displays a halftone image on a computer screen. Thus, new Moiré patterns may appear when viewing the figures in this paper on a computer screen.

III. EXPERIMENT SETTINGS
A. DATASET ORGANIZATION
In our experiments, we use two datasets: D_A and D_B. The D_A dataset contains pre-print images obtained by converting the original RGB images to CMYK and then running a halftone algorithm to generate a CMYK halftone image comprised of 4 binary halftone images. The D_B dataset contains post-print images, with different LPIs, scanned from old comic books.

1) DATASET D_A - SYNTHETIC PRE-PRINT HALFTONES
We used the DIV2K dataset [21] with 900 RGB color images to create the synthetic pre-print halftone dataset D_A. DIV2K is one of the most used datasets to increase the resolution of high definition images.

These images were converted to grayscale (G) and CMYK (G^4) color representations. Then, they were rotated 90 degrees and cropped to 2000×1124 pixels.

We employed area-based interpolation to downscale each G and G^4 image by a factor of 2, resulting in $G_{0.5}$ and $G_{0.5}^4$, respectively. Using the Python halftone library [22], we transformed the images G, G^4, $G_{0.5}$ and $G_{0.5}^4$ into binary halftone images denoted as B, B^4, $B_{0.5}$ and $B_{0.5}^4$. For color images, the halftone algorithm was independently applied to each CMYK channel using clusters of type "Euclidean" (Fig. 2a). The rotation angles were: 15 degrees for cyan, 75 degrees for magenta, 90 degrees for yellow and 45 degrees for black. If the images B, B^4, $B_{0.5}$ and $B_{0.5}^4$ of this dataset were printed on a printer by converting an image pixel to a printer dot, all images would have the same LPI.

We utilized ($B_{0.5}$, B) pairs in single-channel upscaling experiments to find the appropriate patch and batch sizes, and ($B_{0.5}^4$, B^4) pairs for training and testing four-channel upscaling models (both independent-channel and joint-channel models).

2) DATASET D_B - SCANNED POST-PRINT HALFTONES
We used comics from the Digital Comic Museum [23] to construct the post-print halftone dataset D_B. These images display a variety of halftone shapes, lines and elements with constant tone (Fig. 6). We randomly selected 900 images from the comics database, ensuring a maximum of two pages per comic to maintain the dataset unbiased.
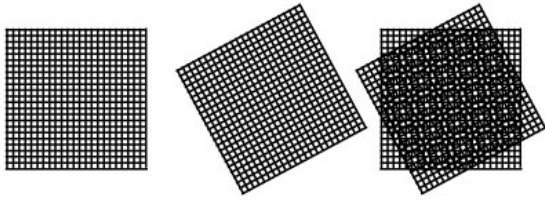
(a) Moiré patterns (patterns similar to circles in the image at right) generated by overlapping two grid layers with different angles (source [20]).
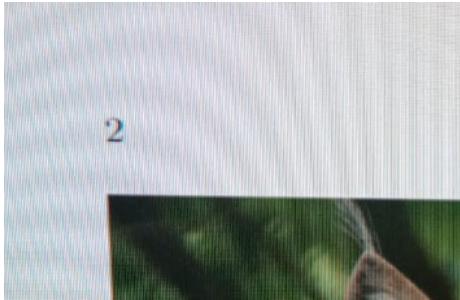
(b) Moiré patterns (dark lines and curves, more visible in the light area of the image) generated when the computer screen was captured by a mobile camera.

(c) Moiré patterns generated by scanning a halftone image.

**FIGURE 5.** Moiré patterns generated by different processes.

(a) Daffy Tunes Comics 012, p. 7.

(b) Elsie the Cow 001, p. 9.

FIGURE 6. Examples of images in D_B dataset.

The original RGB images were converted to grayscale to yield grayscale halftone images, designated as B. The RGB images were also transformed into the CMYK color space, generating halftone color images referred to as B^4. We downscaled the B and B^4 images to create smaller versions named $B_{0.5}$ and $B_{0.5}^4$.
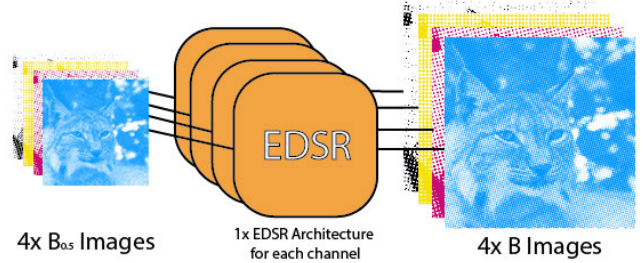
(a) Independent-channel.
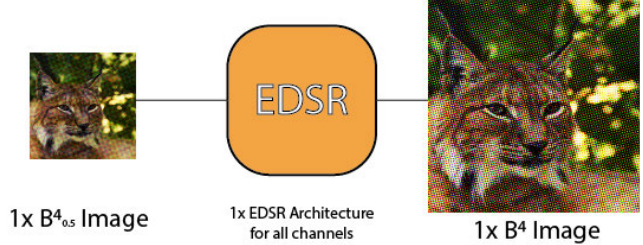
(b) Joint-channel.

FIGURE 7. We tested two types of halftone upsampling systems: a) Independent-channel that processes 4 channels independently. b) Joint-channel that processes 4 channels at the same time.

As this dataset is a post-print dataset, the halftone images B, B^4, $B_{0.5}$ and $B_{0.5}^4$ are continuous-tone, rather than binary. The halftone images $B_{0.5}$ and $B_{0.5}^4$ have half the lines per inch (LPI) compared to B and B^4. As before, we utilized ($B_{0.5}$, B) pairs to find the appropriate patch and batch sizes. We used ($B_{0.5}^4$, B^4) pairs for training and testing four-channel upscaling models.

B. INDEPENDENT AND JOINT CHANNELS

We evaluated two types of systems for increasing the resolution of color halftone images: independent-channel and joint-channel (see Fig. 7).

The *independent-channel system* processes the four color channels (CMYK) separately. We split a pair of CMYK halftone images ($B_{0.5}^4$ and B^4) into four pairs of monochrome halftone images: ($B_{0.5}^C$, B^C), ($B_{0.5}^M$, B^M), ($B_{0.5}^Y$, B^Y) and ($B_{0.5}^K$, B^K). These pairs were used to train four models N^C, N^M, N^Y and N^K, which could be either decision tree- or neural network-based models. It is not possible to use a single model for the four channels, because each channel has a different rotation angle. To up-sample a CMYK test image, we independently processed the four monochrome halftone images using their respective models. We combined the four channels to obtain the final output color halftone image. The mean absolute error (MAE) was calculated in the CMYK color space before converting the image to the RGB color space.

In the *joint-channel system*, a single model N processed the four channels jointly. We trained the model using 4-channel halftone images $B_{0.5}^4$ and B^4 from datasets D_A and D_B. After training, when a 4-channel halftone test image was given, the network N directly generated the upscaled 4-channel image.

C. DATA AUGMENTATION

Data augmentation is a common technique used to increase the diversity and quantity of training data in image-based machine learning systems. The common data augmentation techniques include geometric transformations (rotation, translation, resizing, etc.), flipping, color space transformations, cropping and noise injection [12], [13]. These methods enhance model performance and generalization by introducing diversity in the training data.

However, most distortions used for image data augmentation cannot be used in halftone images due to the unique nature of halftone patterns. Halftone images rely on clusters of dots to form the smallest unit of representation, and any alteration introduced by geometric transformations, color space changes or noise injection may alter the dot format and disrupt the integrity of the halftone pattern. Flipping and rotation also cannot be used as they affect the specific angles and orientations of the halftone dots. Furthermore, translation can lead to distorted halftone dots or introduce halftone-free background.

Fig. 8 illustrates how improper data augmentation can compromise the structure of halftone.

Given these limitations, cropping emerges as the only viable data augmentation for halftone images. Cropping patches from the entire image allows us to retain essential aspects of the halftone pattern while introducing variability by selecting different sections of the image. The random cropping of images during the training works as a form of regularization [24] that helps to prevent overfitting and to improve the generalization capabilities of the model.

Super-resolution networks, including EDSR, generally do not have dense layers. This feature allows to feed these networks with smaller images during training, as well as to upscale images of any size during prediction. The only potential concern in cropping is breaking halftone dots at the edges of the image. However, cropped dots are commonly found at the edges of halftone images.

Super-resolution models employing Vision Transformers such as HAT [25], SwiniR [26] use patch-based approach to extract image features. As shown in Fig. 9, the HAT model generates images with impressive visual quality, although the outputs have little texture, particularly evident in Fig. 9b where the result appears to be a vectorized version of the original image. Although HAT achieves high quality when upscaling regular images, it falls short when applied to halftones, failing to retain their unique textures.

Preserving textures is a priority in halftone images. The dimensionality reduction in embeddings operates similar to a low-pass filter, removing noise and texture, a phenomenon resembling the behavior of auto-encoders [27].

D. TESTING ENVIRONMENT

All executions were carried out on a machine with an Intel Core i7-13700K, 128GB of RAM and NVidia GeForce 4090 RTX with 24GB of VRAM, in Linux Mint 21.1, using the Keras framework.
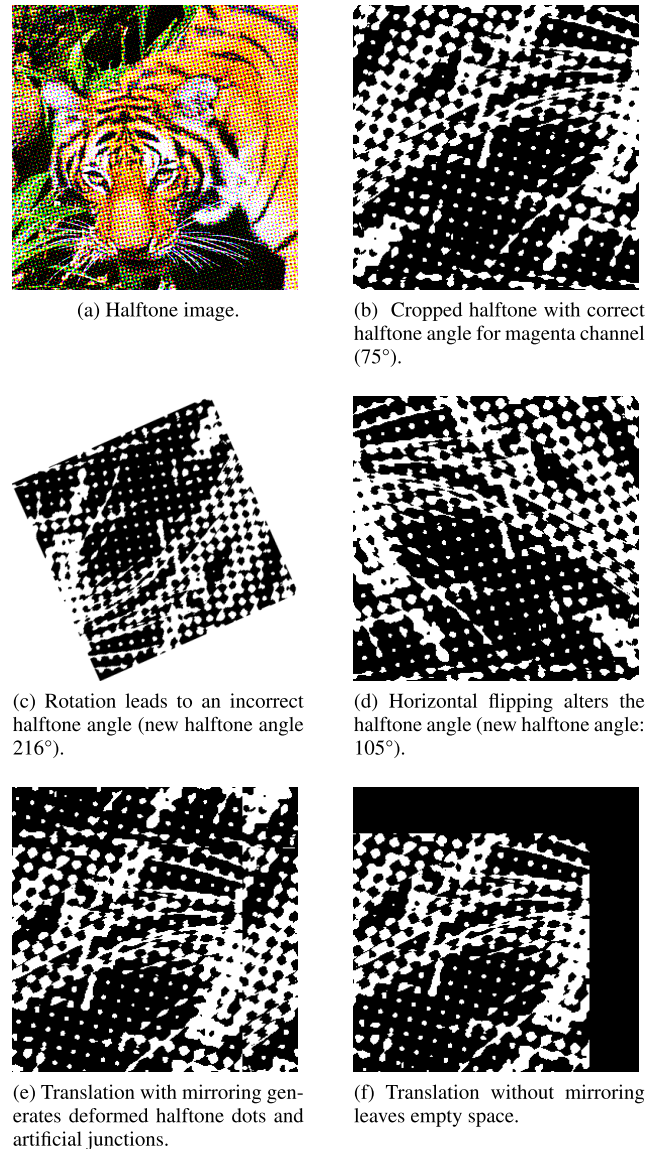

(a) Halftone image.

(b) Cropped halftone with correct halftone angle for magenta channel (75°).

(c) Rotation leads to an incorrect halftone angle (new halftone angle 216°).

(d) Horizontal flipping alters the halftone angle (new halftone angle: 105°).

(e) Translation with mirroring generates deformed halftone dots and artificial junctions.

(f) Translation without mirroring leaves empty space.

FIGURE 8. Examples of how improper data augmentation can compromise the structure of halftone. For clarity, only the magenta channel is depicted.

E. CHOOSING THE PATCH AND BATCH SIZES

In our previous work [11], we used whole images to train the model without any type of data augmentation. Using a batch size of 2 and full 2000×1124 images, training a single model took approximately 12 hours. To reduce the processing time and to augment data, in this work we train the network with image patches instead of whole images (Fig. 10).

During each training epoch, we crop images by randomly choosing patches of a predefined size. As a result, rather than providing the entire page as input, the model receives a random sample from the page.

Experimental results with various patch sizes for monochrome halftone images are presented in Table 1 for dataset D_A and Table 2 for D_B. The best results were achieved with a 256×256 patch size and a batch size of 2. Training the

Input   Real_HAT_GAN Output

(a) Real picture.

Input   Real_HAT_GAN Output
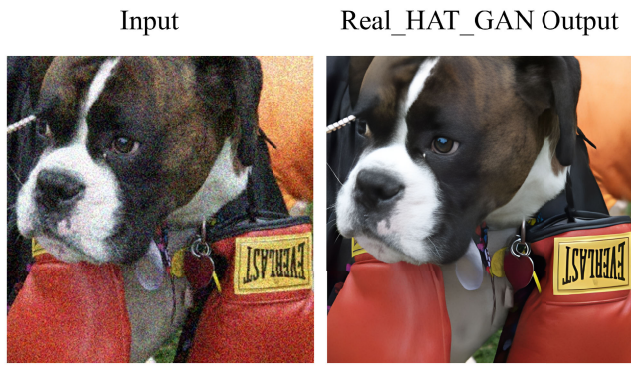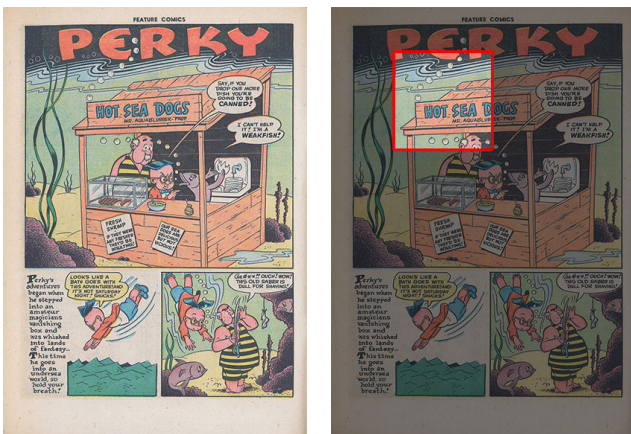
(b) Cartoon picture.

FIGURE 9. Quality of images upsampled by HAT. Left: original images. Right: images reconstructed by HAT. (source [28]).

(a) Full Page   (b) Patch Area

FIGURE 10. Illustration showing a full page and a 256×256 patch used in this work. Feature Comics 122, p. 20.

model with patches only took 1 hour, compared to 12 hours using full 2000×1124 images.

F. 5-FOLD CROSS VALIDATION

To evaluate our system, we used a modified version of 5-fold cross validation. The dataset with 900 pairs of images was divided into 5 subsets, each containing 180 pairs. In each fold, one subset was used for testing, while the remaining 4 subsets (720 pairs) were used for training. Out of the 720 training

TABLE 1. Experiments to choose the best batch and patch sizes using monochrome images from D_A. "MAE ± SD" indicates the mean and standard deviation of 900 MAEs between the processed and the ideal images. We executed 5 training folds with 180 results in each fold.

Batch size	Patch size	MAE ± SD	Time
2	32×32	0.0318 ± 0.0118	20min
1	64×64	0.0298 ± 0.0108	40min
2	64×64	0.0324 ± 0.0132	30min
4	64×64	0.0320 ± 0.0124	25min
32	64×64	0.0365 ± 0.0112	35min
2	128×128	0.0314 ± 0.0123	45min
2	256×256	**0.0289 ± 0.0110**	1h
2	Full page	0.0290 ± 0.0144	12h

pairs, 630 were used for actual training and 90 were used for validation. The trained model was then applied to the test subset to calculate the MAE between the output and ideal images. We repeated this process for 5 folds, resulting in 900 error measurements. Finally, we calculated the mean and standard deviation of these errors.

G. OPTIMIZER AND EPOCHS

We used the Adam optimizer with a learning rate ranging from 10^{-4} to 5×10^{-5}, implementing a piecewise constant decay function with a boundary parameter of 5000. The models underwent 100 epochs of training.

IV. HALFTONE-EDSR IN COLOR IMAGES

We used EDSR model with configuration similar to described in [9] and [11]. This model consisted of 16 residual blocks, using the Mean Absolute Error (MAE) with a single up-sampling layer for doubling the image resolution. We modified the depth of the input layer of the EDSR from three (in the original network to receive RGB images) to one for systems with independent channels and to four for systems with joint channels. We call Halftone-EDSR the EDSR model trained on halftone images to up-sample halftone images. We tested three versions of Halftone-EDSR for color images:

1) Full-independent: Trained on full-pages, each channel is processed independently.
2) Patch-independent: Trained on image patches, each channel is processed independently.
3) Patch-joint: Trained on image patches, a single model processes all four channels jointly.

We trained the models using the optimal batch and patch sizes determined in the previous section for monochrome halftones. The results show the superiority of the "patch-joint" approach compared to the "full-independent" and "patch-independent" methods, both for the D_A and D_B datasets (Tables 3 and 4 and Fig. 11).

A. COMPARING HALFTONE-EDSR WITH WZDT IN D_A

We compared the best Halftone-EDSR approach (patch-joint) with WZDT independent-channel algorithm trained on full pages or image patches to upscale synthetic color halftones in D_A. We did not test joint-channel WZDT, because WZDT is designed to upscale monochrome images.

TABLE 2. Experiments to choose the best batch and patch sizes using monochrome images from D_B.

Batch size	Patch size	MAE ± SD	Time
2	32×32	0.0332 ± 0.0121	20min
1	64×64	0.0321 ± 0.0111	40min
2	64×64	0.0342 ± 0.0116	30min
4	64×64	0.0361 ± 0.0102	25min
32	64×64	0.0384 ± 0.0151	35min
2	128×128	0.0339 ± 0.0139	45min
2	256×256	**0.0311 ± 0.0120**	1h
2	Full page	0.0360 ± 0.0141	12h

TABLE 3. Upscaling CMYK color halftone images in D_A dataset.

Input	Channels	MAE ± SD
Full page	Independent	0.037 ± 0.012
Patch	Independent	0.035 ± 0.014
Patch	Joint	**0.022 ± 0.008**

TABLE 4. Upscaling CMYK color halftone images in D_B dataset.

Input	Channels	MAE ± SD
Full page	Independent	0.068 ± 0.020
Patch	Independent	0.027 ± 0.011
Patch	Joint	**0.023 ± 0.01**

(a) Synthetic data D_A

(b) Comic images D_B

FIGURE 11. Box plots of the results presented in (a) Table 3 and (b) Table 4.

TABLE 5. Comparing WZDT with Halftone-EDSR on D_A color halftone images.

Model	Channels	MAE ± SD
WZDT (full)	Independent	0.063 ± 0.015
WZDT (patch)	Independent	0.062 ± 0.016
Halftone-EDSR (patch)	Joint	**0.022 ± 0.008** (Table 3)

All these methods were adapted to generate binary images as outputs. Each model underwent 900 experiments, and the average mean absolute error and standard deviation are presented in Table 5. We trained the WZDT models using the same parameters as in [1].

The patch-joint Halftone-EDSR is the most effective method for upscaling synthetic color halftones in D_A, with an MAE of 0.022±0.008. WZDT algorithms present far greater error rates and inferior image quality.
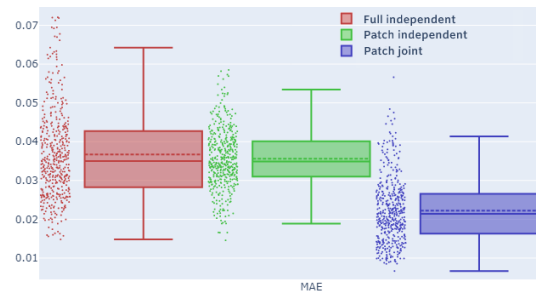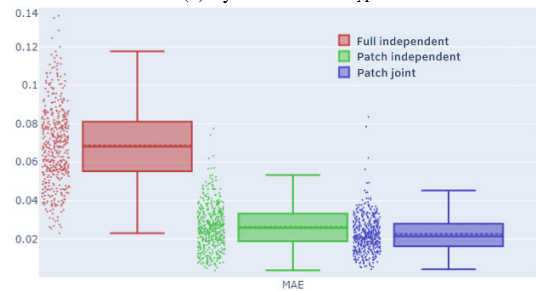
B. COMPARING HALFTONE-EDSR WITH CONVENTIONAL RESAMPLING IN D_B

Four conventional non-machine learning techniques (nearest neighbor, bilinear, bicubic and Lanczos) were employed to upscale the D_B dataset. These techniques were compared to patch-joint Halftone-EDSR (Table 6). All conventional resampling techniques have much higher error rates than Halftone-EDSR. Fig. 13 illustrates that conventional methods produce blurry results. Conventional interpolation techniques were not employed to upscale dataset D_A because they produce non-binary output images.

C. COMPARING HALFTONE-EDSR WITH MODELS TRAINED WITH CONTINUOUS-TONE IMAGES IN D_B

Table 7 evaluates three machine-learning joint-channel models trained on continuous-tone images: ESRGAN [29], StyleGAN2 [30] and EDSR [9]. We used them to upscale images in dataset D_B. The error rates of all machine learning techniques trained on continuous tone images were significantly higher than that of Halftone-EDSR.

Fig. 13 demonstrates that Halftone-EDSR preserves halftone texture and produces high-quality upscaled images.

GAN-based techniques often produce output images similar to those produced by inverse halftoning [8], because they interpret halftone texture as noise, resulting in "clean" images without texture (Fig. 13). ESRGAN ignores halftone texture, while StyleGAN2 retains some texture but generates artifacts. In contrast, the image generated by Halftone-EDSR exhibits few artifacts and effectively preserves fine details.

Properly selecting training images is crucial to obtain successfully upscaling models. This becomes evident when we compare the Halftone-EDSR trained on halftones with the EDSR trained on continuous tone images (Fig. 13).

V. IMPROVING HALFTONE-EDSR
A. IMPERFECTIONS AND MOIRÉ PATTERNS IN D_B

The D_B dataset contains a multitude of imperfections (Fig. 14), including folds, paper marks, misalignment between CMYK channels, variations in paper quality, as well as different Lines Per Inch (LPI) and Dots Per Inch (DPI) in prints.

As depicted in Fig. 14, the Moiré patterns captured during the scanning further contribute to the imperfection of the dataset [19]. Moiré patterns are commonly found in scanned halftone images due to the resampling process that occurs when converting analog images (printed on physical paper) to digital [18]. These patterns arise due to
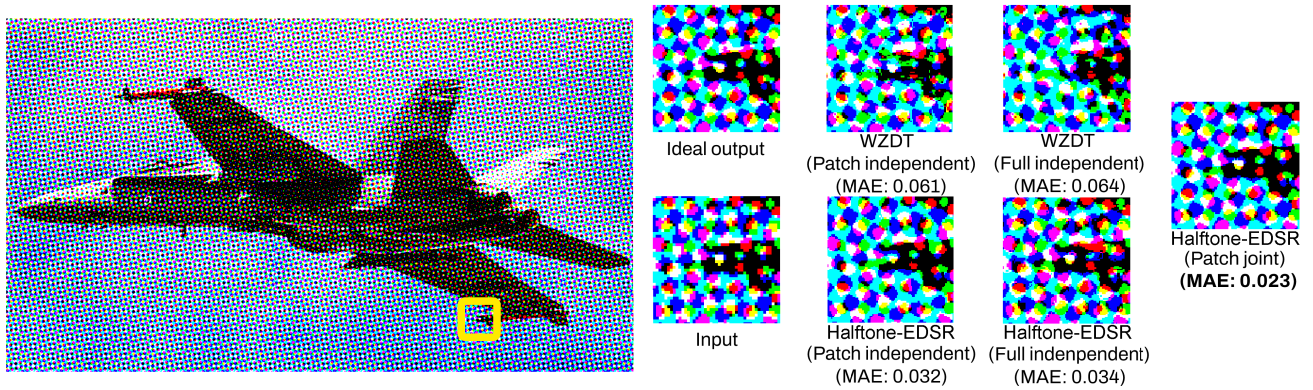
FIGURE 12. Visual comparison of upscaling pre-print halftone images D_A with Halftone-EDSR and other methods.
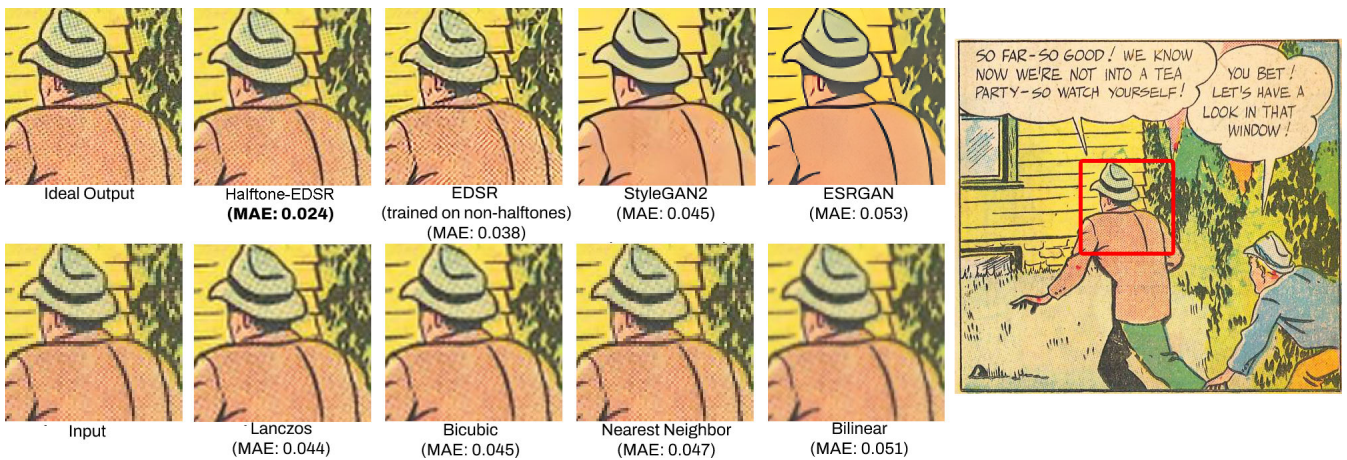

FIGURE 13. Visual comparison of post-print halftone images D_B upscaled with Halftone-EDSR and other methods.

TABLE 6. Comparison of Halftone-EDSR with non-machine learning methods to upscale D_B.

Algorithm/Model	Channels	MAE ± SD
Nearest neighbor	Independent	0.048 ± 0.021
Bilinear	Independent	0.050 ± 0.023
Bicubic	Independent	0.046 ± 0.022
Lanczos	Independent	0.045 ± 0.014
Halftone-EDSR (patch)	Joint	**0.023 ± 0.01** (Table 4)

TABLE 7. Comparison of Halftone-EDSR with other joint-channel machine learning techniques trained on continuous-tone color images to upscale D_B.

Model	Training input	MAE ± SD
ESRGAN	Non-halftone, full images.	0.052 ± 0.03
StyleGAN2	Non-halftone, full images.	0.044 ± 0.03
EDSR	Non-halftone, patches.	0.033 ± 0.02
Halftone-EDSR	Halftone D_B, patches.	**0.023 ± 0.01** (Table 4)

the interaction of frequency components and can obscure fine details [31]. They mostly appear in areas with light colors (Figs. 14b and 14c).

Unlike the images in the D_A dataset, where the image is first halved in size and then halftoned, the images in the D_B dataset are halved after halftoning, making Moiré patterns more noticeable.

B. INAPPROPRIATE ERROR METRIC

The analysis of images generated by Halftone-EDSR reveals an interesting phenomenon. In some cases, the model's effort to mitigate imperfections generates an image that is "better" than the ideal image in some regions (see Fig. 15). This phenomenon is likely due to the variety of irregularities present in the input training images. The machine learning model struggles to learn from diverse data, causing these irregularities to be treated as outliers. As a result, the model does not replicate these imperfections, but rather attempts to eliminate them.

Thus, in the D_B dataset, a high MAE does not necessarily mean that the model is unable to generate good quality output. If the ideal image has significant imperfections, a high MAE may suggest that the model has generated image without the same imperfections. The model would need to replicate even the imperfections present in the ideal images for the MAE to be extremely low, an undesirable scenario.

C. DENOISING

Yang et al. [19] present a study focused on reducing Moiré patterns in scanned grayscale halftone images. The method
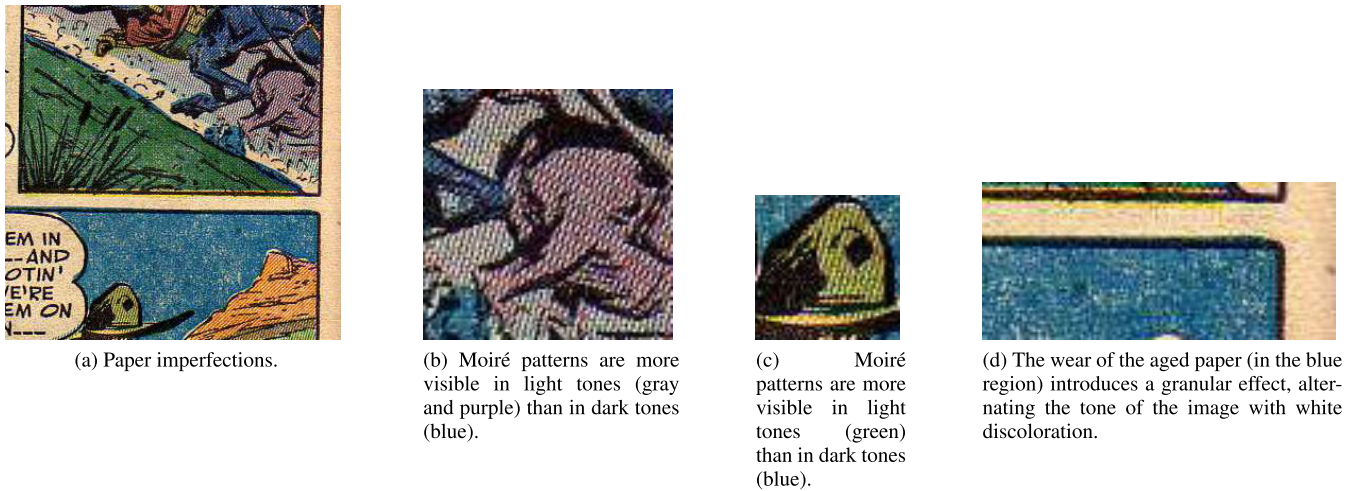
(a) Paper imperfections.

(b) Moiré patterns are more visible in light tones (gray and purple) than in dark tones (blue).

(c) Moiré patterns are more visible in light tones (green) than in dark tones (blue).

(d) The wear of the aged paper (in the blue region) introduces a granular effect, alternating the tone of the image with white discoloration.

FIGURE 14. Different types of imperfections in the D_B dataset.

(a) Ideal high-resolution image. (b) Low-resolution input image. (c) Image upsampled by Halftone-EDSR (MAE: 0.0446).

(d) Ideal high-resolution image. (e) Low-resolution input image. (f) Image upsampled by Halftone-EDSR (MAE: 0.0347).
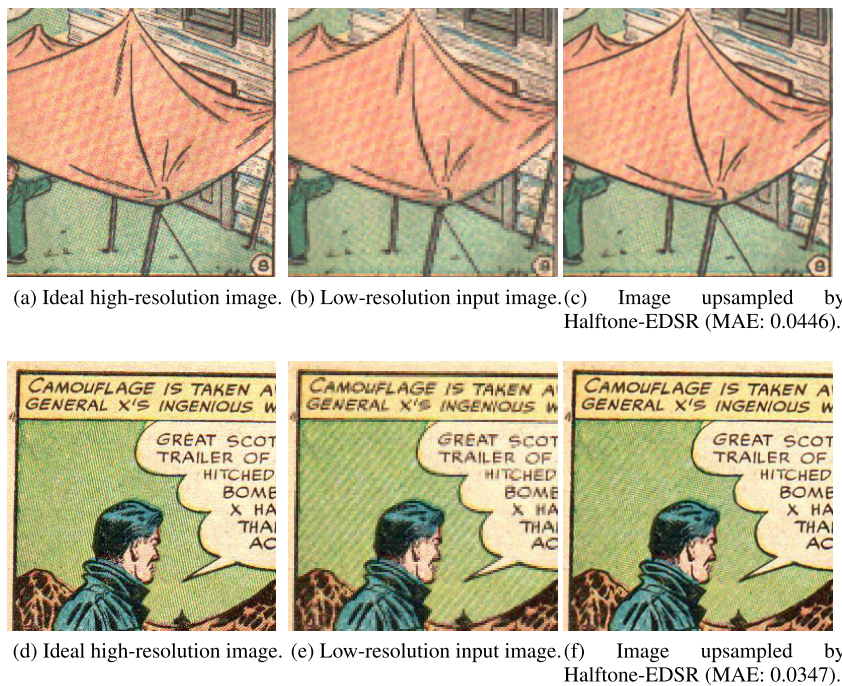
FIGURE 15. The ideal high-resolution images display strong Moiré patterns in green areas. The input low-resolution images also present Moiré patterns in green areas. The images upsampled by Halftone-EDSR have weaker Moiré patterns that differ from those of ideal images. This difference in Moiré patterns results in a high MAE between the ideal image and the Halftone-EDSR output, despite the good quality of the output image.

scans twice the same halftone image, where one of the scans is performed with the image rotated by 45 degrees. This process generates two sets of scanned Moiré patterns. Subsequently, an intersection of the two scanned images is created, resulting in a notable reduction of Moiré patterns. However, it is not possible to implement this technique on the D_B dataset because it is not possible to perform a new rotated scan.

There are two approaches that can mitigate the artifacts like Moiré patterns in the output images:

- The first approach eliminates all textures. This is undesirable because it also eliminates halftone patterns, actually leading to inverse halftoning. As evident in Fig. 13, GAN-based techniques produce results close to inverse halftone.
- The second tries to differentiate between noise and texture, resulting in a filter that mitigates noise and preserves texture. Fig. 13 illustrates the excellent results of Halftone-EDSR in improving resolution while
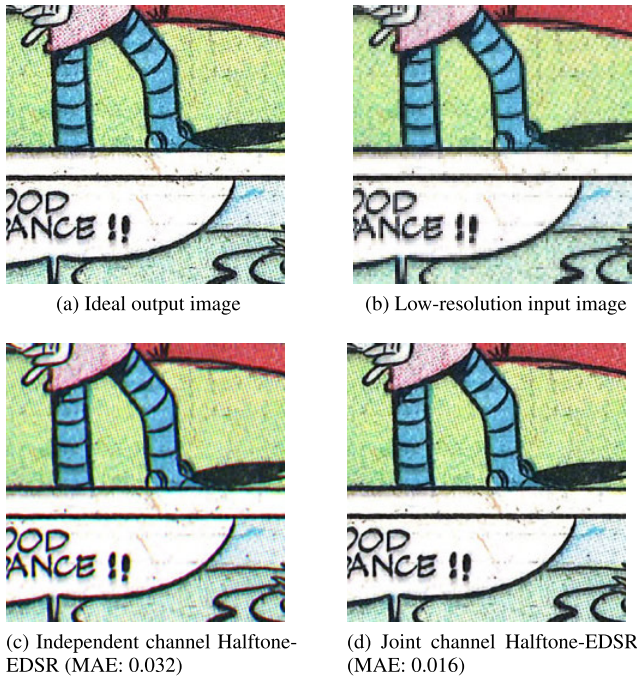
(a) Ideal output image

(b) Low-resolution input image

(c) Independent channel Halftone-EDSR (MAE: 0.032)

(d) Joint channel Halftone-EDSR (MAE: 0.016)

FIGURE 16. Independent channel Halftone-EDSR blurs halftone textures. Joint-channel Halftone-EDSR enhances halftone textures.
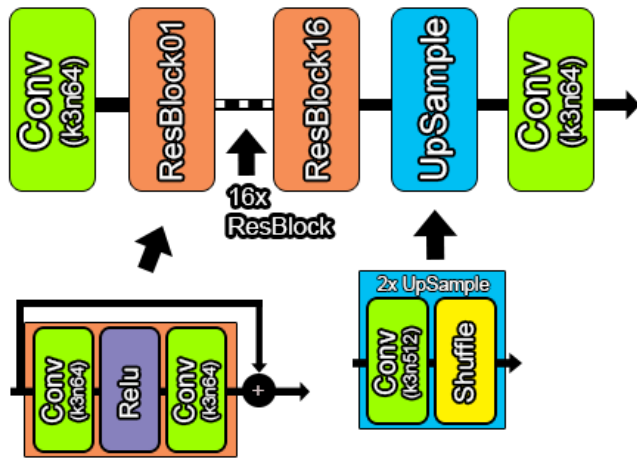

FIGURE 17. Halftone-Net architecture. Convolutional layers are annotated with "k" representing the kernel size and "n" representing the number of filters.

preserving halftone texture. The Halftone-EDSR model minimized the Moiré pattern in both orange tones (character's shirt) and greenish tones (character's hat).

There are cases where both the ideal and input images are strongly affected by the Moiré pattern (see Fig. 15). In these cases, Halftone-EDSR can reduce the Moiré pattern. However, this noise reduction may result in a higher MAE.

Methods with independent channels introduce more Moiré patterns to the output images than those with joint channels. This probably happens because of the inability of independent channel models to learn channel overlap, thus generating Moiré patterns. Fig. 16 illustrates that joint-channel
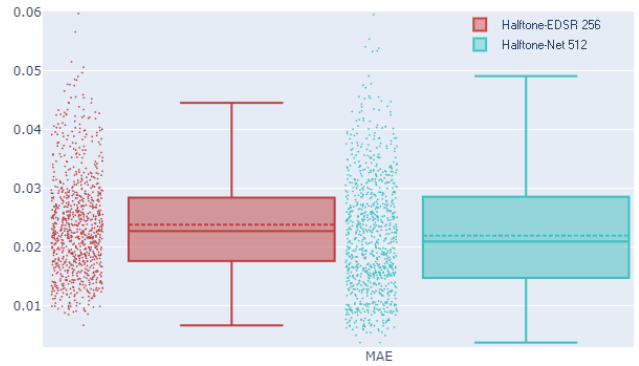

FIGURE 18. Box plots of the models (A) and (D) of Table 8.

Halftone-EDSR manages to reconstruct halftone textures more effectively than the independent-channel approach. Light green areas have halftone textures that the joint-channel model improves, while the independent-channel model blurs.

VI. HALFTONE-NET
A. INTRODUCTION
In this section, we propose a new network architecture called Halftone-Net, specially designed to increase the resolution of halftone images while reducing Moiré patterns. In the literature, there are networks specifically designed to attenuate noise [27] and Moiré patterns [31] for continuous tone images. These Moiré patterns are typically generated when photographing monitor screens, which differs from the patterns found in halftone images (Figs. 5b and 5c).

Unlike Halftone-EDSR, which adds the output of the first convolutional layer to the output of the cascaded residual blocks (white arrow at the top of Fig. 1), the proposed Halftone-Net (Fig. 17) do not use residual connection from input to output. The absence of a long residual connection minimizes the propagation of artifacts from the input image to the output. This modification retains only the 16 residual blocks present in the EDSR architecture. To avoid replicating Moiré patterns, the number of filters at the input of the upscaling layer was doubled from 256 to 512 filters. The purpose of this adjustment is to compensate for the lack of residual connection.

B. EXPERIMENTAL RESULTS
We carried out the experiments using exclusively the 900 images from D_B, as the objective of Halftone-Net is to mitigate Moiré patterns present in scanned images of the real world, a feature absent in D_A with synthetic images.

Table 8 shows the errors produced by Halftone-EDSR and Halftone-Net using 256 and 512 filters. The best Halftone-EDSR results were obtained using 256 filters (A) and the best Halftone-Net results using 512 filters (D). The box plot of models (A) and (D) are depicted in Fig. 18. Halftone-Net (D) produced slightly lower average MAE than Halftone-EDSR (A), but it is not clear whether this apparent improvement is real or mere statistical fluctuation.
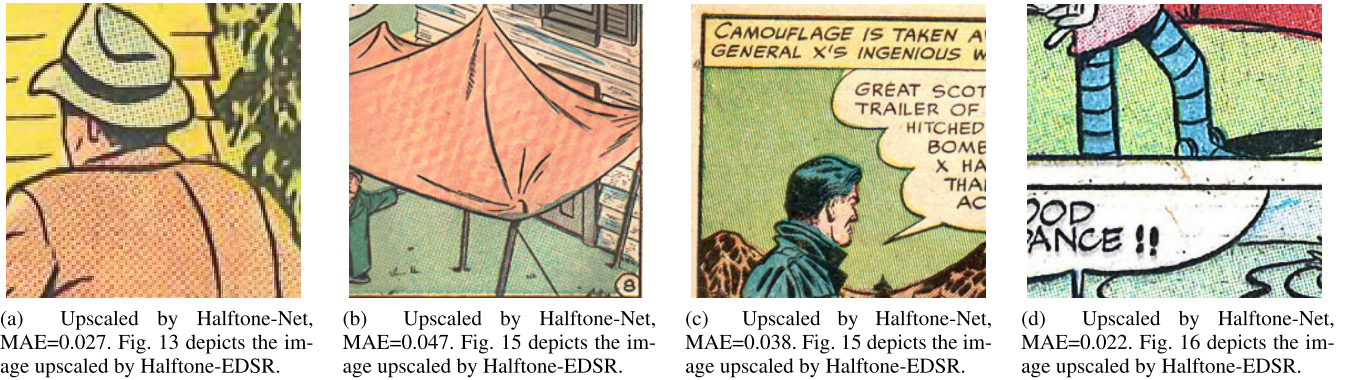
(a) Upscaled by Halftone-Net, MAE=0.027. Fig. 13 depicts the image upscaled by Halftone-EDSR.

(b) Upscaled by Halftone-Net, MAE=0.047. Fig. 15 depicts the image upscaled by Halftone-EDSR.

(c) Upscaled by Halftone-Net, MAE=0.038. Fig. 15 depicts the image upscaled by Halftone-EDSR.

(d) Upscaled by Halftone-Net, MAE=0.022. Fig. 16 depicts the image upscaled by Halftone-EDSR.

FIGURE 19. All the 4 images above were upscaled using Halftone-Net and all have a larger MAE than those generated by Halftone-EDSR. However, all of them were categorized as "weak Moiré" (high quality) while only 1 of the 4 images generated by Halftone-EDSR was categorized as "weak Moiré" (the image corresponding to the image (a) in fig. 13). We can notice that these images have fewer Moiré patterns than the Halftone-EDSR counterparts, and the halftone textures are sharper than in the ideal images.
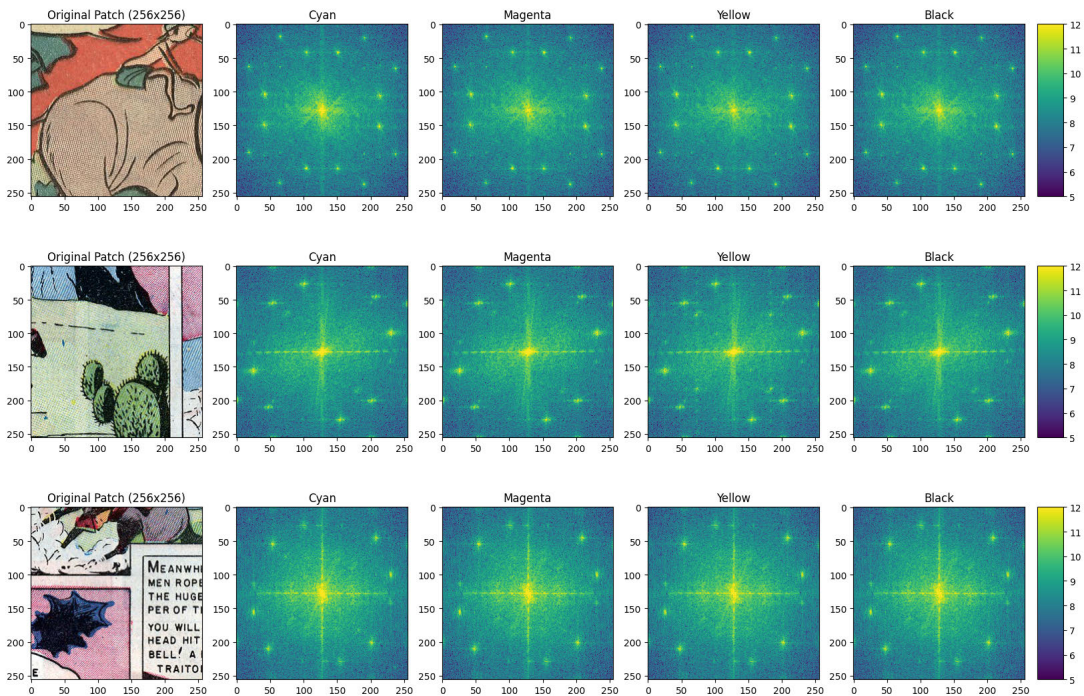

FIGURE 20. Patches with strong Moiré patterns with their FFTs for each channel. FFTs have a grainy appearance and there are several bright spots in the peripheral region.

Running one-tailed paired t-test, we obtained a p-value of 5.56×10^{-6}, showing that Halftone-Net's average MAE is indeed smaller than Halftone-EDSR's. Although the difference in average errors seems small, the p-value indicates that the probability of observing the obtained results if the average errors of the two methods were equal is only 5.56×10^{-6}. This clearly shows the superiority of Halftone-Net over Halftone-EDSR, even using a metric that is not appropriate to measure the quality of the generated image.

C. DETECTING MOIRÉ PATTERNS

As mentioned earlier, mean absolute error (MAE) may not be adequate for measuring upscaling efficiency. If the ideal

TABLE 8. The influence of the number of filters in the quality of the output images.

Model	No. filters	MAE ± SD
(A) Halftone-EDSR	**256**	**0.023 ± 0.01**
(B) Halftone-EDSR	512	0.031 ± 0.01
(C) Halftone-Net	256	0.029 ± 0.01
(D) Halftone-Net	**512**	**0.021 ± 0.009**

image has a strong Moiré pattern, a low MAE may indicate that the resulting image also incorporates a similar Moiré pattern, while a high MAE may indicate that the model has managed to reduce the Moiré pattern. Therefore, it is necessary to look for some other metric to measure upscaling
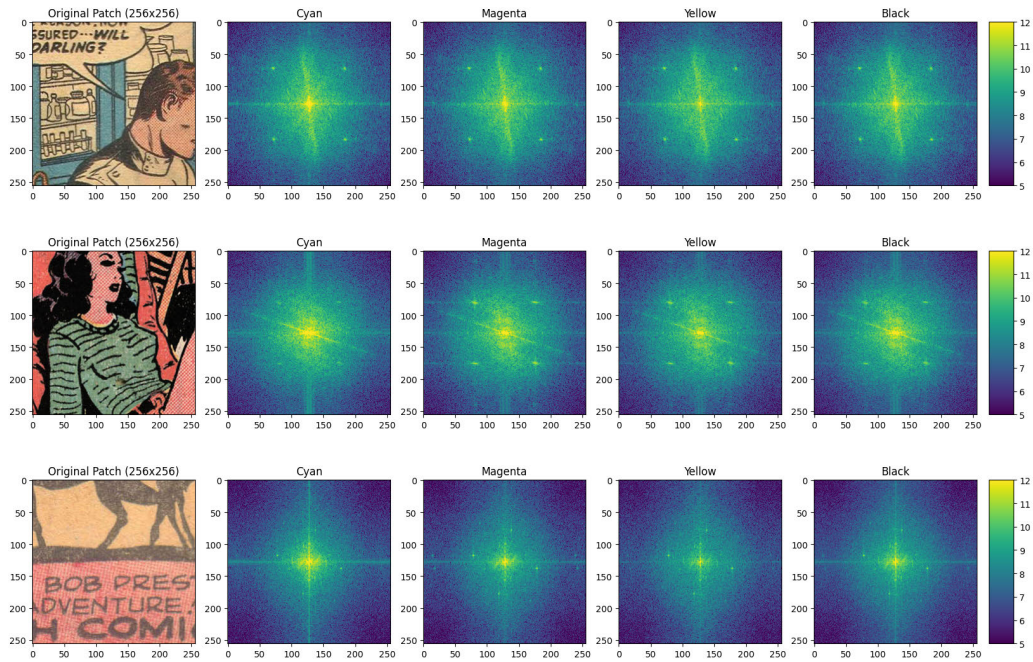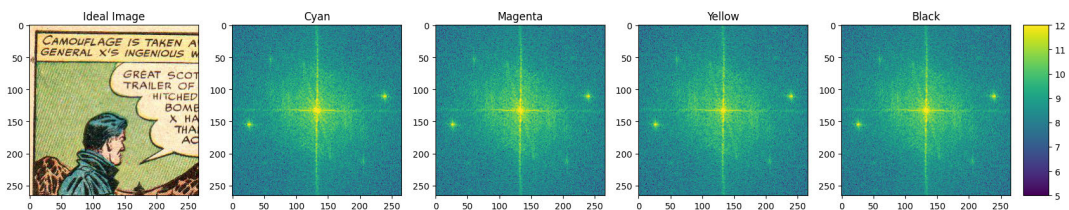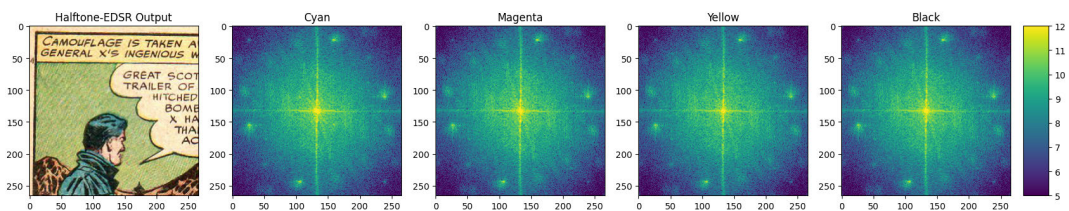
FIGURE 21. Patches with weak Moiré and their FFTs for each channel. Compared with fig. 20, there are fewer bright spots in the peripheral region and the textures are smoother. The central point is more concentrated and brighter.

(a) Ideal output image with strong Moiré patterns. The FFTs have many bright spots in the peripheral regions.

(b) Image processed by Halftone-EDSR. The FFTs show many bright spots in the peripheral regions, despite having a low MAE of 0.0347.
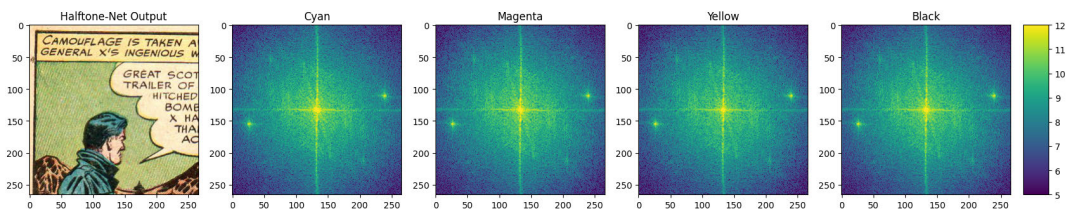
(c) Image processed by Halftone-Net. The FFTs show less bright spots, despite having a higher MAE of 0.0380.

FIGURE 22. Visual comparison of (a) ideal output image, (b) image processed by Halftone-EDSR and (c) image processed by Halftone-Net. The image processed by Halftone-Net has a higher MAE (0.0380) than that processed by Halftone-EDSR (0.0347), but has weaker Moiré patterns and higher quality halftone textures (note the green region).

success. We train a machine learning model that classifies images as containing strong or weak Moiré patterns. Our goal was to show that Halftone-Net generates fewer images with strong Moiré (low-quality) than Halftone-EDSR.

TABLE 9. Number of output images generated by Halftone-EDSR and Halftone-Net categorized as "weak Moiré" and "strong Moiré".

Model	No. filters	Weak Moiré	Strong Moiré
(A) Halftone-EDSR	**256**	**468 (52%)**	**432 (48%)**
(B) Halftone-EDSR	512	387 (43%)	513 (57%)
(C) Halftone-Net	256	477 (53%)	423 (47%)
(D) Halftone-Net	**512**	**621 (69%)**	**279 (31%)**

In the first attempt, we handpicked some images with strong Moiré and others with weak Moiré. Then, we trained a CNN to classify images into strong or weak Moiré. The classification accuracy was 50%, indicating that the network was unable to detect Moiré.

In the second attempt, we decided to pre-process images using Fast Fourier Transform (FFT). The FFT is widely used for Moiré pattern detection [32], [33]. We computed the absolute FFT values, followed by log_{10} transformation, and mapped them to a color scale within the range [5.0, 12.0]. Figs. 20 and 21 illustrates FFTs, highlighting differences between patches with strong and weak Moiré patterns.

We initially curated 400 256×256-sized patches with strong Moiré and 400 patches with weak Moiré. Then, we calculated the FFTs for each color channel of each patch. We trained a model inspired by VGG16 [34], with MSE loss, Adam optimizer with a learning rate of 10^{-6}, a batch size of 20, and for 50 epochs. We conducted 5-fold cross-validation, in each fold using 80% of images for the training and 20% for the testing. This approach achieved an average accuracy of 87.5±3.5%.

Subsequently, we applied this model to all 4×900 images generated by the Halftone-EDSR and Halftone-Net, with both 256 and 512 upsampling filters. Table 9 shows the classification results. As before (Table 8), the best Halftone-EDSR model is (A) with 256 filters and the best Halftone-Net model is (D) with 512 filters. Comparing the best Halftone-EDSR model (A) with the best Halftone-Net model (D), model (D) generated more images with weak Moiré patterns (good visual quality) than model (A).

Fig. 22 visually compares the ideal image, the image upscaled by Halftone-EDSR and by Halftone-Net. The MAE of the image upscaled by Halftone-EDSR (0.0347) is lower than that of the image upscaled by Halftone-Net (0.0380). However, the image upscaled by Halftone-EDSR shows stronger Moiré patterns than the one upscaled by Halftone-Net (visible, for example, in green background).

VII. CONCLUSION

This work has proposed an effective method for upscaling resolution of color halftone images. We took the well-known EDSR model and trained it with halftone images, obtaining Halftone-EDSR.

We introduced cropping as a viable data augmentation technique to train halftone upscaling models. This is particularly significant since the peculiar characteristics of halftones restrict the use of conventional data augmentation techniques. By using cropping and image patches as training samples,

we were able to substantially speed up the training and get models with better quality.

Our experiments have shown that joint-channel model (that upscales four-channel CMYK images at once) is superior to the independent-channel model (that increases each of the four channels independently and then merges them). We experimentally demonstrated that the proposed Halftone-EDSR is superior to all previous techniques for increasing the resolution of halftone images, both for pre-print and post-print halftone images.

We have argued that conventional loss functions (such as MAE, MSE, etc.) may not be adequate for measuring the quality of upscaled halftone images. We proposed to use FFT, followed by a CNN, to measure the presence of strong Moiré patterns in halftone images.

To minimize Moiré patterns, we have proposed a new network model called Halftone-Net. We demonstrated that the new model efficiently increases the resolution of halftone images while reducing Moiré patterns in post-print images.

Despite the effective reduction of Moiré patterns in scanned images by Halftone-Net, 31% of tested images continue to exhibit strong Moiré patterns. As a prospect for future work, efforts will be directed towards further minimizing this 31% occurrence.

Our method has the potential to improve the quality of digitized documents.

REFERENCES

[1] H. Y. Kim, "Binary halftone image resolution increasing by decision tree learning," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1136–1146, Aug. 2004, doi: 10.1109/TIP.2004.828424.

[2] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," 2014, *arXiv:1501.00092*.

[3] T. Frank, O. Haik, S. Gat, O. B. Mor, J. P. Allebach, and Y. Yitzhaky, "A study on halftoning improvement for low-resolution digital print engines with machine learning methods," *IEEE Access*, vol. 10, pp. 19780–19795, 2022.

[4] H. Jiang, D. Xiong, X. Jiang, L. Ding, L. Chen, and K. Huang, "Efficient halftoning via deep reinforcement learning," *IEEE Trans. Image Process.*, vol. 32, pp. 5494–5508, 2023, doi: 10.1109/TIP.2023.3318937.

[5] J.-M. Guo and S. Seshathiri, "Self-supervised learning for scanned halftone classification with novel augmentation techniques," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2023, pp. 2650–2654, doi: 10.1109/ICIP49359.2023.10222434.

[6] L. Shao, E. Zhang, and M. Li, "An efficient convolutional neural network model combined with attention mechanism for inverse halftoning," *Electronics*, vol. 10, no. 13, p. 1574, Jun. 2021, doi: 10.3390/electronics10131574.

[7] Z. Huajian, M. Dazhong, and C. Peng, "An inverse halftoning method using invertible neural network," in *Proc. 8th Int. Conf. Intell. Comput. Signal Process. (ICSP)*, Apr. 2023, pp. 1987–1990, doi: 10.1109/ICSP58490.2023.10248941.

[8] M. Li, E. Zhang, Y. Wang, J. Duan, and C. Jing, "Inverse halftoning methods based on deep learning and their evaluation metrics: A review," *Appl. Sci.*, vol. 10, no. 4, p. 1521, Feb. 2020, doi: 10.3390/app10041521.

[9] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," 2017, *arXiv:1707.02921*.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[11] Y. V. Jawahar, M. Varalakshmi, and P. Mohideen, "Performance analysis of a micromodel-based multinomial classifier," in *Proc. Int. Conf. Comput. Sci. Comput. Intell. (CSCI)*, Dec. 2022, pp. 275–278, doi: 10.1109/CSCI58124.2022.00052.

[12] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[13] S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, and F. Shen, "Image data augmentation for deep learning: A survey," 2022, *arXiv:2204.08610*.

[14] D. L. Lau and G. R. Arce, *Modern Digital Halftoning*. Boca Raton, FL, USA: CRC Press, 2018, doi: 10.1201/9781315219790.

[15] J. Waite, C. Willis, and G. Oliver, *Setting Halftone LPI: Taming the Beasts of Resolution*, The Technology Teacher, 2006, pp. 1–10.

[16] T. N. Pappas, "Digital halftoning techniques for printing," in *Proc. ICPS*, vol. 94, 1994, p. 47.

[17] F. A. Baqai, J.-H. Lee, A. U. Agar, and J. P. Allebach, "Digital color halftoning," *IEEE Signal Process. Mag.*, vol. 22, no. 1, pp. 87–96, Jan. 2005.

[18] P. Goyal and J. P. Allebach, "Print quality assessment for stochastic clustered-dot halftones using compactness measures," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3792–3796, doi: 10.1109/ICIP.2016.7533069.

[19] J. C.-Y. Yang and W.-H. Tsai, "Suppression of Moiré patterns in scanned halftone images by double scans with grid movements," *Pattern Recognit. Lett.*, vol. 18, no. 3, pp. 213–227, 1997, doi: 10.1016/S0167-8655(97)00011-1.

[20] *Moire & False Color*. Accessed: Nov. 11, 2023. [Online]. Available: https://www.nikonusa.com/en/learn-and-explore/a/products-and-innovation/moir%C3%A9-false-color.html

[21] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1122–1131.

[22] B. Zámečník. (Nov. 2020). *Bzamecnik/Halftone: Halftoning in Python*. [Online]. Available: https://github.com/bzamecnik/halftone

[23] D. C. Museum. (2022). *Digital Comic Museum*. [Online]. Available: https://cdn.digitalcomicmuseum.com

[24] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep CNNs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 9, pp. 2917–2931, Sep. 2020, doi: 10.1109/TCSVT.2019.2935128.

[25] X. Chen, X. Wang, J. Zhou, Y. Qiao, and C. Dong, "Activating more pixels in image super-resolution transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 22367–22377.

[26] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "SwinIR: Image restoration using Swin transformer," 2021, *arXiv:2108.10257*.

[27] K. Bajaj, D. K. Singh, and M. A. Ansari, "Autoencoders based deep learner for image denoising," *Proc. Comput. Sci.*, vol. 171, pp. 1535–1541, Jan. 2020, doi: 10.1016/j.procs.2020.04.164.

[28] XPixelGroup. *XPixelGroup/Hat: CVPR2023—Activating More Pixels in Image Super-Resolution Transformer*. Accessed: Oct. 3, 2023. [Online]. Available: https://github.com/XPixelGroup/HAT

[29] J. Wu, J. Wang, J. Zhao, X. Luo, and B. Ma, "ESGAN for generating high quality enhanced samples," *Multimedia Syst.*, vol. 28, no. 5, pp. 1809–1822, Oct. 2022, doi: 10.1007/s00530-022-00953-3.

[30] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8107–8116.

[31] Y. Sun, Y. Yu, and W. Wang, "Moiré photo restoration using multiresolution convolutional neural networks," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 4160–4172, Aug. 2018, doi: 10.1109/TIP.2018.2834737.

[32] F. Joucken, F. Frising, and R. Sporken, "Fourier transform analysis of STM images of multilayer graphene Moiré patterns," *Carbon*, vol. 83, pp. 48–52, Mar. 2015, doi: 10.1016/j.carbon.2014.11.030.

[33] B. P. Ölveczky and A. Antal, "Spectral analysis of Moiré images," *Periodica Polytechnica Mech. Eng.*, vol. 41, pp. 85–93, Jan. 1997. [Online]. Available: https://api.semanticscholar.org/CorpusID:56091558

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

GUILHERME APOLINARIO SILVA NOVAES (Graduate Student Member, IEEE) was born in Santos, São Paulo, Brazil. He received the B.Sc. degree in computer science from Universidade Católica de Santos (UniSantos), São Paulo, in 2015, and the M.S. degree in microelectronics from the University of São Paulo (USP), São Paulo, in 2019, where he is currently pursuing the Ph.D. degree in electronics systems. Since 2013, he has been actively involved in research in the field of computer vision for electronic systems. He holds a patent in the field of microelectronics and combinatorial optimization. In 2021, he assumed the role of an Assistant Professor with UniSantos. His research interests include combinatorial optimization, image processing, computational creativity, and deep learning.

HAE YONG KIM was born in South Korea. He received the B.S. and M.S. degrees (Hons.) in computer science and the Ph.D. degree in electrical engineering from Universidade de São Paulo (USP), Brazil, in 1988, 1992, and 1997, respectively.

He is currently an Associate Professor with the Department of Electronic Systems Engineering, USP. He is the author of more than 100 articles and holds three patents. His research interests include image processing, machine learning, medical image processing, and computer security.

Dr. Kim and colleagues received the sixth edition of the Petrobras Technology Award in the "Refining and Petrochemical Technology" category, in 2013; the "Best Paper in Image Analysis" Award at the Pacific-Rim Symposium on Image and Video Technology, in 2007; and the Thomson ISI Essential Science Indicators "Hot Paper" Award, for writing one of the top 0.1% of the most cited computer science papers, in 2005.

• • •