

Deep residual neural networks for inverse halftoning

Heri Prasetyo¹, Muhamad Aditya Putra Anugrah¹, Alim Wicaksono Hari Prayuda², Chih-Hsien Hsia³

¹Department of Informatics, Faculty of Mathematics and Natural Sciences, Universitas Sebelas Maret (UNS), Surakarta, Indonesia

²Department of Electrical Engineering, College of Electrical Engineering and Computer Science, National Taiwan University of Science and Technology, Taipei City, Taiwan

³Department of Computer Science and Information Engineering, College of Electrical Engineering and Computer Science, National Ilan University, Yilan County, Taiwan

Article Info

Article history:

Received Apr 09, 2021

Revised Jul 30, 2022

Accepted Aug 15, 2022

Keywords:

Deep learning
Error diffusion
Floyd-steinberg
Inverse halftoning
Residual networks

ABSTRACT

This paper presents a simple technique to perform inverse halftoning using the deep learning framework. The proposed method inherits the usability and superiority of deep residual learning to reconstruct the halftone image into the continuous-tone representation. It involves a series of convolution operations and activation function in forms of residual block elements. We investigate the usage of pre-activation function and standard activation function in each residual block. The experimental section validates the proposed method ability to effectively reconstruct the halftone image. This section also exhibits the proposed method superiority in the inverse halftoning task compared to that of the handcrafted feature schemes and former deep learning approaches. The proposed method achieves 30.37 dB and 0.9481 on the average peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) scores, respectively. It gives the improvements around 1.67 dB and 0.0481 for those values compared to the most competing scheme.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Heri Prasetyo

Department of Informatics, Faculty of Mathematics and Natural Sciences

Universitas Sebelas Maret (UNS), 36 Ir. Sutami Road, Kentingan, Jebres, Surakarta City 57126, Indonesia

Email: heri.prasetyo@staff.uns.ac.id

1. INTRODUCTION

The digital halftoning technique is powerful tools in the rendering devices such as the digital printing [1]. It converts an image with continuous-tone representations into only two-tone values. This conversion always considers the quality of rendered image in two-tone representations. Each pixel in the rendered image, or called as halftone image, only has two values, i.e. black or white pixels. The combinations of all pixels in black and white can generate visual illusion such that one may recognize the halftone image as the original image under human perception over a specific distance view. A good digital halftoning is able to produce the halftone image with similar visual illusion compared to the original input image.

Some applications require the conversion of the halftone image consisting of two-tone values back into its continuous-tone image. This conversion process is referred as inverse halftoning. Numerous works have been presented in order to develop a new method or to gain an increased performance for effective inverse halftoning technique. Commonly, the former inverse halftoning methods are based on the handcrafted features [2]–[6]. However, some advanced progresses have been made in the inverse halftoning task under the deep learning approaches [7]–[9]. As reported in literature, the deep learning-based methods deliver better results compared to that of the handcrafted feature. The deep learning frameworks have also been successfully reported to produce sophisticated results in some applications, such as handwriting recognition [10], road recognition [11], image reconstruction [12], impulsive noise suppression [13], sound event detection [14],

and handwriting identification [15]. The presented method in this paper performs inverse halftoning task under the residual learning framework. This residual learning method can be further extended to several applications such as image forgery detection [16], image enhancement [17], automatic image retrieval [18], and other computer vision tasks. This paper is composed with the following organizations: section 2 briefly discusses the digital halftoning technique using the error diffusion approach. This section also presents several techniques for inverse halftoning. The proposed method for inverse halftoning using deep residual learning is delivered in section 3. Section 4 discusses and summarizes some inverse halftoning experiments. The conclusion remarks are then given at the of this paper.

2. IMAGE HALFTONING AND ITS INVERSE PROBLEM

The section briefly reviews the digital halftoning technique using the error diffusion approach [1]. This halftoning method utilizes a specific error kernel, namely floyd-steinberg kernel, to perform the image thresholding approach. Subsequently, several methods for performing inverse halftoning are also presented in this section. We firstly begin our discussion with the error diffusion using the floyd-steinberg kernel for two dimensional grayscale image. This method can be easily extended for color image by treating each color channel as an individual two dimensional grayscale image. Let $F(x, y)$ be an image pixel at position (x, y) for $x = 1, 2, \dots, W$ and $y = 1, 2, \dots, H$, where W and H denote the weight and height of an image, respectively. This pixel is regarded as continues-tone since it lies in an interval $[0, 255]$. The main goal of digital halftoning is to convert this continues-tone pixel into two-tone presentations under the constraint that the halftoned image gives almost similar visual appearance compared to the original continues-tone. For performing the digital halftoning, one needs to compute the mean value over all image pixels using the (1).

$$\mu = \sum_{x=1}^W \sum_{y=1}^H F(x, y) \quad (1)$$

Where μ is the average value of all pixels. Afterwards, the hard thresholding applies for each continues tone pixel with:

$$I(x, y) = \begin{cases} 0, & F(x, y) < \mu \\ 255, & F(x, y) \geq \mu \end{cases} \quad (2)$$

Where $I(x, y)$ denotes the thresholded pixel at position (x, y) . Each thresholded pixel constitutes into a single image, namely halftone image. The thresholding technique in (2) simply classifies a pixel which is higher than mean value into the bright tone (white pixel), and vice versa.

In fact, the hard thresholding as defined in (2) produces an error, i.e. the difference between the pixel $F(x, y)$ and $I(x, y)$. The error caused by the thresholding procedure can be compute as:

$$e(x, y) = F(x, y) - I(x, y) \quad (3)$$

Where $e(x, y)$ represents the error value at pixel position (x, y) . The quality of halftone image $I(x, y)$ will be improved and more acceptable if we are able to diffuse this error into its neighboring pixel. To conduct this diffusion process, we need an auxiliary kernel, namely floy steinberg error kernel as shown in Figure 1. The error diffusion process can be performed as:

$$F(x, y) \leftarrow F(x, y) + e(x, y) \times \varepsilon \quad (4)$$

Where ε and \times are the value of error kernel and convolution operator, respectively. All thresholding and error diffusion processes are repeated over all pixels to obtain the halftone image I . Figure 2 illustrates the digital halftoning for color image. Herein, each color channel is independently processed with error diffusion halftoning using floyd-steinberg kernel.

In most common situations, the human vision cannot fully accept the quality of halftone image in dot-disperse and noisy-like patterns. It is caused by the truncated continuous-tone into only two tone representations. Several efforts have been undertaken for improving the quality of halftone image or reconstructing the halftone image back into its original continuous tone. Typically, a technique aiming to convert the halftone image into continuous-tone is referred as inverse halftoning method. Figure 3 shows the process of inverse halftoning. The methods [2]–[8] perform the inverse halftoning effectively by producing high quality of reconstructed continuous tone. A simple approach for inverse halftoning is presented in [2] which employs the maximum a posteriori. Whereas, the method in [4] executes the anisotropic diffusion to reconstruct the halftone image. The technique in [3] improves the quality of halftone image using the deconvolution operation and regularized wiener inverse, while the methods in [5], [6] exploit the superiority of sparse representation technique to replace the halftone image patch with the continue-tone patch. All these

techniques can be regarded as handcrafted inverse halftoning method. In recent years, some progresses and developments have been achieved in the inverse halftoning task incorporating the recent advances of deep learning approach. The method in [7] performs the inverse halftoning using deep learning approach under the encoder-decoder computation, whereas former scheme in [8] takes the structure-aware deep learning for inverse halftoning. All these mentioned methods yield promising results in the reconstructed halftoning image. The halftoning method has been proven to yield an effective result on image retrieval [19] and image compression [20].

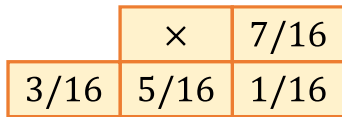


Figure 1. Floyd-steinberg error kernel

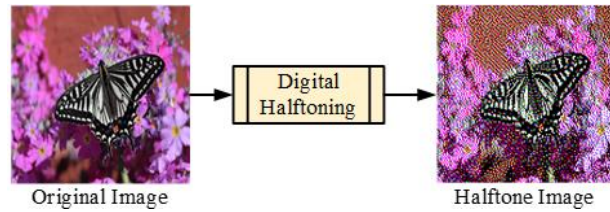


Figure 2. Illustration of digital halftoning

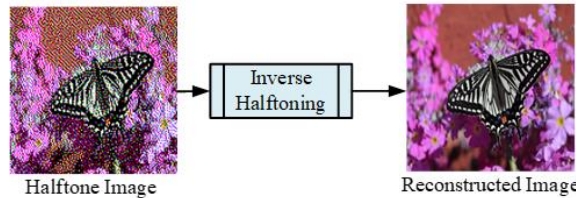


Figure 3. Illustration of inverse halftoning

3. PROPOSED INVERSE HALFTONING

3.1. Proposed architecture

The proposed method exploits the residual learning to perform the inverse halftoning task. Let I be the halftone image. The proposed method aims to learn an efficient mapping that transforms a halftone image I into its continuous approximation, i.e. the reconstructed halftone image \hat{I} . This process is denoted as $\hat{I} = \mathfrak{F}\{I\}$, while $\mathfrak{F}\{\cdot\}$ is the proposed end-to-end mapping. The proposed method should consider the requirement that the quality of \hat{I} should be as similar as possible to I , i.e. $\hat{I} = I$ or $\hat{I} \approx I$.

Suppose that I be the halftone image in color space, while its size is denoted as $H \times W \times C$. Herein, the image height, width, and the number of color channels are represented as H , W , and C , respectively. The value of $C = 3$ indicates the color image, whereas $C = 1$ is grayscale image. Figure 4 depicts the proposed method for inverse halftoning task. It receives an input image of size $H \times W \times C$, afterwards, it produces the output image over identical size with the original image. Firstly, the proposed method performs the convolution operation on I using the (5).

$$I_a = \eta(I \times W_a + b_a) \tag{5}$$

Where W_a and b_a are the weights and biases, respectively, in this convolution process. This convolution process consists of n_a filters, where each filter is of size $f_a \times f_a \times C$. The symbol $\eta(\cdot)$ is the activation function. This convolution process produces n_a feature maps denoted as I_a .

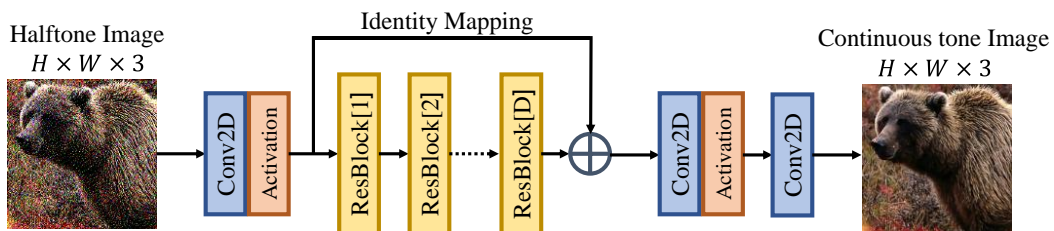


Figure 4. Schematic diagram of the proposed resnet architecture for inverse halftoning

The feature maps produced at the first convolution layer are subsequently fed into a series of residual blocks. The process of each residual block is formally defined:

$$I_d = R_d\{I_{d-1}\} \quad (6)$$

Where $R_d\{\cdot\}$ is the operator in the d -th residual block, for $d = 1, 2, \dots, D$. The symbol D indicates the number of residual blocks. While I_d denotes the feature maps produced at the d -th residual block. For $d = 1$, we simply set $I_0 = I_a$. The proposed method produces the feature maps I_D at the end of residual blocks, i.e. $d = D$. The dimensionality of feature maps I_D should be identically maintained as in the feature maps I_a , i.e. n_a . Since, we perform identity mapping I_a and element-wise addition at the end of residual blocks [21]. The process of element-wise addition is denoted as:

$$I_b = I_a \oplus I_D \quad (7)$$

Where I_b and \oplus denote the feature maps after element-wise addition, and operator of this addition, respectively. The dimensionality of I_b is also n_a .

After element-wise addition process, the two convolution series are applied to the feature maps I_b . The first convolution process of I_b is indicated with the:

$$I_c = \eta(I_b \times W_c + b_c) \quad (8)$$

Where W_c and b_c are the weights and biases, respectively, for this convolution. This process involves n_c filters, each of size $f_c \times f_c \times n_a$. This process yields the feature maps I_c of size n_c . This process utilizes the activation function. Subsequently, the second process performs the convolution of I_c as:

$$I_e = I_c \times W_e + b_e \quad (9)$$

Where W_e and b_e are the weights and biases, respectively, involved in the convolution operation. This stage does not involve the activation function. In addition, this process needs n_e filters, each of size $f_e \times f_e \times n_c$ to yield feature maps I_d of size n_e . By setting $n_e = C$, we obtain the final feature maps I_e with an identical size to that of the original input image. We denote I_e as the reconstructed halftone image, i.e. $\hat{I} = I_e$. We set $n_e = C = 3$ for color image, while $n_e = C = 1$ for grayscale image.

3.2. Elements of normal residual block

This subsection explains the elements of each residual block used in the proposed method. As discussed before, each residual block performs feature mapping as $I_d = R(I_{d-1})$ for $d = 1, 2, \dots, D$. In the normal residual block setting, we simply employ two convolution series. The illustration of this residual block is given in Figure 5. The first convolution is denoted as:

$$M_a = \eta\{Norm\{I_{d-1} \times W_1 + b_1\}\} \quad (10)$$

Where W_1 and b_1 are the weights and biases of this convolution process, respectively. The symbol $Norm\{\cdot\}$ denotes the operator of image batch normalization [22]. This stage produces the feature maps M_a . The second convolution processes the feature maps M_a using the:

$$M_b = \eta\{Norm\{M_a \times W_2 + b_2\}\} \quad (11)$$

Where W_2 and b_2 are the weights and biases in this convolution process. This process produces the feature maps M_b . The element-wise addition is performed at the end of each residual block as:

$$M_c = M_a \oplus M_b \quad (12)$$

Where M_c is the produced feature maps implying $I_d = M_c$. These two convolution layers are applied to all residual blocks for $d = 1, 2, \dots, D$.

3.3. Elements of residual block with pre-activation function

The residual block can be alternatively designed with the pre-activation function. In this scenario, the activation function is firstly executed before the convolution process. The process of each residual block is denoted as $I_d = R(I_{d-1})$ for $d = 1, 2, \dots, D$. The residual block with pre-activation function also involves two convolution series. Figure 6 shows the element of residual block with pre-activation function. The first convolution process is performed as:

$$M_a = \eta\{Norm\{I_{d-1}\}\} \times W_1 + b_1 \quad (13)$$

Whereas, the second convolution process on this residual block is conducted as:

$$M_b = \eta\{Norm\{M_a\}\} \times W_2 + b_2 \quad (14)$$

It can be seen from these two formulations, the activation function is applied before convolution operation. The element-wise operation is further computed as:

$$M_c = M_a \oplus M_b \quad (15)$$

The feature maps produced at this residual block are denoted as M_c or $I_d = M_c$. This process is performed for all residual blocks $d = 1, 2, \dots, D$.

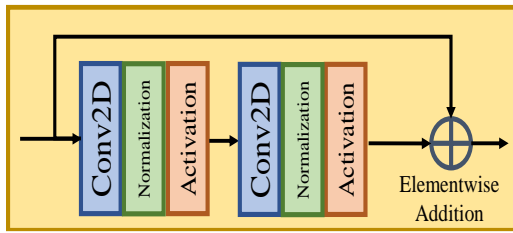


Figure 5. Elements of normal residual block

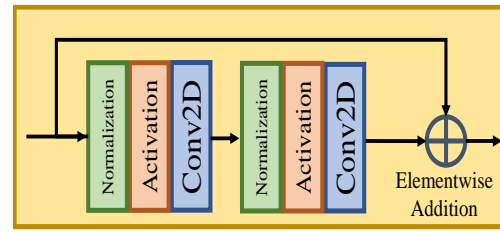


Figure 6. Elements of residual block with pre-activation function

3.4. Loss function in learning process

The proposed method performs end-to-end learning to reconstruct the halftone image. This process involves the training activity under a specific loss function. The mean-squared error (MSE) is chosen as our loss function since of its simplicity. The MSE definition is given as:

$$l(\hat{I}, F) = \|\hat{I} - F\|_F^2 \quad (16)$$

Where $l(\hat{I}, F)$ denotes the loss function between the reconstructed image \hat{I} and original image F . The symbol $\|\cdot\|_F$ is the operator of frobenius norm. To reduce the computational time, we conduct the training process on the image patch basis rather than the full size image. Let $\{I_t, F_t\}_{t=1}^N$ be a paired halftone image patch and its original version, while N denotes the number of image patches involved in the training process. Let $\Theta = \{W_a, b_a, W_b, b_b, \dots, W_e, b_e\}$ is the trainable parameters for the proposed method. Subsequently, the optimization process can be easily performed under the image patch basis as:

$$arg \min_{\Theta} \frac{1}{N} \sum_{t=1}^N l(I_t, \Theta, F_t) = arg \min_{\Theta} \frac{1}{N} \sum_{t=1}^N \|I_t, \Theta - F_t\|_F^2 \quad (17)$$

Where symbol (I_t, Θ) denotes the inverse halftoning on image patch I_t with the network parameters Θ . This optimization requires a number of iterations to yield optimum results.

4. EXPERIMENTAL RESULTS

4.1. Parameters tuning

In this subsection, we perform parameters tuning for our proposed resnet method. The parameters tuning is performed in the training process. Herein, we need two image sets in color version. The first image set is referred as training set obtained from the DIV2K image dataset [23]. Each image is divided into non-overlapping image patches, each of size 128×128 . Figure 7 displays some training image samples. Whereas, the second image set is the validation set composed from DIV2K image dataset [23] in the downsampled version with bicubic interpolation by scale $\times 4$. Figure 8 exhibits some image samples for the validation set. We need a paired images to train the proposed network, i.e. clean (original) image and halftone image version. In this stage, each halftone image is fed into the proposed network, then the error between the reconstructed image produced by the proposed network is computed against the clean image. All experiments are conducted under the computational environments: AMD Ryzen threadripper 1950X CPU and Nvidia GTX 1080 Ti GPU.



Figure 7. Some image samples as training set

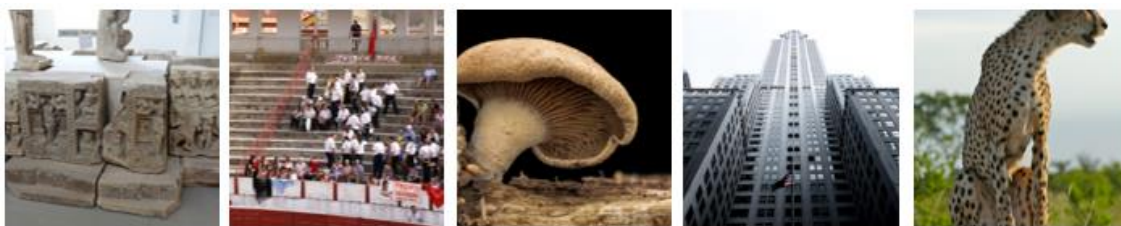


Figure 8. A set of validation images

We firstly overlook the effect of different numbers of features for the proposed network. Herein, we simply use the MSE metric as a loss function. We employ an image batch with size 32. The Adam optimizer [24] performs optimization of network parameters with initial learning rate 0.001. This learning value is further divided with 2 on every 5 epochs. The number of epoches is set as 30, while each epoch consists of 1261 iteration. All weights and biases are initialized using the similar strategy as used in [25]. In this experiment, we simply use four residual blocks without batch normalization. The parametric rectified linear unit (PReLU) is employed as an activation function.

Figure 9(a) shows the average loss function in terms of peak signal-to-noise ratio (PSNR) score over validation image set under various number of feature maps, i.e. $N = 16$, $N = 32$, and $N = 48$. As depicted from this figure, the 48 feature maps yield the best performance in the training process. Thus, we simply use the feature maps as $N = 48$ for the subsequent experiment.

Subsequently, we investigate the effect of various residual blocks for the proposed network. Herein, we observe different number of residual blocks $D = \{2, 4, \dots, 10\}$ under the number of feature maps $N = 48$. Whereas, the other experimental settings remain unchanged. Figure 9(b) exhibits the performance of the proposed network during the training stage over various number of residual blocks. This figure reveals that the number of residual blocks $D = 10$ gives the best performance for the proposed method indicating with the highest average PSNR score in the validation set. Thereafter, we utilize $N = 48$ and $D = 10$ for the subsequent training process.

The effects of activation function and image batch normalization method are examined for the proposed network. In this experiment, we employ $N = 48$ and $D = 10$, while the other parameters setting is maintained unchanged. Several activation functions such as PReLU, rectified linear unit (RELU), and others are investigated. Figure 9(c) reports the average loss of validation set over various activation functions. This figure indicates that the PReLU activation function offers the best performance compared to the other functions. As a result, the PReLU activation function is more preferable compared to the others.

The effect of image batch normalization is further observed under $N = 48$, $D = 10$, and PReLU activation function. Herein, we investigate the effect of batch normalization (BN) and instance normalization (IN) with per-activation and normal activation setting. Figure 9(d) displays the effect of various batch normalization for the proposed method. Based on this figure, we can conclude that the proposed method without image batch normalization gives the best performance for validation set during the training process. Thus, we apply the proposed method with experimental settings $N = 48$, $D = 10$, PReLU activation function and without image batch normalization for the inverse halftoning task.

4.2. Visual investigation

This subsection inspects the performance under visual investigation. The quality of reconstructed image is visually noticed and compared with the original image. Herein, we examine the performance under several images as previously used in [6]. Figure 10 displays some color images for the testing set. These testing images contain rich image detail and texture making it challenging in the inverse halftoning task. For performing the inverse halftoning, we utilize the proposed method with the best parameters setting as

described in section 4.1. Figure 11 exhibits the visual comparisons on the reconstructed halftone images. As shown in these figures, the proposed method produces the best visual quality of reconstructed image in comparison to other scheme [6]. Since of the paper length limitation, we only give visual comparison against [6]. Therefore, it offers competitive advantage in the inverse halftoning task.

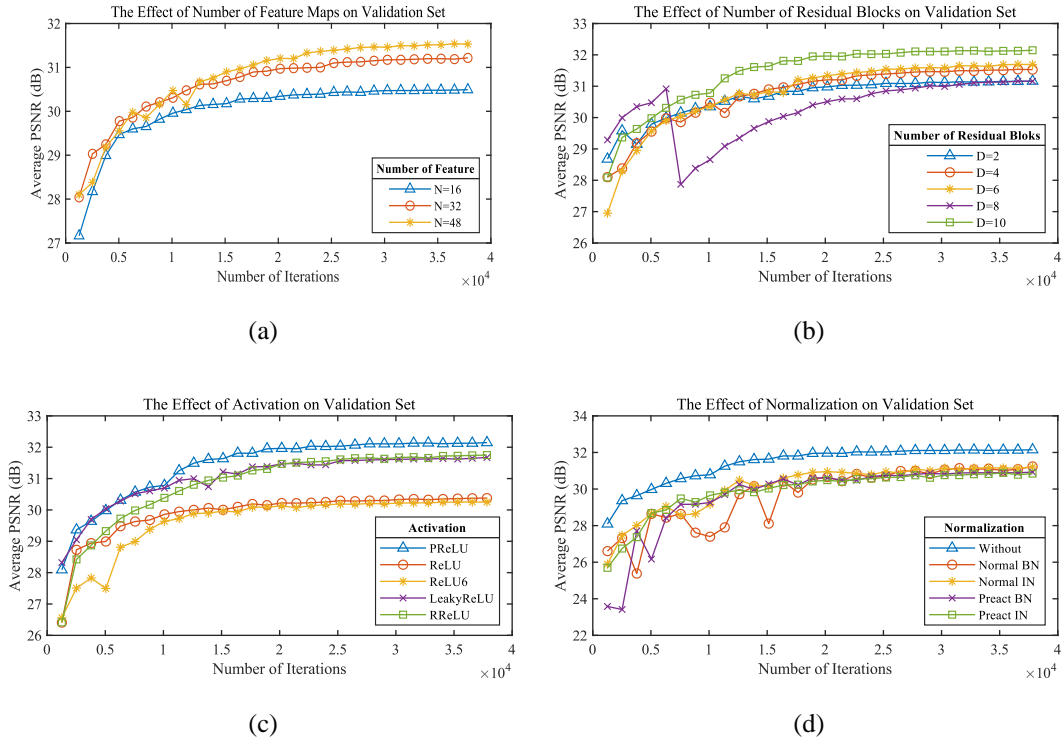


Figure 9. The average PSNR during the training process over various conditions: (a) feature maps, (b) the number of residual blocks, (c) activation function, and (d) batch normalization



Figure 10. A set of testing images used for experiments

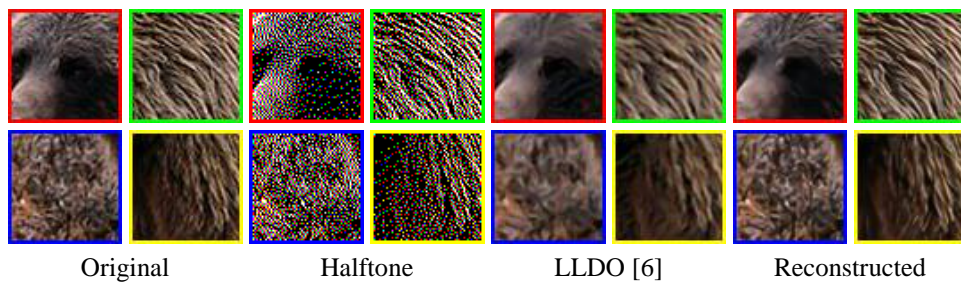


Figure 11. Visual investigation on bear image

4.3. Performance comparisons

Additional comparisons are summarized and discussed in this subsection. We compare the performances in terms of objective image quality assessments, i.e. the average PSNR and structural similarity index (SSIM) scores over testing set as given in Figure 10. The inverse halftoning technique is applied for each image in the testing set. Subsequently, the average PSNR and SSIM values are computed over all testing images. We employ the proposed method with the best parameters tuning as discussed in section 4.1. Table 1 tabulates the performance comparisons in terms of average PSNR, while Table 2 recapitulates these comparisons in terms of average SSIM score. As tabulated in these two tables, the proposed method outperforms the former existing schemes in the inverse halftoning task indicated with the highest values of PSNR and SSIM. Herein, the proposed method and former schemes employ an identical experimental environments, i.e. use identical image dataset, utilize the same floyd-steinberg halftoning technique, and measure the performance under the similar image assessment metrics. The proposed method yields a good performance in the inverse halftoning task, not only under visual investigation, but it also proves its outstanding performance in the objective measurement metrics. It is worth noted that the proposed method yields better performance compared to the handcrafted feature methods [2]–[5] and deep learning-based approaches [7], [8]. The proposed method requires lower network parameters compared to that of [7] and [8]. In addition, the proposed method should be firstly considered while one conducts and implements the inverse halftoning task.

Table 1. Performance comparisons in terms of PSNR score

Testing images	ALF [3]	MAP [2]	LPA-ICI [4]	GLDP [5]	LLDO [6]	SADCNN [8]	CNN Inv [7]	Proposed method
Koala	22.36	23.33	24.17	24.58	25.01	25.66	27.63	29.05
Cactus	22.99	23.95	25.04	25.4	25.55	25.63	27.69	29.07
Bear	21.82	22.63	23.14	23.66	24.17	-	26.35	27.98
Barbara	25.41	26.24	27.88	27.12	28.48	29.41	31.79	33.36
Shop	22.14	22.46	24.12	23.86	24.61	25.47	27.27	29.25
Peppers	30.92	28.25	30.7	30.92	31.07	32.29	31.44	33.52
Average	24.27	24.48	25.84	25.92	26.48	-	28.7	30.37

Table 2. Performance comparison in terms of SSIM value

Testing Images	ALF [3]	MAP [2]	LPA-ICI [4]	GLDP [5]	LLDO [6]	SADCNN [8]	CNN Inv [7]	Proposed method
Koala	0.6592	0.7412	0.7557	0.7831	0.7987	0.824	0.89	0.917
Cactus	0.6368	0.7746	0.7871	0.8083	0.8175	0.907	0.92	0.9589
Bear	0.6198	0.7746	0.724	0.766	0.7815	-	0.89	0.9322
Barbara	0.7138	0.7804	0.8293	0.7993	0.8463	0.882	0.92	0.952
Shop	0.64	0.6919	0.7719	0.7535	0.7976	0.866	0.89	0.9419
Peppers	0.8674	0.7681	0.8735	0.8695	0.8698	0.982	0.89	0.9865
Average	0.6895	0.7461	0.7903	0.7966	0.8185	-	0.9	0.9481

5. CONCLUSION

A simple technique for performing the inverse halftoning task has been proposed. The presented technique exploits the usability of residual network framework in order to recover the continuous-tone image from its halftone version. The presented method consists of a series of convolutional operators formed in the residual block to iteratively adjust the quality of reconstructed image. The experimental section shows that the proposed method gives the best performances compared to that of the former schemes under subjective and objective image quality assessments. Furthermore, the deep learning approach with residual learning can be a good candidate for inverse halftoning technique. It is simple approach with outstanding performance.

For the future works, the proposed scheme can be extended for the other halftoning techniques, not only for the floyd-steinberd based image halftoning. The residual framework can be simply replaced with more sophisticated learnings such as shared source residual learning, xception module, and others. In addition, the generative adversarial networks (GAN), transformer, linear time transformer, and recent deep learning networks can be investigated to replace the convolutional neural networks (CNN)-based module in the proposed method. The other loss functions such as SSIM or patch-based loss computation can also examined in order to improve the proposed method performance. The combination of several loss functions may improve the proposed method performance accordingly.




REFERENCES

- [1] K. -L. Chung *et al.*, "A gradient-based adaptive error diffusion method with edge enhancement," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1591–1601, 2011, doi: 10.1016/j.eswa.2010.07.079.
- [2] R. L. Stevenson, "Inverse halftoning via MAP estimation," *IEEE Transactions on Image Processing*, vol. 6, no. 4, pp. 574–583, 1997, doi: 10.1109/83.563322.




- [3] T. D. Kite, N. D. Venkata, B. L. Evans, and A. C. Bovik, "A fast, high-quality inverse half-toning algorithm for error diffused half-tones," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1583–1592, 2000, doi: 10.1109/83.862639.
- [4] A. Foi, V. Katkovnik, K. Egiazarian, and J. Astola, "Inverse half-toning based on the anisotropic LPA-ICI deconvolution," *Proceedings of The 2004 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, SMMSP.*, 2004. [Online]. Available: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.80.6964&rep=rep1&type=pdf>
- [5] C. -H. Son, "Inverse half-toning based on sparse representation," *Optics Letters*, vol. 37, no. 12, pp. 2352–2354, 2012, doi: 10.1364/OL.37.002352.
- [6] C. -H. Son and H. Choo, "Local learned dictionaries optimized to edge orientation for inverse half-toning," *IEEE Transactions on Image Processing*, vol. 23, no. 6, pp. 2542–2556, 2014, doi: 10.1109/TIP.2014.2319732.
- [7] X. Hou and G. Qiu, "Image companding and inverse half-toning using deep convolutional neural networks," *arXiv preprint arXiv:1707.00116*, 2017. [Online]. Available: <https://arxiv.org/pdf/1707.00116.pdf>
- [8] C. -H. Son, "Inverse half-toning through structure-aware deep convolutional neural networks," *Signal Processing*, vol. 173, 2020, doi: 10.1016/j.sigpro.2020.107591.
- [9] H. P. A. Wicaksono, H. Prasetyo, and J. -M. Guo, "Deep learning based inverse half-toning via stationary wavelet domain," in *2020 27th International Conference on Telecommunications (ICT)*, 2020, pp. 1–5, doi: 10.1109/ICT49546.2020.9239532.
- [10] A. Palvanov and Y. I. Cho, "Comparisons of deep learning algorithms for MNIST in real-time environment," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 18, no. 2, pp. 126–134, 2018, doi: 10.5391/IJFIS.2018.18.2.126.
- [11] S. -M. Cho and B. -J. Choi, "CNN-based recognition algorithm for five classes of roads," *International Journal of Fuzzy Logic and Intelligent Systems* 2020, vol. 20, no. 2, pp. 114–118, 2020, doi: 10.5391/IJFIS.2020.20.2.114.
- [12] H. Prasetyo, A. W. H. Prayuda, C. -H. Hsia, and J. -M. Guo, "Deep concatenated residual networks for improving quality of half-toning-based BTC decoded image," *Journal of Imaging*, vol. 7, no. 2, 2021, doi: 10.3390/jimaging7020013.
- [13] M. F. I. Amal, H. P. A. Wicaksono, and H. Prasetyo, "Deep residual networks for impulsive noise suppression," in *2020 27th International Conference on Telecommunications (ICT)*, 2020, pp. 1–5, doi: 10.1109/ICT49546.2020.9239522.
- [14] S. -H. Jung and Y.-J. Chung, "Sound event detection using deep neural networks," *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, vol. 18, no. 5, pp. 2587–2596, doi: 10.12928/telkomnika.v18i5.14246.
- [15] O. Sudana, I. W. Gunaya, and I. K. G. D. Putra, "Handwriting identification using deep convolutional neural network method," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 4, pp. 1934–1941, 2020, doi: 10.12928/telkomnika.v18i4.14864.
- [16] I. B. K. Sudiatmika, F. Rahman, Trisno, and Suyoto, "Image forgery detection using error level analysis and deep learning," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 2, pp. 653–659, 2019, doi: 10.12928/telkomnika.v17i2.8976.
- [17] J. -M. Guo, H. Prasetyo, and K. Wong, "Half-toning-based block truncation coding image restoration," *Journal of Visual Communication and Image Representation*, vol. 35, pp. 193–197, 2016, doi: 10.1016/j.jvcir.2015.12.016.
- [18] H. A. Al-Jubouri and S. M. Mahmood, "A comparative analysis of automatic deep neural networks for image retrieval," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 19, no. 3, pp. 858–871, 2021, doi: 10.12928/telkomnika.v19i3.18157.
- [19] J. -M. Guo, H. Prasetyo, and J. -H. Chen, "Content-based image retrieval using error diffusion block truncation coding features," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 466–481, 2015, doi: 10.1109/TCSVT.2014.2358011.
- [20] H. Prasetyo, C. -H. Hsia, and B. A. P. Akardihas, "Half-toning-based BTC image reconstruction using patch processing with border constraint," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 1, pp. 394–406, 2020, doi: 10.12928/telkomnika.v18i1.12837.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," *2016 European Conference on Computer Vision*, 2016, vol. 9908, pp. 630–645, doi: 10.1007/978-3-319-46493-0_38.
- [22] A. Zaemzadeh, N. Rahnavard, and M. Shah, "Norm-preservation: why residual networks can become extremely deep?," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 3980–3990, 2021, doi: 10.1109/TPAMI.2020.2990339.
- [23] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: dataset and study," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 1122–1131, doi: 10.1109/CVPRW.2017.150.
- [24] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [Online]. Available: <https://arxiv.org/pdf/1412.6980.pdf>
- [25] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on imagenet classification," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1026–1034. [Online]. Available: https://openaccess.thecvf.com/content_iccv_2015/html/He_Delving_Deep_into_ICCV_2015_paper.html

BIOGRAPHIES OF AUTHORS






Heri Prasetyo    received his Bachelor's Degree from the Department of Informatics Engineering, Institut Teknologi Sepuluh Nopember (ITS), Indonesia in 2006. He received his Master's and Doctoral Degrees from the Department of Computer Science and Information Engineering, and Department of Electrical Engineering, respectively, at the National Taiwan University of Science and Technology (NTUST), Taiwan, in 2009 and 2015, respectively. He received the Best Dissertation Award from the Taiwan Association for Consumer Electronics (TACE) in 2015, Best Paper Awards from the International Symposium on Electronics and Smart Devices 2017 (ISESD 2017), ISESD 2019, and International Conference on Science in Information Technology (ICSITech 2019), and the Outstanding Faculty Award 2019 from his current affiliated university. His research interests include multimedia signal processing, computational intelligence, pattern recognition, and machine learning. He can be contacted at email: heri.prasetyo@staff.uns.ac.id.






Muhamad Aditya Putra Anugrah    received his Bachelor's Degree from the Department of Informatics, Universitas Sebelas Maret (UNS), Indonesia in 2021. His research interests include digital signal processing, image processing, deep learning, and inverse imaging problem. He can be contacted at email: mapaditya@student.uns.ac.id.



Alim Wicaksono Hari Prayuda    received the bachelor degree from the Universitas Sebelas Maret, Surakarta, Indonesia, in 2020. He is currently working toward the master degree in Department of Electrical Engineering at National Taiwan University of Science and Technology, Taipei, Taiwan. His research interests include Deep Learning application for Digital Image Processing and Multimedia Engineering. He can be contacted at email: wicayudha.wy@gmail.com.



Chih-Hsien Hsia    received the Ph.D. degree in Electrical Engineering from Tamkang University, New Taipei, Taiwan, in 2010. He was an Associate Professor with the Chinese Culture University and National Ilan University from 2015 to 2017. He is currently a Professor with the Department of Computer Science and Information Engineering, National Ilan University, Taiwan. Dr. Hsia is the Chapter Chair of IEEE Young Professionals Group, Taipei Section, and the Director of the IET Taipei Local Network. He serves as Associate Editor of the Journal of Imaging Science and Technology and the Journal of Computers. His research interests include DSP IC Design, Multimedia Signal Processing, and Cognitive Learning. He can be contacted at email: chhsia625@gmail.com.