Reversible Image Processing via Reversible Data Hiding

Dongdong Hou[1], Weiming Zhang[2], Zihao Zhan[3], Ruiqi Jiang[4], YangYang[5] and Nenghai Yu[6]
School of Information Science and Technology, University of Science and Technology of China, Hefei, China.
{ Email: [1]houdd@mail.ustc.edu.cn, [2]zhangwm@ustc.edu.cn, [3]dulldoll@mail.ustc.edu.cn, [4]jrq123@mail.ustc.edu.cn, [5]skyyang@mail.ustc.edu.cn, [6]ynh@ustc.edu.cn.}

***Abstract*—Image processing is a very popular and widely used technique. In this paper, we propose a framework for realizing reversible image processing, which enables that the users can return the processed image to the original copy without loss. In the proposed method, the original image is firstly processed to get the desired target image by a classic image processing method. Then the original image is reversibly processed according to the transition probability matrix, getting the processed image similar to the target image. We take histogram equalization and gamma transform as examples to show that the proposed method can realize reversible image processing and achieve nearly the same processing effect as done by irreversible image processing tools.**

***Index Terms*—reversible image processing, reversible data hiding, histogram equalization, gamma transform.**

I. Introduction

Nowadays, image is the most popular media on the Internet, and image processing [1] becomes a widely used technique by which people process the image to their desired result with various kinds of tools.

Although many image processing methods have been proposed, almost all of them are not reversible, which means that the user cannot restore the original image from the processed image without loss. Such irreversibility may cause great inconvenience in some applications. For instance, in the applications of medical or military image processing, the original image is very important which cannot be lost. If the users want to keep the original image, they must save or transmit the original copy with the processed image together, which will cost more storage space or communication bandwidth. Therefore, reversible image processing (RIP) technique is desired.

As shown in Fig. 1, RIP guarantees that the processed image can be returned to the original copy without loss. Hence, the image owners can get both the desired image and the original image without saving the original copy. RIP is a novel topic for image processing, and few related works have been done in this field to the best of our knowledge.

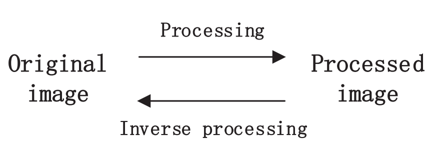

Fig. 1. Reversible image processing.

One related technology is "reversible data hiding with contrast enhancement" [2]–[5], by which we can get an enhanced image after hiding some watermarks, and losslessly reconstruct the cover image from the watermarked image. However, the enhancing performance is limited by the data hiding manner, so the user cannot get the desired result as done by the classic contrast enhancement methods. Another related work is the visible mosaic image transformation (VMIT) [6]–[8], which transforms one image to a given target image and gets a mosaic image similar to the target image. Both VMIT and RIP transform the original image to a target image, so RIP can be viewed as a special case of image transformation. However, the visual quality of the mosaic image generated by VMIT is relatively poor and cannot satisfy the needs of image processing applications, especially for medical or military images.

In the present paper, we propose a framework of RIP. Firstly the correlation between the original image and the target image obtained with an image processing tool is explored, by which the original image can be efficiently compressed. Then the compressed data together with some parameters are reversibly embedded into the target image with a reversible data hiding (RDH) scheme [9], which yields the ultimate processed image. Note that, by RDH the receiver can reconstruct the cover image without loss after extracting the hidden message, which enables the proposed image processing scheme to be reversible. We take histogram equalization and gamma transform as examples to show that the proposed method can realize reversible image processing and achieve nearly the same processing effect as done by irreversible image processing tools.

The rest of the paper is organized as follows. Section II elaborates the proposed scheme of RIP. The experimental results on applications for reversible histogram equalization (RHE) and reversible gamma transform (RGT) are given in Section III, and finally the paper is concluded with a discussion in Section IV.

II. Reversible image processing

A. The Framework of RIP

Assume the original image to be $\mathbf{X} = (x_1, x_2, \cdots, x_N)$ with the histogram $\mathbf{H_Q} = (h_{q_1}, \cdots, h_{q_m})$, and the target image to be $\mathbf{Y} = (y_1, y_2, \cdots, y_N)$ with the histogram $\mathbf{H_T} = (h_{t_1}, \cdots, h_{t_n})$, where "$q_i$" and "$t_j$" are the pixel values (bins in histogram). We use h_{q_i} and h_{t_j} to represent the number of

978-1-5090-4165-7/16/$31.00 ©2016 IEEE

pixel value "q_i" in $\mathbf{X}$ and the number of pixel value "t_j" in $\mathbf{Y}$ respectively.

In this paper, an image processing algorithm is assumed as a map from $\{q_1, \cdots, q_m\}$ to $\{t_1, \cdots, t_n\}$ such that

$$t_j = f(q_i), i = 1, 2, \cdots, m, \tag{1}$$

where $t_j \in \{t_1, \cdots, t_n\}$.

To do RIP, we process the original image to get the desired target image by using an image processing algorithm or software firstly, and then explore the correlation between such two images, with which we compress the original image to generate the accessorial information for recovery. Finally, we get the processed image by embedding the accessorial information into the target image with an RDH scheme. The flowchart of the proposed RIP method comparing with some classic image processing methods is shown as Fig.2.

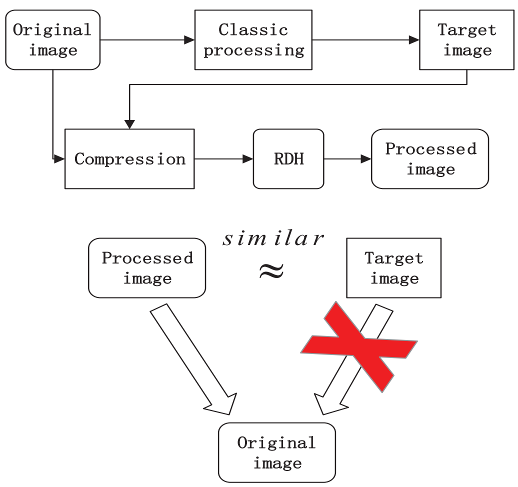

Fig. 2. The flowchart of the proposed RIP method comparing with some classic image processing methods.

B. The Transition Probability Matrix for RIP

To recover $\mathbf{X}$ from $\mathbf{Y}$ without loss, the direct way is to compress $\mathbf{X}$, and then reversibly embed the compressed $\mathbf{X}$ denoted by $O(\mathbf{X})$ into $\mathbf{Y}$ to generate $\mathbf{Y}'$. Usually the amount of the accessorial information (AAI) for recovering $\mathbf{X}$ is large, which may introduce two problems. One is that the RDH scheme cannot reach enough capacity to accommodate these accessorial information. The other is that the more information is embedded into $\mathbf{Y}$, the worse quality of $\mathbf{Y}'$ will result in.

However, the image $\mathbf{Y}$ usually is very similar to $\mathbf{X}$ when $\mathbf{Y}$ is got from $\mathbf{X}$ via an image processing algorithm. Thus $\mathbf{X}$ can be efficiently compressed with the help of $\mathbf{Y}$ and the amount of $O(\mathbf{X})$ will be greatly reduced. In this case, the least AAI (denoted by L) for recovering $\mathbf{X}$ is

$$L = H(\mathbf{X}|\mathbf{Y}) * N, \tag{2}$$

and the average AAI needed for recovering each pixel is

$$\rho = L/N = H(\mathbf{X}|\mathbf{Y}). \tag{3}$$

Assume that the transition probability matrix from $\mathbf{Y}$ to $\mathbf{X}$ is $P_{\mathbf{X}|\mathbf{Y}}$. To compress $\mathbf{X}$ based on $\mathbf{Y}$, i.e., compress $\mathbf{X}$ according to $P_{\mathbf{X}|\mathbf{Y}}$. The transition probability matrix $P_{\mathbf{X}|\mathbf{Y}}$ must be recorded as one part of the accessorial information for restoring the original image.

The transition probability matrix $P_{\mathbf{X}|\mathbf{Y}}$ can be calculated by observing $\mathbf{X}$ and $\mathbf{Y}$, but such matrix is usually hard to be compressed. However, since

$$P_{\mathbf{X}|\mathbf{Y}} = \frac{P_{\mathbf{X}} P_{\mathbf{Y}|\mathbf{X}}}{P_{\mathbf{Y}}}, \tag{4}$$

and $P_{\mathbf{Y}}$ can be calculated from $\mathbf{Y}$ directly. To record $P_{\mathbf{X}|\mathbf{Y}}$, we only need to record $P_{\mathbf{Y}|\mathbf{X}}$ and $P_{\mathbf{X}}$. The matrix $P_{\mathbf{Y}|\mathbf{X}}$ is usually sparse for many image processing methods and is much easier to be compressed than $P_{\mathbf{X}|\mathbf{Y}}$.

To get $P_{\mathbf{Y}|\mathbf{X}}$, we only need to find the position of each "q_i" in $\mathbf{X}$ and then observe the pixel values at the same position in $\mathbf{Y}$. Denote the index set of symbol "q_i" in $\mathbf{X}$ by $\mathbf{IX}_{q_i}$, and extract a sub-sequence $\mathbf{Y}_{q_i}$ from $\mathbf{Y}$ according to $\mathbf{IX}_{q_i}$, such that $\mathbf{Y}_{q_i} = \{y_k | y_k \in \mathbf{Y} \text{ and } k \in \mathbf{IX}_{q_i}\}$. Thus the pixels of image $\mathbf{Y}$ are divided into m disjoint sub-sequences $\mathbf{Y}_{q_i}$ with the length h_{q_i} for $i = 1, 2, \cdots, m$. Assume that the number of "t_j" in $\mathbf{Y}_{q_i}$ is h_{ij}, and the probability of "q_i" to "t_j" is $\mathbf{P}_{t_j|q_i} = \frac{h_{ij}}{h_{q_i}}$, then the transition probability matrix $P_{\mathbf{Y}|\mathbf{X}}$ is given as

$$P_{\mathbf{Y}|\mathbf{X}} = (P_{\mathbf{Y}|q_1}, P_{\mathbf{Y}|q_2}, \cdots, P_{\mathbf{Y}|q_m})^{\mathrm{T}}, \tag{5}$$

where $P_{\mathbf{Y}|q_i} = (P_{t_1|q_i}, P_{t_2|q_i}, \cdots, P_{t_n|q_i}), i = 1, 2, \cdots, m$. Similarly,

$$P_{\mathbf{X}|\mathbf{Y}} = (P_{\mathbf{X}|t_1}, P_{\mathbf{X}|t_2}, \cdots, P_{\mathbf{X}|t_n})^{\mathrm{T}}, \tag{6}$$

where $P_{\mathbf{X}|t_j} = (P_{q_1|t_j}, P_{q_2|t_j}, \cdots, P_{q_m|t_j}), j = 1, 2, \cdots, n$.

$P_{\mathbf{X}}$ and $P_{\mathbf{Y}}$ can be obtained by normalizing $\mathbf{H_Q}$ and $\mathbf{H_T}$ respectively. To compress $\mathbf{H_Q}$, the differential pulse-code modulation encoder utilized in [10] can be adopted. However, as for some histogram-based image processing methods, such as histogram equalization and gamma transform, $\mathbf{H_Q}$ can be compressed by $\mathbf{H_T}$ and $P_{\mathbf{Y}|\mathbf{X}}$ more effectively. Indeed, from $P_{\mathbf{Y}|\mathbf{X}}$ we can see which bins of $\mathbf{H_Q}$ are transformed to bin "t_j", $j = 1, 2, \cdots, n$, but cannot get the height of each bin. Assume that, by observing $P_{\mathbf{Y}|\mathbf{X}}$ we see that bin "t_j" is generated from g bins of $\mathbf{H_Q}$ denoted by $(q_{1,j}, q_{2,j}, ..., q_{g,j})$. The height of each bin, $h_{q_{k,j}}$ $(k = 1, 2, ..., g)$, needs to be recorded, which however may be large. To compress $h_{q_{k,j}}$ for $k = 1, 2, ..., g$, since

$$h_{t_j} = \sum_{k=1}^{g} h_{q_{k,j}}, j = 1, 2, \cdots, n, \tag{7}$$

we only need to calculate and record the residuals as

$$\Delta h_{q_{k,j}} = h_{q_{k,j}} - round(h_{t_j}/g), k = 1, 2, \cdots, g-1. \tag{8}$$

Obviously, with $\Delta \mathbf{h}_{t_j} = (\Delta h_{q_{1,j}}, \Delta h_{q_{2,j}}, ..., \Delta h_{q_{g-1,j}})$ and h_{t_j}, we can recover $(h_{q_{1,j}}, h_{q_{2,j}}, ..., h_{q_{g,j}})$. Therefore, the AAI for recording $\mathbf{H_Q}$ and $P_{\mathbf{X}}$ can be much reduced by recording $\Delta \mathbf{H} = (\Delta \mathbf{h}_{t_1}, \Delta \mathbf{h}_{t_2}, ..., \Delta \mathbf{h}_{t_n})$ instead.

C. Accessorial Information for RIP

Next, we show how to generate the accessorial information for RIP by compressing $\mathbf{X}$ according to $\mathbf{Y}$ and $P_{\mathbf{X}|\mathbf{Y}}$.

Denote the index set of all pixels "t_j" in $\mathbf{Y}$ by $\mathbf{IY}_{t_j}$, and extract a sub-sequence $\mathbf{X}_{t_j}$ from $\mathbf{X}$ according to $\mathbf{IY}_{t_j}$, such that $\mathbf{X}_{t_j} = \{x_k | x_k \in \mathbf{X} \text{ and } k \in \mathbf{IY}_{t_j}\}$, by which the pixels of image $\mathbf{X}$ are divided into n disjoint sub-sequences $\mathbf{X}_{t_j}$ with the length h_{t_j} for $j = 1, 2, \cdots, n$. The pixels in the sub-sequence $\mathbf{X}_{t_j}$ satisfy the distribution $P_{\mathbf{X}|t_j}$, so according to $P_{\mathbf{X}|t_j}$ we compress $\mathbf{X}_{t_j}$ with an entropy coder $Comp()$, such as the arithmetic coder, getting

$$O(\mathbf{X}_{t_j}) = Comp(\mathbf{X}_{t_j}, P_{\mathbf{X}|t_j}), j = 1, 2, \cdots, n. \quad (9)$$

We concatenate these compressed sub-sequences as follows,

$$O(\mathbf{X}) = O(\mathbf{X}_{t_1}) || O(\mathbf{X}_{t_2}) || \cdots || O(\mathbf{X}_{t_n}). \quad (10)$$

Apart from $O(\mathbf{X})$, the compressed transition probability matrix $P_{\mathbf{Y}|\mathbf{X}}$ and $\Delta\mathbf{H}$ are also the accessorial information, which need to be embedded into $\mathbf{Y}$ with an RDH method. In the present paper, we propose to embed the accessorial information into $\mathbf{Y}$ to yield the ultimate processed image $\mathbf{Y}'$ with the method "recursive histogram modification (RHM)" presented in [9], which is proved to be an optimal coding scheme for RDH.

To restore the original image $\mathbf{X}$ from $\mathbf{Y}'$, we firstly reconstruct $\mathbf{Y}$ from $\mathbf{Y}'$ after extracting $O(\mathbf{X})$, $P_{\mathbf{Y}|\mathbf{X}}$ and $\Delta\mathbf{H}$ by the decoding algorithm of the RDH method. From $\mathbf{Y}$, we calculate the histogram $\mathbf{H_T}$, which combining with $\Delta\mathbf{H}$ and $P_{\mathbf{Y}|\mathbf{X}}$ is used to recover $\mathbf{H_Q}$. After working out $\mathbf{P_X}$ and $\mathbf{P_Y}$ by normalizing $\mathbf{H_Q}$ and $\mathbf{H_T}$ respectively, we can obtain $P_{\mathbf{X}|\mathbf{Y}}$ according to Eq. (4) with $\mathbf{P_X}$, $\mathbf{P_Y}$ and $P_{\mathbf{Y}|\mathbf{X}}$ as parameters.

Then we decompress $O(\mathbf{X}_{t_j})$ with the entropy decoder $Decomp()$ to recover $\mathbf{X}_{t_j}$ according to $\mathbf{P}_{\mathbf{X}|t_j}$, and substitute the pixels equal to "t_j" in $\mathbf{Y}$ with $\mathbf{X}_{t_j}$. The recovery process can be formulated as

$$(\mathbf{X}_{t_j}, \mathbf{B}_{j+1}) = Decomp(\mathbf{B}_j, h_{t_j}, P_{\mathbf{X}|t_j}), j = 1, 2, \cdots, n, \quad (11)$$

where $\mathbf{B}_1 = O(\mathbf{X})$, and $\mathbf{B}_{j+1}$ is the rest information of $\mathbf{B}_j$ apart from the utilized message. We decompress the message of $\mathbf{B}_j$ according to $P_{\mathbf{X}|t_j}$ until generating h_{t_j} pixels. Indeed, the message utilized to generate the h_{t_j} pixels composes $O(\mathbf{X}_{t_j})$, so $\mathbf{B}_{j+1}$ is the rest information of $\mathbf{B}_j$ apart from $O(\mathbf{X}_{t_j})$, and $\mathbf{B}_{n+1} = 0$.

A simple example of RIP is shown in Fig. 3, in which we label $\mathbf{IY}_1$ to be "red", and $\mathbf{IY}_3$ to be "white". We can see that the pixel "1" and "2" in $\mathbf{X}$ are changed to the pixel "1" in $\mathbf{Y}$ by observing $P_{\mathbf{Y}|\mathbf{X}}$. The number of pixel "1" in $\mathbf{Y}$ is 10, so $\Delta\mathbf{h}_1 = \{3 - 10/2\} = \{-2\}$ is enough to record the number of pixel "1" and "2" in $\mathbf{X}$, and in the same way we use $\Delta\mathbf{h}_3 = \{4 - 6/2\} = \{1\}$ to record the number of pixel "3" and "4" in $\mathbf{X}$. We compress $P_{\mathbf{Y}|\mathbf{X}}$, $\Delta\mathbf{H}$, $\mathbf{X}_1$, $\mathbf{X}_3$ and then embed them into $\mathbf{Y}$ to get $\mathbf{Y}'$. We restore $\mathbf{Y}$ from $\mathbf{Y}'$ after extracting these accessorial information, then calculate $P_{\mathbf{X}}$ through $P_{\mathbf{Y}|\mathbf{X}}$, $\mathbf{H_T}$ and $\Delta\mathbf{H}$, and calculate $P_{\mathbf{X}|\mathbf{Y}}$ through $P_{\mathbf{Y}|\mathbf{X}}$, $P_{\mathbf{X}}$ and $P_{\mathbf{Y}}$. Finally we decompress $O(\mathbf{X}_1)$ to restore $\mathbf{X}_1$ according to $\{3/10, 7/10, 0, 0\}$, and decompress $O(\mathbf{X}_3)$ to restore $\mathbf{X}_3$ according to $\{0, 0, 2/3, 1/3\}$.

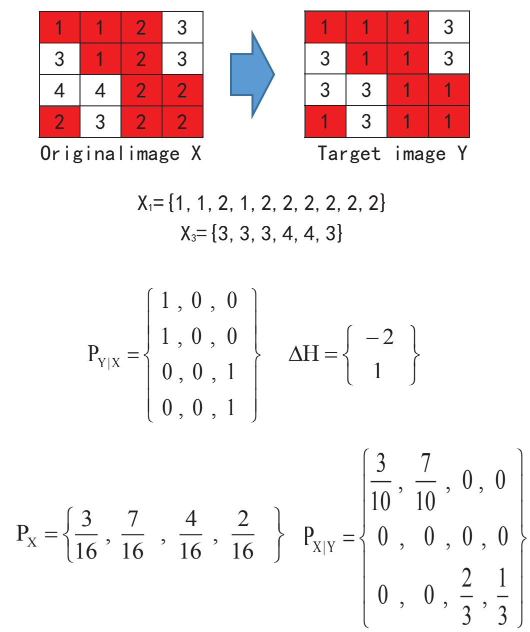

Fig. 3. A simple example of RIP.

D. Algorithm of RIP

Based on the above discussion, we describe the detailed processes of RIP in Algorithm. 1.

Algorithm 1 Reversible Image Processing

Reversibly Processing the Image

- **1**. Process the original image $\mathbf{X}$ to be the desired target image $\mathbf{Y}$ through some image processing tools.
- **2**. Calculate the transition matrices $P_{\mathbf{Y}|\mathbf{X}}$, $P_{\mathbf{X}|\mathbf{Y}}$ and $\Delta\mathbf{H}$ by observing $\mathbf{X}$ and $\mathbf{Y}$.
- **3**. Divide $\mathbf{X}$ into n disjointed sub-sequences $\mathbf{X}_{t_j}$ with the length h_{t_j}, and compress $\mathbf{X}_{t_j}$ according to $P_{\mathbf{X}|t_j}$ for $j = 1, 2, \cdots, n$ to generate $O(\mathbf{X})$.
- **4**. Compress the transition probability matrix $P_{\mathbf{Y}|\mathbf{X}}$ and $\Delta\mathbf{H}$, and collect them with $O(\mathbf{X})$ together as the accessorial information, then embed these accessorial information into $\mathbf{Y}$ with the RHM [9] to generate the ultimate processed image $\mathbf{Y}'$.

Recovering the Original Image

- **1**. From $\mathbf{Y}'$, extract $P_{\mathbf{Y}|\mathbf{X}}$, $\Delta\mathbf{H}$ and $O(\mathbf{X})$ and restore $\mathbf{Y}$ with the decoding method of RHM.
- **2**. Calculate $P_{\mathbf{X}}$ through $P_{\mathbf{Y}|\mathbf{X}}$, $\mathbf{H_T}$ and $\Delta\mathbf{H}$, then calculate $P_{\mathbf{X}|\mathbf{Y}}$ through $P_{\mathbf{Y}|\mathbf{X}}$, $P_{\mathbf{X}}$ and $P_{\mathbf{Y}}$.
- **3**. Decompress $O(\mathbf{X}_{t_j})$ according to $\mathbf{P}_{\mathbf{X}|t_j}$ with the decompression algorithm to recover $\mathbf{X}_{t_j}$, and substitute the symbol "t_j" in $\mathbf{Y}$ with $\mathbf{X}_{t_j}$ for $j = 1, 2, \cdots, n$, which yields the original image $\mathbf{X}$.

III. EXPERIMENTAL RESULTS

In this section, taking histogram equalization and gamma transform as examples, we apply the proposed scheme to realize RIP. In additional to the peak signal to noise ratio (PSNR) and the structural similarity (SSIM) [11], we also adapt the earth mover's distance (EMD) [12] to appraise the similarity between the histograms of the reversibly processed image and the target image, because both of the above examples are histogram-based image processing. The smaller the EMD, the more similarity exists between such two histograms.

We embed the accessorial information with the RDH coding method, RHM [9]. Note that most of RDH methods use the prediction errors of pixels as the host sequence to embed data, but we directly apply RHM to the pixels rather than the pixels' prediction errors, by which we can also get good performance for such two examples.

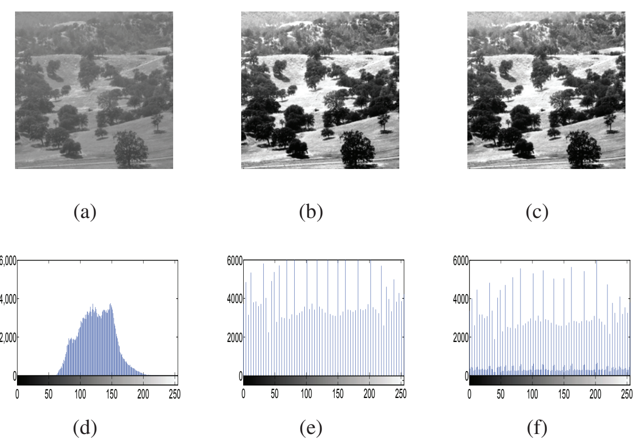

Fig. 4. (a) Original image. (b) Target image. (c) Processed image with SSIM=0.9997 and PSNR=55.88 with respect to the target image. (d) The histogram of the original image. (e) The histogram of the target image. (f) The histogram of the processed image with EMD=0.1678 with respect to the target histogram (Fig. 4(e)).

Histogram equalization increases the global contrast of image by effectively spreading out the most frequent intensity values. As shown in Fig. 4, we enhance the contrast of the original image (Fig. 4(a)) to generate the target image (Fig. 4(b)) by PhotoShop. It can be seen that the reversibly processed image (Fig. 4(c)) got by the proposed method has the same enhancing effect as the target image. Although the AAI for reconstruction reaches 0.8228 bits per pixel (bpp), the processed image can still keep good quality, achieving SSIM=0.9997 and PSNR=55.88 with respect to the target image. The value of EMD is as small as 0.1678, which means that the histogram of the processed image is similar to that of the target image.

Gamma transform can enhance not only the contrast but also the details of image, which uses the gamma coefficient "γ" to determine to enhance the shadow region or the highlighted region. In Fig. 5, the target image (Fig. 5(b)) is got from the original image (Fig. 5(a)) by using the MATLAB function "$imadjust()$" with "$\gamma = 1.2$". Since the AAI is only 0.0636 bpp in this example, the ultimate processed image (Fig. 5(c)) got by the proposed method is nearly the same as the target image.

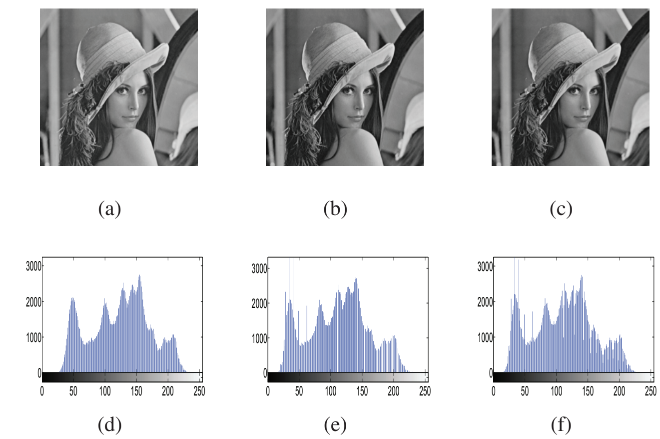

Fig. 5. (a) Original image. (b) Target image. (c) Processed image with SSIM=0.9998 and PSNR=62.87 with respect to the target image. (d) The histogram of the original image. (e) The histogram of the target image. (f) The histogram of the processed image with EMD=0.0334 with respect to the target histogram.

To verify the feasibility of the proposed RIP, we execute RHE and RGT on 100 images [13], and the average results are listed in Table. I. From Table. I we can see that the AAI for recovering $\mathbf{X}$ from $\mathbf{Y}$ almost reaches the lower bound, $H(\mathbf{X}|\mathbf{Y})$. Table. I shows that RHE can reach almost the same effect as the traditional histogram equalization with PSNR=53.52 and SSIM=0.9978. For gama transform, the reversible scheme can achieve more close effect to the traditional method with PSNR larger than 56 and SSIM also larger than 0.99. The EMD is smaller than 0.6 for histogram equalization and is smaller than 0.2 for gama transform, which means that the processed image and the corresponding target image have a similar histogram.

TABLE I
AVERAGE RESULTS ON 100 TEST IMAGES (THE UNIT OF BOTH THE SECOND AND THIRD COLUMNS IS BPP).

RIP	$H(\mathbf{X}\|\mathbf{Y})$	AAI	PSNR	SSIM	EMD
RHE	0.9591	0.9678	53.52	0.9978	0.5303
RGT (γ=1.2)	0.0797	0.0863	56.98	0.9981	0.1450
RGT (γ=0.8)	0.0923	0.0990	56.67	0.9975	0.1969

IV. CONCLUSION

In this paper, we propose a framework to realize RIP that is a new topic for image processing. So far, we only take simple image processing, histogram equalization and gamma transform, as examples to show the feasibility of the proposed scheme. For some complex image processing methods, the amount of the accessorial information for recovery may be large, which will greatly degrade the quality of the processed image. In our future work, we will study how to extend the proposed method to other kinds of image processing algorithms.

REFERENCES

[1] R. C. Gonzalez, *Digital image processing*. Pearson Education India, 2009.

[2] H.-T. Wu, J.-L. Dugelay, and Y.-Q. Shi, "Reversible image data hiding with contrast enhancement," *IEEE Signal Processing Letters*, vol. 22, no. 1, pp. 81–85, 2015.

[3] H.-T. Wu, J. Huang, and Y.-Q. Shi, "A reversible data hiding method with contrast enhancement for medical images," *Journal of Visual Communication and Image Representation*, vol. 31, pp. 146–153, 2015.

[4] G. Gao and Y.-Q. Shi, "Reversible data hiding using controlled contrast enhancement and integer wavelet transform," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2078–2082, 2015.

[5] Y. Yang, W. Zhang, D. Liang, and N. Yu, "Reversible data hiding in medical images with enhanced contrast in texture area," *Digital Signal Processing*, vol. 52, pp. 13–24, 2016.

[6] I.-J. Lai and W.-H. Tsai, "Secret-fragment-visible mosaic image–a new computer art and its application to information hiding," *IEEE Trans. Information Forensics and Security*, vol. 6, no. 3, pp. 936–945, 2011.

[7] Y.-L. Lee and W.-H. Tsai, "A new secure image transmission technique via secret-fragment-visible mosaic images by nearly reversible color transformations," *IEEE Trans. Circuits Syst.* & *Video Technol.*, vol. 24, no. 4, pp. 695–703, 2014.

[8] D. Hou, W. Zhang, and N. Yu, "Image camouflage by reversible image transformation," *Journal of Visual Communication and Image Representation*, vol. 40, Part A, pp. 225–236, 2016.

[9] W. Zhang, X. Hu, X. Li, and N. Yu, "Recursive histogram modification: establishing equivalency between reversible data hiding and lossless data compression," *IEEE Trans. Image Processing*, vol. 22, no. 7, pp. 2775–2785, 2013.

[10] W. Zhang, X. Hu, X. Li, and Y. Nenghai, "Optimal transition probability of reversible data hiding for general distortion metrics and its applications," *IEEE Trans. Image Processing*, vol. 24, no. 1, pp. 294–304, 2015.

[11] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[12] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International journal of computer vision*, vol. 40, no. 2, pp. 99–121, 2000.

[13] Related images of the experiments, [Online]. Available: http://decsai.ugr.es/cvg/dbimagenes/g512.php.