Reversible Data Hiding with Automatic Brightness Preserving Contrast Enhancement

Suah Kim*, Rolf Lussi*, Xiaochao Qu, Fangjun Huang, Hyoung Joong Kim

***Abstract*—This is not the final published version, it is an accepted version of the paper.** **Reversible data hiding with automatic contrast enhancement methods provide an interoperable way to reduce the storage requirement for automatic image enhancement applications: original image can be recovered from the enhanced image without any additional information. Unlike the previous work, where the goal was to maximize the contrast, the proposed method increases the contrast to an appropriate level using an idea called brightness preservation. This is achieved by using an adaptive bin selection process based on the original brightness. Extensive experimental results verify that the enhanced images produced using the proposed method are visually and quantitatively superior than the existing work.**

***Index Terms*—Automatic brightness preserving contrast enhancement, reversible data hiding, histogram shifting, reversible contrast enhancement, automatic image enhancement**

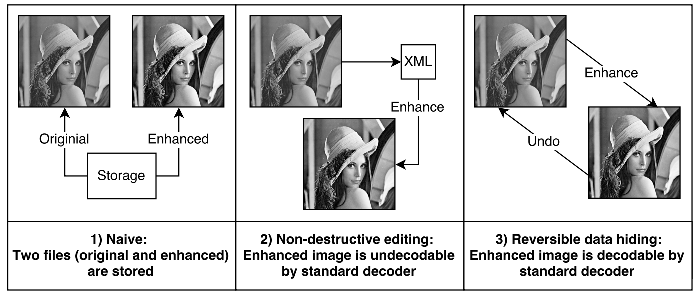


Fig. 1. Existing methods of managing original and enhanced image.

I. Introduction

THE number of images uploaded for backup and sharing has increased with the widespread adoption of smartphones and tablets. Many of the uploaded images benefit from enhancements, such as contrast enhancement, gamma correction, and sharpening. However, average users either lack the technical skills or find the process laborious. Automatic image enhancement (AIE) improves user experience by offering automatic and reversible image enhancement solution.

AIE works in the following way: when a user uploads an image, the image is enhanced with no user interactions, but if the user is not satisfied with the enhancement at any time, the original image can be recovered.

Existing AIEs can be implemented by

1) Naively: backup the original (see Fig. 1 left box) or
2) Using non-destructive editing [1] approach: only the original image and a small enhancement record file are kept (see Fig. 1 middle box) or
3) Using reversible data hiding: the original image can be perfectly recovered from the watermarked/enhanced image without any additional data (see Fig. 1 right box).

The first two approaches have been commercialized. Picasa's[1] AIE is an example of the naive approach, where the enhanced image is saved as a JPEG image and the original image is backed up in a separate folder. For non-destructive editing, services provided by Apple's iPhoto[2] and Google Photos[3] automatically enhance the image and keep the enhancement record in a proprietary file format [2].

These two approaches are disadvantageous for AIE. The first approach has the big storage requirement because the original and the enhanced images need to be backed up. In the second approach, although the storage requirement is reduced, only the proprietary decoder can decode the enhancement record. Furthermore, to view the enhanced image, the enhancement must be applied to the original image every time, increasing the total computation per usage. This is clearly not ideal for the smartphones or tablets, where images are frequently viewed and shared across different applications. Thus, much of the support has been discontinued for mobile applications.

The third approach by Kim et al. [3], automatic contrast enhancement using reversible data hiding, is a recent contribution to AIE for addressing the disadvantages. In their approach, the original image does not have to be backed up because the enhancement is perfectly reversible without using additional data: enhanced image is decodable using the standard decoder while the original image is recovered using a special decoder. However, one disadvantage of this approach is that only few types of enhancements are available now.

The proposed method improves the work by Kim et al.

This is not the final published version, it is an accepted version of the paper.
*The first two authors equally contributed to this work and Hyoung Joong Kim is the corresponding author. S. Kim, R. Lussi, and H. J. Kim are with Graduate School of Information Security, Korea University, Seoul, South Korea (email: suahnkim@gmail.com, rolf.lussi@gmail.com, khj-@korea.ac.kr). X. Qu is with MTlab (Meitu, Inc.), China (email: qxc@meitu.com). F. Huang is with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, GD510006, China (email: huangfj@mail.sysu.edu.cn).

[1] Edit photos in Picasa - Picasa and Picasa Web Albums Help, https://support.google.com/picasa/answer/156342?hl=en&rd=1, accessed July 21, 2016

[2] iPhoto for iOS (iPhone): Automatically enhance a photo, http://support.apple.com/kb/PH3258, accessed January 7, 2015

[3] Auto-enhance, https://support.google.com/plus/answer/3338435?hl=en-GB, accessed January 7, 2015

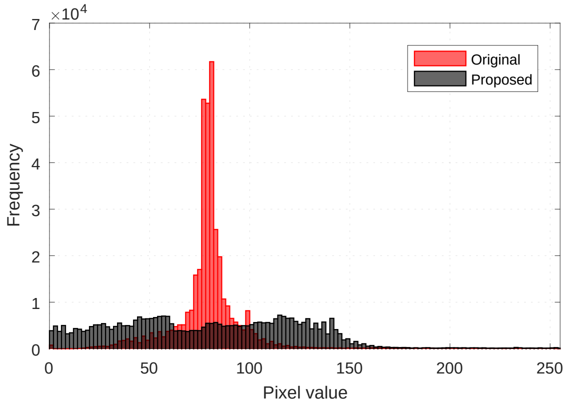


Fig. 2. Histogram equalization effect on pixel frequency histogram.

[3] by incorporating an idea called the brightness preservation (BP). BP is used to limit the overenhancement effect which are readily found in images enhanced with global histogram equalization-based methods. In the context of contrast enhancement, the overenhancement effect as the name suggests, results in unnaturally looking image where the increased contrast makes the parts of the image too dark or too light. More details on the previous work on BP is provided in the Appendix A. Additional to providing BP, the proposed method also does not reduce the contrast when applied to image with already high contrast, making the proposed method a good candidate for AIE tool.

The remainder of this paper is organized as follows. Section II discusses the related work in reversible data hiding with contrast enhancement methods. Section III describes the proposed method. Section IV describes the experimental results and verifies the improvements. Section V concludes the paper.

II. Related Work in Reversible Data Hiding with Automatic Contrast Enhancement

Reversible data hiding (RDH) is traditionally used as an interoperable method for hiding data with an ability to recover the original without any additional data. Much of the previous work focused on improving the embedding capacity while maximizing the PSNR. On the other hand, the goal of the reversible data hiding with contrast enhancement is to produce watermarked images with good contrast.

RDH with contrast enhancement methods are based on the principle of the histogram equalization. Fig. 2 shows the original pixel frequency histogram of Kodak image 4 and the equalized histogram after using the proposed method. The figure clearly shows that the proposed method spreads the frequent pixel values along the nearby neighbors while shifting the neighboring pixels close to the pixel boundary values 0 and 255, which in turn artificially increases the range of pixels perceived by the eyes, *i.e.,* increases the contrast.

The first work to suggest RDH for contrast enhancement was by Wu et al. [4] (referred to as RDHCE from here on). Several variations on the work have focused on fine tuning the enhancement effect [5–7], and the localization of the enhancement and its application as an extension to medical

(a) Original (b) RDHCE [4] (c) ACERDH [3] (d) Proposed

Fig. 3. Reversible data hiding with contrast enhancement used on Kodak image 4. Intensity mismatched artifacts are present around the scarf near the left ear and eye lashes for RDHCE [4].

imaging [8–11]. However, all except for the work by Kim et al. [3], require manual setting of the parameters, making them unsuitable for AIE.

Kim et al. [3] first proposed a reversible data hiding with automatic contrast enhancement method for AIE (referred to as ACERDH from here on). They showed that a reversible data hiding based contrast enhancement provides file saving feature for AIE. This is because original image can be recovered directly from the enhanced image.

The remainder of this subsection discusses RDHCE and ACERDH in more details.

A. RDHCE

RDHCE uses multiple repetitions of histogram shifting to achieve histogram equalization. In each repetition, the two highest bins of the histogram are chosen for data hiding and their respective neighboring bins are shifted towards their respective boundary bins (0 and 255 for 8 bit case) to create empty bins for data hiding. The highest bins are used to embed the payload bits, where if the payload is 1, the pixel is shifted to its neighbor, and if it is 0, it is not shifted.

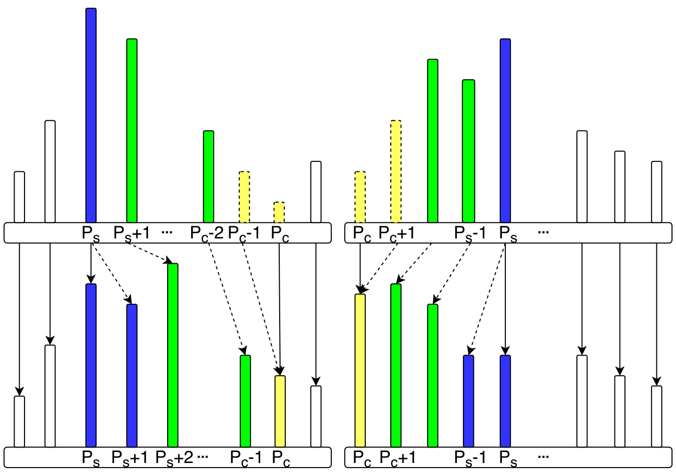
During the histogram shifting, underflow or overflow is possible: pixels are shifted to unencodable bins of below 0 or above 255. To prevent this, a preprocessing step is applied in the beginning, where boundary bins are combined with their neighboring bins, so that the bins are still encodable after histogram shifting.

However, with increase in number of repetitions, number of bins that are combined is increased as well, which creates a visual distortion called intensity mismatched artifact [3]. Fig. 3b shows an example of the intensity mismatched artifacts from preprocessing for RDHCE. This artifact is usually visible only with a large number of histogram shifting, but for some specific images, it is visible even for a very small number of repetitions [3].

In conclusion, RDHCE requires preprocessing step, which results in intensity mismatched artifacts and the optimal number of repetitions of histogram shifting cannot be determined, making it unsuitable for applications in AIE.

B. ACERDH

ACERDH is a recent contribution to reversible data hiding with automatic contrast enhancement. ACERDH uses multiple



(a) RHS (b) LHS

Fig. 4. Visualization of RHS and LHS on the pixel frequency histogram. Arrows indicate the movement of the pixel values due to RHS and LHS.

repetitions of uni-directional histogram shifting using a concurrent location map generation, which generates a location map for every repetition of histogram shifting to eliminate the intensity mismatched artifacts that occur in RDHCE.

For the histogram shifting step, one splitting bin (the most frequent pixel values) and one combining bin (the least frequent pixel values) are chosen. The splitting bin is used for data hiding, while the combining bin is combined with its neighbor to create a space for the new bin resulting from the split. This setup allows for the small bins to be combined regardless of their distance from the boundary bins. In other words, unlike RDHCE where distortion from combining the bins is localized near the boundary bins, ACERDH spreads the distortion around the whole pixel range. In addition, the number of repetitions is not set manually, but is adaptively chosen according to the image content to maximize the histogram equalization effect.

In conclusion, ACERDH provides similar contrast enhancement results as global histogram equalization methods, which is beneficial for enhancing low contrast images. However, it is not always suitable for everyday use because it may excessively increase the contrast.

III. Proposed Method

In the proposed method, a classical contrast enhancement primitive called brightness preservation is incorporated to reduce overenhancement effects and reduce the overall computation. The proposed method involves following four steps:

- Adaptively select bins based on the original brightness (Section III-C).
- Generate side information for reversibility and prepends it to the payload (Section III-B and III-E).
- Apply histogram shifting to embed the payload (Section III-A).
- Repeat the above three steps until the stop condition is met (Section III-F).

The proposed method applies the four steps in repetitions to achieve contrast enhancement with reversible data hiding.

A. Histogram shifting

For every repetition of the histogram shifting, splitting bin (P_s) and combining bin (P_c) are chosen.

Fig. 4 shows a graphical representation of the right histogram shifting (RHS) and left histogram shifting (LHS). The top two histograms correspond to the histograms before shifting, and the bottom two correspond to the histograms after shifting: This blue bin (P_s) is used for data hiding, the green bins (values between P_s and P_c) are shifted to their neighboring bins, the yellow bins (P_c and $P_c - 1$ or $P_c + 1$) are combined, and finally white bins are unmodified.

The following formally describes RHS and LHS: let p'_i be the ith pixel value, and p''_i be the histogram shifted version of p'_i.

1) RHS:

$$p''_i = \begin{cases} p'_i + b_k & \text{if } p'_i = P_s \\ p'_i + 1 & \text{else if } P_s < p'_i < P_c \\ p'_i & \text{else} \end{cases} \tag{1}$$

2) LHS:

$$p''_i = \begin{cases} p'_i - b_k & \text{if } p'_i = P_s \\ p'_i - 1 & \text{else if } P_c < p'_i < P_s \\ p'_i & \text{else} \end{cases} \tag{2}$$

where $b_k \in \{0, 1\}$ is the kth payload bit. Pixels valued P_s are used for embedding the payload, while pixels between P_s and P_c are shifted. Note that, P_s must be smaller than P_c to apply RHS, while P_s must be bigger than P_c to apply LHS. This fact is used to control the direction of the histogram shifting. The selection process for P_S and P_c is explained in the later subsection.

Finally, before applying next repetition of histogram shifting, current p''_i is set as new p'_i and the steps are repeated all over again. Extending the definition, we refer the collections of p'_i as the reference image, denoted as $(p'_i)_{i=1}^{I}$. Naturally, $(p'_i)_{i=1}^{I} = (p_i)_{i=1}^{I}$ for the very first repetition.

Furthermore, Eq. 1 shows that RHS modifies pixels valued $P_c - 1$ to P_c; pixels valued $P_c - 1$ and P_c are combined. Similarly, Eq. 2 shows LHS combines pixels valued $P_c + 1$ with P_c. To know how to reverse the combining process, a record called location map is generated. The generation of location map is explained in the next subsection.

B. Location map

Before every repetition of histogram shifting is applied, location map $L = (L_1, ..., L_m, ...)$ is generated, where L is the record of which bins were originally P_c and which were its neighbor.

The location map generation is different for RHS and LHS because pixels valued $P_c - 1$ are modified to P_c for RHS, while pixels valued $P_c + 1$ are modified to P_c for LHS. To differentiate each direction of the histogram shifting, we use d:

$$d = \begin{cases} 1 & \text{if } P_s < P_c \text{ (For RHS)} \\ -1 & \text{if } P_s > P_c \text{ (For LHS)} \end{cases} \tag{3}$$

Now that d is defined, instead of discussing separate cases for RHS and LHS, we can simply say histogram shifting combines P_c and $P_c - d$. Thus, L_m is defined as following:

$$L_m = \begin{cases} 0 & \text{if } p_i' = P_c \\ 1 & \text{if } p_i' = P_c - d \end{cases} \tag{4}$$

Note that $L = (L_1, ..., L_m, ...)$ is only defined for pixels valued P_c and $P_c - d$, thus the size of L should be equal to number of pixels equal to P_c and $P_c - d$.

C. Adaptive bin selection

Unlike existing work, the proposed method adaptively selects P_s and P_c such that the histogram shifted image's brightness differs by the original at most 1, *i.e.,* $|B - B'| \leq 1$, where B is the original brightness and B' is the reference image's brightness. The brightness is defined as the average pixel value:

$$B = \frac{1}{I} \sum_{i=1}^{I} p_i \tag{5}$$

Similarly, the brightness of the reference image is:

$$B' = \frac{1}{I} \sum_{i=1}^{I} p_i' \tag{6}$$

To ensure that $|B - B'| \leq 1$ after each repetition of the histogram shifting, the direction of the histogram shifting (RHS or LHS) is chosen based on B and B' values; RHS increases B' by at most +1, whereas LHS decreases B' by at most -1.

The rest of the section describes the selection process for P_s and P_c.

1) P_s selection: P_s is selected first. P_s is defined as the *most* frequent pixel value from pixel range J_{P_s}.

$$P_s = \underset{n \in J_{P_s}}{\arg\max}\, H_n' \tag{7}$$

$$H_n' = \sum_{i=1}^{I} ||p_i' = n|| \tag{8}$$

$$||p_i' = n|| = \begin{cases} 1 & \text{if } p_i' = n \\ 0 & \text{if } p_i' \neq n \end{cases} \tag{9}$$

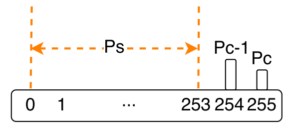
$$J_{P_s} = \begin{cases} [0, 253] & \text{if } B - B' > 0 \\ [2, 255] & \text{if } B - B' < 0 \\ [0, 255] & \text{if } B - B' = 0 \end{cases} \tag{10}$$

where H_n' is the number of pixel values equal to n, and $[A, B]$ is the integer interval between A and B, *i.e.,* $[0, 255]$ represents the set integers from 0 to 255.

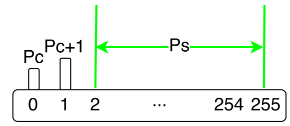
Selection of P_s is restricted by the histogram shifting direction for brightness preservation and the space requirement for P_c and $P_c - d$.

There are three cases (Eq. 10) for determining the histogram shifting direction based on B and B':

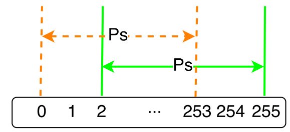
- $B - B' > 0$: *Apply RHS*. Using LHS can decrease B' up to -1, such that $B - B' > 1$.
- $B - B' < 0$: *Apply LHS*. Using RHS can increase B' up to +1, such that $B - B' < -1$.
- $B - B' = 0$: *Apply RHS or LHS*. The direction of the histogram shifting does not matter.



(a) $B - B' > 0$, *apply RHS*



(b) $B - B' < 0$, *apply LHS*



(c) $B - B' = 0$, *apply RHS or LHS*

Fig. 5. Different cases for J_{P_s} for selecting P_s.

For the space requirement, selection of P_s is restricted by Eq. 10 to ensure enough space for P_c and $P_c - d$. Graphical representation of each case is shown in Fig. 5:

- $B - B' > 0$: Fig. 5a shows that $P_s \in [0, 253]$, to allocate enough spaces for $P_c - 1$ and P_c.
- $B - B' < 0$: Fig. 5b shows that $P_s \in [2, 255]$, to allocate enough spaces for P_c and $P_c + 1$.
- $B - B' = 0$: Fig. 5c shows that P_s can be selected without restrictions.

2) P_c selection: P_c is defined as the pixel value with the smallest sum of the pixel valued P_c and its neighbor from pixel range J_{P_c}:

- If $B - B' > 0$ or $P_s < 2$

$$P_c = \underset{n \in J_{P_c}}{\arg\min}\, H_{n-1}' + H_n' \tag{11}$$

$$J_{P_c} = [P_s + 2, 255] \tag{12}$$

- If $B - B' < 0$ or $P_s > 253$

$$P_c = \underset{n \in J_{P_c}}{\arg\min}\, H_n' + H_{n+1}' \tag{13}$$

$$J_{P_c} = [0, P_s - 2] \tag{14}$$

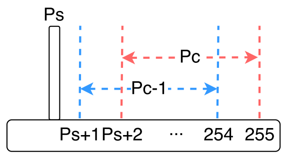
- For all other cases,

$$P_c = \underset{n \in J_{P_c}}{\arg\min} \begin{cases} H_{n-1}' + H_n' & \text{if } n > P_s \\ H_n' + H_{n+1}' & \text{if } n < P_s \end{cases} \tag{15}$$

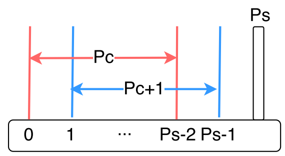
$$J_{P_c} = [0, P_s - 2] \cup [P_s + 2, 255] \tag{16}$$

The selection of P_c requires more attention than P_s. On top of brightness preservation, the location map's size $|L| = H_{P_c}' + H_{P_c - d}'$ needs to be minimized as well. The benefit of minimizing the location map size is that it can increase embedding capacity while reducing the distortion from combining the P_c with $P_c - d$.

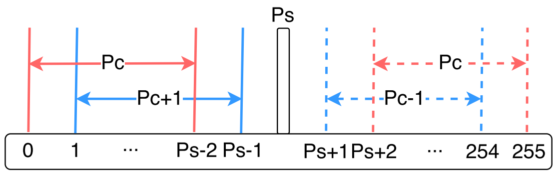
Eq. 11-16 show different range of values that J_{P_c} can take, depending on $B - B'$ and P_s. Graphical representations for each case is shown in Fig. 6:



(a) $B - B' < 0$ or $P_s < 2$, *apply RHS*



(b) $B - B' > 0$ or $P_s > 253$, *apply LHS*



(c) For all other cases, *apply RHS or LHS*

Fig. 6. Different cases for J_{P_c} for selecting P_c and its neighbor $P_c - d$.

- $B - B' < 0$ or $P_s < 2$: Fig. 6a shows that $P_c \in [P_s + 2, 255]$ due to P_s and $P_c - 1$'s locations.
- $B - B' > 0$ or $P_s > 253$: Fig. 6b shows that $P_c \in [0, P_s - 2]$ due to P_s and $P_c + 1$'s locations.
- For all other cases: Fig. 6c shows that $P_c \in [0, P_s - 2] \cup [P_s + 2, 255]$.

3) Multiple candidates: During the adaptive bin selection process, it is possible that multiple candidates of P_s and P_c to exist. For P_s, although it is rare, the smallest candidate is selected as P_s because there is no clear best candidate. For P_c, multiple candidates could be very common for first few repetitions of the histogram shifting, since many histogram bins could be empty. In this case, P_c candidate which is the closest to P_s is selected as P_c (if there are two such values, the smallest one is selected as there is no clear best candidate).

D. Extraction and recovery

The extraction and recovery process are performed one repetition at a time to reverse each repetition of the histogram shifting.

First, the histogram shifting direction d is implicitly determined using Eq. 3.

- If $d = 1$, RHS needs to be reversed.

For payload extraction:

$$b_k = \begin{cases} 0 & \text{if } p_i'' = P_s \\ 1 & \text{if } p_i'' = P_s + 1 \end{cases} \tag{17}$$

For pixel recovery:

$$p_i' = \begin{cases} p_i'' - 1 & \text{if } P_s < p_i'' < P_c \\ p_i'' & \text{else} \end{cases} \tag{18}$$

However, the recovery step do not return the combined bin P_c to their respective original values. Recovery of the P_c and $P_c - 1$ is done using following equation.

$$p_i' = \begin{cases} P_c & \text{if } p_i'' = P_c \text{ and } L_m = 0 \\ P_c - 1 & \text{else if } p_i'' = P_c \text{ and } L_m = 1 \end{cases} \tag{19}$$

- Similarly, if $d = -1$, LHS needs to be reversed.

For payload extraction:

$$b_k = \begin{cases} 0 & \text{if } p_i'' = P_s \\ 1 & \text{if } p_i'' = P_s - 1 \end{cases} \tag{20}$$

For pixel recovery:

$$p_i' = \begin{cases} p_i'' + 1 & \text{if } P_c < p_i'' < P_s \\ p_i'' & \text{else} \end{cases} \tag{21}$$

$$p_i' = \begin{cases} P_c & \text{if } p_i'' = P_c \text{ and } L_m = 0 \\ P_c + 1 & \text{else if } p_i'' = P_c \text{ and } L_m = 1 \end{cases} \tag{22}$$

E. Side information

To perfectly reconstruct the original image, side information $S = L||P_s^{-1}||P_c^{-1}||16$ LSBs is prepended to the payload before each repetition of histogram shifting. The detailed description for each component of S is provided right below.

• ***L*:** During the recovery step, we use the size of L to find how many bits of the payload is L. Due to the characteristic of the histogram shifting which combines pixels valued $P_c - d$ with P_c, the size of L is equal to the number of pixels equal to P_c in the histogram shifted image $(p_i'')_{i=1}^{I}$.

• **P_s^{-1} and P_c^{-1}:** For every repetition of the histogram shifting, the splitting and combining bins from the previous repetition, denoted P_s^{-1} and P_c^{-1} are included in S. During the recovery and extraction, the same P_s^{-1} and P_c^{-1} are extracted from the payload and used for the next repetition of recovery and extraction. This is possible because recovery and extraction are done in the opposite order of histogram shifting. Furthermore, for the very first repetition of histogram shifting, P_s^{-1} and P_c^{-1} are saved as 16 zeros. The 16 zeros are used as a value for the decoder to know that no more repetitions of recovery and extraction are needed. Doing it this way does not accidentally end the decoding early because the selection of $P_s = 0$ and $P_c = 0$ is not possible.

• **16 LSBs:** Only for the last repetition of histogram shifting, P_s and P_c cannot be recorded on the next S because there is no next repetition. Instead, they are embedded using a least significant bit (LSB) replacement method in the first 16 pixels' LSBs, while the existing 16 pixels' LSB values are added to S to be used during the recovery. To prevent further modification to the 16 pixels' LSBs, which contain P_s and P_c values, they are excluded from the very last repetition of the histogram shifting and consequently for the very first repetition of extraction and recovery.

Note that the maximum size of S, denoted as $|S|_{max}$ is $32 + H'_{P_c} + H'_{P_c - d}$ bits:

- L requires $H'_{P_c} + H'_{P_c - d}$ bits.
- P_s^{-1} and P_c^{-1} require 8 bits each.
- 16 LSBs require 16 bits (included only for the last repetition of histogram shifting).

F. Stop condition

The histogram shifting process is repeated until the stop condition is met. The stop condition triggers when it is determined that histogram shifting cannot be applied in a

Data: Original Image $\{p_i\}_{i=1}^{I}$ and payload.
Set $\{p_i'\}_{i=1}^{I} = \{p_i\}_{i=1}^{I}$.
while do
  Find B, B', P_s, and P_c. Construct S.
  if *Stop condition has reached* **then**
    Exit loop.
  else
    Apply histogram shifting. Set $\{p_i'\}_{i=1}^{I} = \{p_i''\}_{i=1}^{I}$.
  end
end
Restore $\{p_i'\}_{i=1}^{I}$, P_s and P_c back to values prior to applying the last repetition of histogram shifting.
Construct last S and apply histogram shifting (excluding the first 16 pixels).
Replace first 16 LSBs with P_s and P_c.
Result: Enhanced Image $\{p_i''\}_{i=1}^{I}$.

Algorithm 1: Encoder.

Data: Enhanced Image $\{p_i''\}_{i=1}^{I}$.
Read P_s and P_c from the 16 pixels' LSB.
Extract last S and payload (exclude first 16 pixels).
Recover p_i' from p_i'' (exclude first 16 pixels).
Recover first 16 pixels' LSBs using S.
Set $\{p_i''\}_{i=1}^{I} = \{p_i'\}_{i=1}^{I}$.
while $P_s \neq 0$ *and* $P_c \neq 0$ **do**
  Extract S and payload.
  Recover p_i' from p_i''.
  Set $\{p_i''\}_{i=1}^{I} = \{p_i'\}_{i=1}^{I}$.
end
Result: Original Image $\{p_i\}_{i=1}^{I}$ and payload.

Algorithm 2: Decoder.

reversible manner, *i.e.,* if the theoretic embedding capacity is smaller than the maximum size of the side information:

$$\sum_{i=17}^{I} ||p_i' = P_s|| < |S|_{max} \tag{23}$$

We refer $\sum_{i=17}^{I} ||p_i' = P_s||$ as the theoretic embedding capacity, because the first 16 pixels are excluded from being counted as potentials for embedding. This is because during the last repetition of the histogram shifting, they are reserved for storing P_s and P_c, and should not be used for histogram shifting.

When the stop condition is met, the reference image, P_s, and P_c are restored to the values before the last histogram shifting. Then, histogram shifting is applied for the one last time, without including the first 16 pixels.

For clarity, the encoder and decoder are shown in Algorithm 1 and 2, respectively.

G. Numerical example

One repetition of histogram shifting example is provided to aid with understanding.

Let original image $(p_i)_{i=1}^{I}$ be:

$$(p_i)_{i=1}^{55} = (p_i')_{i=1}^{55} = (10, 13, 1, 13, 10, 10, 1, 1, ..., 1)$$

The corresponding frequency table is shown in Table 1. Let

TABLE I
H_n' VALUES FOR 1ST REPETITION.

n	0	1	2	3	4	...	10	...	13	...	255
H_n'	0	50	0	0	0	...	3	...	2	...	0

payload be $b = (0, 1, 1, 0, 1)$.

For histogram shifting, we need to find P_s and P_c based on $B - B'$ value.

For selecting P_s: $J_{P_s} = [0, 255]$ (Eq. 10) since $B - B' = 0$. Then, using Eq. 7:

$$P_s = \arg\max_{n \in [0,255]} H_n = 1$$

as highlighted in yellow in Table 1.

Similarly, for selecting P_c: $J_{P_c} = [3, 255]$ since $P_s = 1$ (Eq. 12). Then, using Eq. 15:

$$P_c = \arg\min_{n \in [3,255]} H_n = 3$$

Note that there are multiple candidates for selecting P_c. We choose $n = 3$, since it is the value which is closest from P_s (highlighted in green in Table 1).

Now, we will generate side information $S = L||P_s^{-1}||P_c^{-1}||16$ LSBs. For the location map L, because $d = 1$, $H_{P_{c-d}} = H_{P_2} = 0$ and $H_{P_c} = H_{P_3} = 0$, it means $L = [\ ]$ *i.e.*, it is not needed. For P_s^{-1} and P_c^{-1}, since it is the first repetition of the histogram shifting, $P_s^{-1} = P_c^{-1} = (0, 0, 0, 0, 0, 0, 0, 0)$. Since this is not the last repetition of the histogram shifting, storing 16 LSBs are not required. Thus, side information is:

$$S = (0, 0, 0, 0, 0, 0, 0, 0)||(0, 0, 0, 0, 0, 0, 0, 0) = \{0\}_1^{16}$$

Now that S is defined, we check for the stop condition: $\sum_{i=17}^{55} ||p_i' = P_s|| = 39$ and $|S|_{max} = 32$, therefore $\sum_{i=17}^{55} ||p_i' = P_s|| > |S|_{max}$; since the stop condition hasn't been met, we can apply histogram shifting.

Apply histogram shifting: We first append S in front of b, therefore $b = (\{0\}_1^{16}, 0, 1, 1, 0, 1)$. Finally, applying RHS with $P_s = 1$ and $P_c = 3$, the histogram shifted image is

$$(p_i'')_{i=1}^{55} = (10, 13, 1, 13, 10, 10, \{1\}_1^{16}, 1, \mathbf{2}, \mathbf{2}, 1, \mathbf{2}, 1, ..., 1)$$

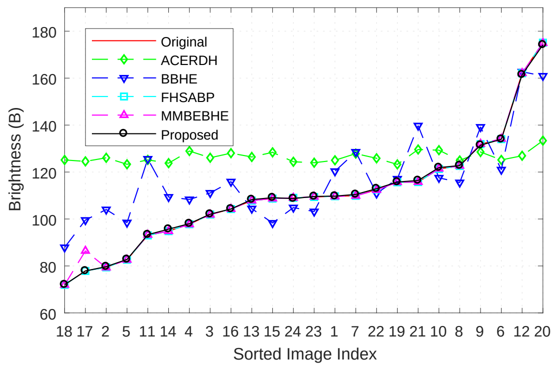
The bold fonts represent the values changed due to histogram shifting.

Now we set the new reference image $(p_i')_{i=1}^{55}$ as the histogram-shifted image $(p_i'')_{i=1}^{55}$ and repeat the process until the stop condition is met.

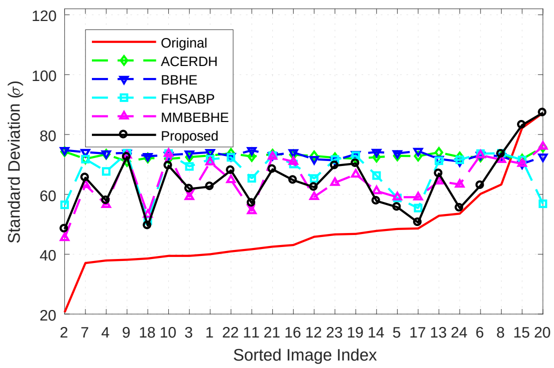
IV. Experimental Result and Analysis

The experimental result and analysis are organized as follows:

1) Section IV-A compares the BP property.
2) Section IV-B analyzes the standard deviation reduction effect.
3) Section IV-C provides the visual comparison.
4) Section IV-D analyzes the effect of the number of histogram shifting repetitions with several key factors.



(a) Brightness



(b) Standard deviation

Fig. 7. Quantitative comparison.

5) Section IV-E analyzes the effect of the histogram shifting in terms of file size gain and embedding capacity.
6) Section IV-F provides an extension for color images.

Four methods are used for comparison: ACERDH [3] is the most recent contribution to the AIE. This is reversible and automatic, and it provides similar contrast enhancement to that of the global histogram equalization. This is the only non-BP method used for the comparison. The second method is the brightness preserving bi-histogram equalization (BBHE) by Kim [12]. The third method is an extension of BBHE, the minimum mean brightness error bi-histogram equalization (MMBEBHE) by Chen et al. [13]. Finally, the fourth method is the flattest histogram specification with accurate brightness preservation (FHSABP) by Wang et al.[4]. For a fair comparison, only automatic methods are compared, *i.e.,* reversible contrast enhancement methods [4, 5, 8] are not compared. In terms of the images used for the comparison, 24 images from the Kodak lossless true color image suite[5] are used. The images are converted to grayscale images, and common contrast enhancement measures (brightness error and standard deviation)[6] are measured.

[4]See Appendix A for detailed description of related works

[5]http://www.r0k.us/graphics/kodak/

[6]See Appendix B for detailed definition for each metric

A. Brightness preservation analysis

The enhanced image's brightness for each method are shown in Fig. 7a, where the results are sorted by the original brightness values from the smallest to the largest.

- FHSABP and the proposed method preserve the brightness well. For FHSABP, because it can enhance the image to have a specific brightness, it preserves the brightness very well. For the proposed method, it achieves BP up to a difference of 1 due to adaptive bin selection.
- BBHE and ACERDH preserves the brightness the least. BBHE preserves the brightness only for 7 out of 24 images, while and ACERDH does not preserve the brightness at all because it is modeled after global histogram equalization; it is always between 120 and 140.

B. Standard deviation reduction analysis

Standard deviation or the change in standard deviation are used to numerically measure the contrast. The positive change in standard deviation indicates that the contrast has been increased, whereas negative change indicates that the contrast has been reduced.

However, a large positive change in contrast is not always desirable because it may indicate removal of fine details at the expense of the increased contrast. On the other hand, a method can also reduce the contrast if it is applied to an image with already high standard deviation. We define this as standard deviation reduction effect.

The standard deviation values are shown in Fig. 7b, where the results are sorted based on the standard deviation of the original images.

- ACERDH and BBHE cause the most change in standard deviation (average values of +27 and +28, respectively).
- BP methods cause less change to standard deviation.
- Only the proposed method does not exhibit standard deviation reduction effect for images 15 and 20, while the remaining methods reduce between -10.4 and -30.3.

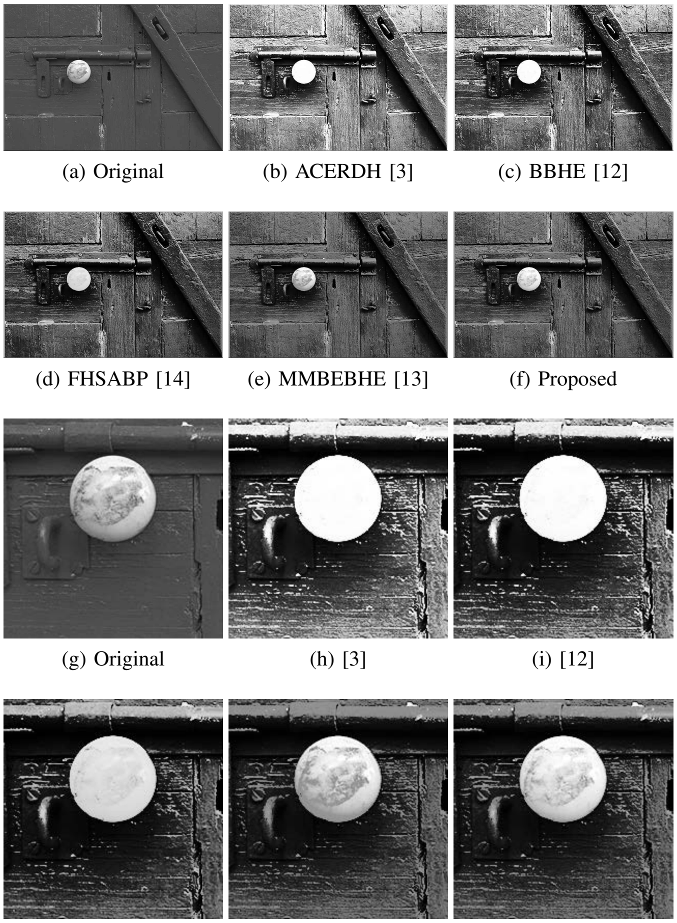
C. Visual analysis

In this section, visual analysis is provided along with the quantitative results to show that the proposed method performs better than existing methods with respect to brightness preservation and standard deviation reduction effect. Due to the length of the paper, only three images and their close-ups are shown here[7].

1) Visual effects of brightness preservation: Images 2 and 12 are selected to examine the visual effect of the brightness preservation and specifically how brightness preserving methods fair against ACERDH and BBHE, the two methods that do not preserve the brightness well.

Enhanced images of the door (Kodak image 2) are shown in Fig. 8. From the close-up views of the doorknob, all methods degrade the details on the doorknob to some extent. ACERDH and BBHE remove most of the details on the doorknob, which is congruent with the numerical change in the brightness

[7]Full sized images can be found at https://github.com/suahnkim/rdhmbp



(a) Original (b) ACERDH [3] (c) BBHE [12]

(d) FHSABP [14] (e) MMBEBHE [13] (f) Proposed

(g) Original (h) [3] (i) [12]

(j) [14] (k) [13] (l) Proposed

Fig. 8. Kodak image 2: Only MMBEBHE [13] and the proposed method retains most of the details on the doorknob.

(a) Original (b) ACERDH [3] (c) BBHE [12]

(d) FHSABP [14] (e) MMBEBHE [13] (f) Proposed

(g) Original (h) [3] (i) [12]

(j) [14] (k) [13] (l) Proposed

Fig. 9. Kodak image 12: ACERDH [3] darkens the image to a level which reduces visibility, while MMBEBHE [13] eliminates the man's facial and hair features.

values. ACERDH and BBHE increase the brightness by +47 and +25 respectively, while the remaining methods cause unnoticeable change (see Fig. 7a for image 2).

Enhanced images of a couple walking on the beach (Kodak image 12) are shown in Fig. 9. From the close-up views, the man's face is much darker for ACERDH. This makes the man's facial features hardly distinguishable, making him appear to have a very dark skin, even though the original image shows him to be a tanned white man. In terms of the brightness value, Fig. 7a shows that ACERDH reduces the brightness by -35.2, while the other methods cause almost no changes. For MMBEBHE, it flattens out the man's facial features to the extent that, his neckline is not very visible, and a patch of his hair is the same color as his face, making him look partially bald. The proposed method and FHSABP darken the man's face to a certain degree, but it is not as severe as with ACERDH, and facial features are not lost. To summarize, only FHSABP and the proposed method produce an acceptable image.

2) Standard deviation reduction effect: Standard deviation reduction effect is present when contrast enhancement method is used on the image with already high contrast. The proposed method is not affected by the standard deviation reduction effect as much as other methods.

Fig. 10b-e, enhanced version of Kodak image 15, show brightened girl's face and darkened paint, which makes it harder to distinguish the two. Moreover, close-up views in Fig. 10h-k show that there is color degradation around the hair, where a large region of the hair color is changed from black to light gray, making it look unnatural. Only the proposed image does not visibly reduce the contrast.

D. Effect of the number of repetitions

The proposed method and ACERDH [3] achieve contrast enhancement by applying histogram shifting in repetitions.

Fig. 11 summarizes the number of repetitions required for each image. The proposed method requires about 40% fewer repetitions than ACERDH, which is result of the proposed adaptive bin selection that impose restrictions to guarantee BP.

Fig. 12 summarizes the relationship between the number of repetitions and other metrics. Following are detailed analysis:

1) Standard deviation vs. number of repetitions: The change in standard deviation results are shown in Fig. 12a. Images that require many repetitions also have the largest gain in standard deviation. While the proposed method achieves

(a) Original (b) ACERDH [3] (c) BBHE [12]

(d) FHSABP [14] (e) MMBEBHE [13] (f) Proposed

(g) Original (h) [3] (i) [12]

(j) [14] (k) [13] (l) Proposed

Fig. 10. Kodak Image 15: Only the proposed method retains the black hair color.

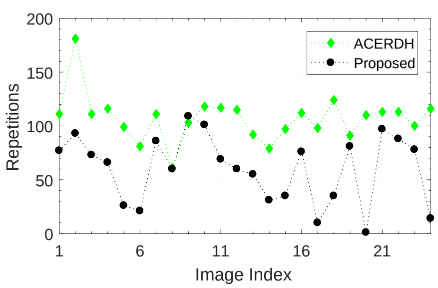
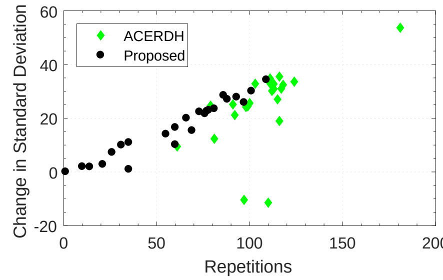
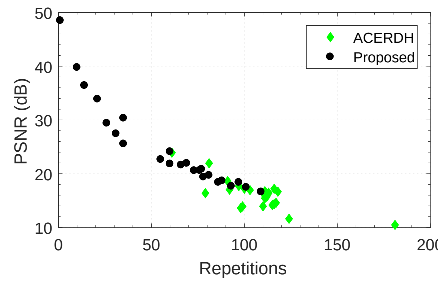


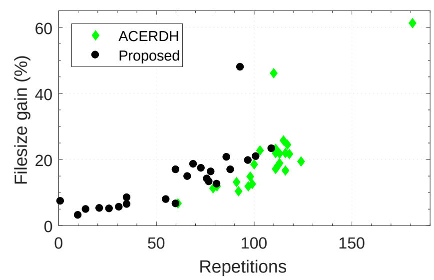
Fig. 11. Number of repetitions.



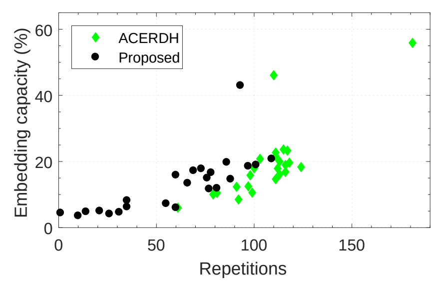
(a) Change in standard deviation



(b) PSNR



(c) File size gain



(d) Embedding capacity

Fig. 12. Comparison between ACERDH and proposed method.

smaller change, it does not have two outliers like ACERDH, where the change is negative. In fact, for image 20, which already has high standard deviation value of over 80, the proposed method only applies 1 repetition of histogram shifting, while ACERDH applies more than 100 repetitions.

2) PSNR vs. number of repetitions: PSNR results are shown in Fig. 12b. They show that the PSNR values are decreasing logarithmically compared to the number of repetitions. This is not surprising, since PSNR represents the log squared sum of the change in pixel values, *i.e.,* higher number of repetitions of histogram shifting leads to more pixels being modified. Naturally, the proposed method has higher PSNR than ACERDH.

3) File size vs. number of repetitions: The file size gain results are summarized in Fig. 12c, where the original and the enhanced images are losslessly compressed using PNG format. The enhanced images generally have larger file size than the original image, and the number of repetitions and the file size gain are positively correlated, indicating that the reversible data hiding tends to increase the file size. This correlation is as expected because histogram equalization process distributes the frequent pixels to its neighbors, which makes it harder to predict and thus harder to compress.

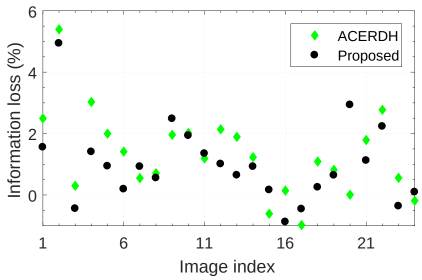


Fig. 13. Comparison of information loss for each image.

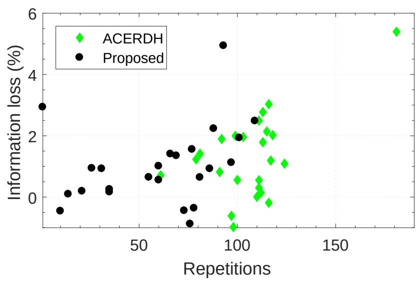


Fig. 14. Information loss vs. repetition.

4) Embedding capacity vs. number of repetitions: The embedding capacity (excluding the space required to store the side information) results are shown in Fig. 12d, where embedding capacity is measured relative to the original image's file size. Images that require a greater number of repetitions also correspondingly have a bigger embedding capacity and the figure suggests strong linear relationship between the embedding capacity and the number of repetitions.

E. Information loss due to embedding

In this subsection, we discuss the relationship between the file size gain due to embedding and the embedding capacity. We define "information loss due to embedding" as a percentage measure of the difference between the file size gain and the embedding capacity relative to the original file size:

$$\text{Information loss} = \frac{\text{File size gain - Embedding capacity}}{\text{Original file size}} \times 100 \tag{24}$$

Fig. 13 shows that the information loss due to embedding is overall positive for both methods, meaning that embedding cause more file size gain than the embedding capacity. In average, the proposed method has a lower information loss due to embedding than ACERDH; 1% for the proposed method and 1.32% for ACERDH. Furthermore, the proposed method produces negative information loss for images 3, 16, 17 and 23, meaning that for those images, embedding capacity is bigger than the file size gain; ACERDH produces negative information loss for images 15, 17 and 24.

Images enhanced with a greater number of repetitions should have a greater information loss, due to the increased distortion. Fig. 14 shows that for many cases, higher number of repetitions of the histogram shifting does seem to lead to greater information loss for most cases. For instance, image 2

(a) Original (b) Maximum repetition (c) Proposed extension

Fig. 15. Color Extension.

enhanced using ACERDH with 181 repetitions results in the largest information loss of 5.4%.

However, for some results with many repetitions also result in a smaller information loss, which means information loss is not only dependent on the number of repetitions, but also on the image content as well. The dependency on the image content can be observed in Fig. 13, where information loss values are close for both methods even though the number of repetitions observed in Fig. 11 vary quite a bit.

F. Extension to color images

The proposed method can be easily extended to the color domain by applying it to R, G, and B channels and combining them to form a one image. However, a careful consideration is required to ensure that the enhancement is applied evenly among all three channels. For example, suppose that the contrast increased along the 'R' channel is considerably less than that of the other two channels. Then, objects that were originally black or white may become greener and bluer making the enhanced image look unnatural.

The proposed extension prevents overenhancement in one or two channels by equalizing the number of repetitions across all channels. First, the maximum number of repetitions possible for each channel is determined. Then, the same number of repetitions of histogram shifting is applied for each channel, where the number of repetitions is the smallest of the maximum number of repetitions among the three channels, preventing overenhancement just in one or two channels. Fig. 15 shows the original Kodak image 2 and its color enhancements. Fig. 15b shows the case where the maximum number of histogram shifting is applied for all three channels, whereas Fig. 15c shows the case of the proposed extension. Although the brightness is preserved for both cases, the proposed extension results in fewer darker spots (from overenhancement of B and G channels) and fewer shades of olive color around the doorknob's shadow, making the image look more natural.

V. Conclusion

This paper presents a novel automatic contrast enhancement method using reversible data hiding. The proposed method shows a framework, where a classical image enhancement primitive called brightness preservation can be incorporated using adaptive bin selection process. The experimental results show that the enhanced images are less susceptible to overenhancement. Furthermore, the proposed method requires smaller storage requirement for automatic image enhancement purpose, since the original image does not need to be backed up.

VI. Acknowledgment

The authors would like to thank all the anonymous reviewers for their constructive feedback to improve the content and the presentation of the paper.

Appendix A
Related work on BP contrast enhancement

BP methods have been studied to reduce the excessive increase in contrast. Brightness preserving bi-histogram equalization (BBHE) by Kim et al. [12] is one of the first such methods. The histogram is first divided into two histograms with respect to the brightness. The two histograms are then independently equalized. Since pixels are distributed equally only within their respective parts, the pixels are modified within a smaller range. Compared to the global histogram equalization, where pixels are modified within the entire pixel range, BBHE causes smaller change to the brightness. Chen et al. [13] proposed an extension of BBHE called minimum mean-brightness bi-histogram equalization (MMBEBHE). Like BBHE, MMBEBHE splits the histogram into two, but the split is based not on the brightness of the original image but based on the change in brightness value. Although, it is possible to try all 255 different histogram bins for the split, in [13], an estimation method was proposed to determine the split that would yield the smallest brightness error. Therefore, the brightness error obtained from MMBEBHE will always be smaller or equal to BBHE. Instead of equalizing only using two histograms, Huang et al. [15] proposed a method that independently equalizes several sub-histograms. Finally, to achieve close to perfect BP, a method called flattest histogram specification with accurate brightness preservation (FHSABP) was proposed by Wang et al. [14]. This method is based on the exact histogram specification method by Coltuc et al. [16]. Instead of distributing the pixels to each bin equally, pixels are distributed to each bin such that the number of pixels in each bin is either linearly increasing or decreasing. The linear factor is used to control the brightness to be almost equal to that of the original image.

Appendix B
Quantitative Evaluation Methods

A. Change in standard deviation

The contrast can be measured using the standard deviation σ:

$$\sigma = \sqrt{\sum_{n=0}^{255} (n-B)^2 \cdot \frac{H_n}{I}} \tag{25}$$

where B is the average pixel value, H_n is number of pixel values equal to n, and I is the total number of pixels.

The change in standard deviation $\Delta\sigma$ is

$$\Delta\sigma = \sigma_e - \sigma_o \tag{26}$$

where σ_o and σ_e represent the standard deviation of the original and the enhanced image, respectively.

B. Similarity to the original image

Similarity between the enhanced and the original image can be measured using the peak-signal-to-noise ratio PSNR.

$$PSNR = 10 \cdot \log_{10}\left(\frac{255^2}{\sum_{i=1}^{I}(p_i - p_i')^2}\right) \tag{27}$$

The ratio is calculated over all I pixels, where p_i is the original pixel value and p_i' is the reference pixel value.

References

[1] B. Delean, "Method and system for non-destructive image editing," U.S. Patent 6 181 836, 2001.

[2] P. McLean, "System for non-destructive image processing," U.S. Patent 8 995 794, 2015.

[3] S. Kim, R. Lussi, X. Qu, and H. J. Kim, "Automatic contrast enhancement using reversible data hiding," in *IEEE International Workshop on Information Forensics and Security*, Nov. 2015, pp. 1–5.

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

[5] 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, Nov. 2015.

[6] H. Chen, W. Hong, J. Ni, and J. Yuan, "A tunable bound of the embedding level for reversible data hiding with contrast enhancement," in *International Conference on Cloud Computing and Security*. Springer, 2016, pp. 134–144.

[7] H. Chen, J. Ni, W. Hong, and T.-S. Chen, "Reversible data hiding with contrast enhancement using adaptive histogram shifting and pixel value ordering," *Signal Processing: Image Communication*, vol. 46, pp. 1–16, 2016.

[8] 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.

[9] Y. Yang, W. Zhang, D. Liang, and N. Yu, "A roi-based high capacity reversible data hiding scheme with contrast enhancement for medical images," *Multimedia Tools and Applications*, pp. 1–23, 2017.

[10] ——, "Reversible data hiding in medical images with enhanced contrast in texture area," *Digital Signal Processing*, vol. 52, pp. 13–24, 2016.

[11] G. Gao, X. Wan, S. Yao, Z. Cui, C. Zhou, and X. Sun, "Reversible data hiding with contrast enhancement and tamper localization for medical images," *Information Sciences*, vol. 385, pp. 250–265, 2017.

[12] Y.-T. Kim, "Contrast enhancement using brightness preserving bi-histogram equalization," *IEEE Transactions on Consumer Electronics*, vol. 43, no. 1, pp. 1–8, 1997.

[13] S.-D. Chen and A. R. Ramli, "Minimum mean brightness error bi-histogram equalization in contrast enhancement," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 4, pp. 1310–1319, 2003.

[14] C. Wang, J. Peng, and Z. Ye, "Flattest histogram specification with accurate brightness preservation," *IET Image Processing*, vol. 2, no. 5, pp. 249–262, 2008.

[15] S.-C. Huang and C.-H. Yeh, "Image contrast enhancement for preserving mean brightness without losing image features," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 5–6, pp. 1487–1492, 2013.

[16] D. Coltuc, P. Bolon, and J. M. Chassery, "Exact histogram specification," *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1143–1152, May 2006.

Suah Kim Suah Kim (SM'15) received the Bachelor of Mathematics from University of Waterloo in 2012, and is working towards the Ph.D degree from Korea University. He was a visiting young scientist at Sun Yat-sen University in 2016. His research interests include reversible data hiding, and image forensics.

Rolf Lussi Rolf Lussi received the Master of Science degree in engineering in information and communication technologies from ZHAW, Winterthur, Switzerland, in 2014. He is currently working as development engineer for Kistler Group, Winterthur, Switzerland. His research interests include reversible watermarking, information security and image processing.

Xiaochao Qu Xiaochao Qu is currently a senior engineer at MTlab (Meitu, Inc). He received the PhD degree from the Department of Information Security and Management at Korea University in 2015. Before that, he revieved the BS degree from the Department of Computer Science and Technology in 2009. He mainly focuses on the research topics in the area of Computer Vision, particularly object detection, semantic segmentation, human pose estimation and deep learning-related problems.

Fangjun Huang Fangjun Huang (M'07) received the B.S. degree from Nanjing University of Science and Technology, Nanjing, China, in 1995, and the M.S. and Ph.D. degrees from Huazhong University of Science and Technology, Wuhan, China, in 2002 and 2005, respectively. He was a Post-Doctoral Researcher with New Jersey Institute of Technology, Newark, NJ, USA, from 2009 to 2010. From 2013 to 2014, he was a Korea Foundation for Advanced Studies Scholar with Korea University, Seoul, Korea. He is currently a Professor with the Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His research interests include reversible data hiding, steganography, steganalysis, and digital forensics.

Hyoung Joong Kim Hyoung Joong Kim (M'09) received the BS, MS and PhD degrees from Seoul National University in 1978, 1986, and 1989, respectively. He was a Professor of Kangwon National University from 1989 to 2006. He is currently a Professor with Korea University. His research interests include distributed computing, machine learning, multimedia security, and cryptocurrency.