An Advanced Reversible Data Hiding Algorithm Using Local Similarity, Curved Surface Characteristics, and Edge Characteristics in Images

Soo-Mok Jung 1,* and Byung-Won On 2

1 Division of Computer Mechatronics Engineering, Sahmyook University, Seoul 01795, Korea
2 Department of Software Convergence Engineering, Kunsan National University, Kunsan 54150, Korea;
   bwon@kunsan.ac.kr
* Correspondence: jungsm@syu.ac.kr

Received: 10 December 2019; Accepted: 20 January 2020; Published: 24 January 2020

Abstract: In this paper, we proposed methods to accurately predict pixel values by effectively using local similarity, curved surface characteristics, and edge characteristics present in an image. Furthermore, to hide more confidential data in a cover image using the prediction image composed of precisely predicted pixel values, we proposed an effective data hiding technique that applied the prediction image to the conventional reversible data hiding technique. Precise prediction of pixel values greatly increases the frequency at the peak point in the histogram of the difference sequence generated using the cover and prediction images. This considerably increases the amount of confidential data that can be hidden in the cover image. The proposed reversible data hiding algorithm (ARDHA) can hide up to 24.5% more confidential data than the existing algorithm. Moreover, it is not possible to determine the presence of hidden confidential data in stego-images, as they possess excellent visual quality. The confidential data can be extracted from the stego-image without loss, and the original cover image can be restored from the stego-image without distortion. Therefore, the proposed algorithm can be effectively used in digital image watermarking, military, and medical applications.

Keywords: cover image; stego-image; reversible data hiding; local similarity; curved surface characteristics; edge characteristics

1. Introduction

Data hiding is an important technique for embedding confidential data in cover media such as digital images. A stego-image is generated by hiding confidential data in a cover image, and the confidential data are extracted from the stego-image. Imperceptibility is vital in data hiding, to make it impossible for humans to know whether confidential data is hidden in stego-images [1]. To meet this requirement, the quality of the stego-image must be excellent. Most data hiding techniques that aim to improve the stego-image’s image quality cause distortion in the recovered cover image after extracting the confidential data from the stego-image. Thus, the recovered cover image and the original cover image are inconsistent [2]. The reversible data hiding technique, in which the recovered cover image matches the original cover image, is important in digital image watermarking, military, medical, and digital library [3] applications. Various reversible data hiding techniques have been recently proposed [2–38]. Various techniques have been proposed to hide data using the least significant bit (LSB) of image pixels [4–9]. The data hiding techniques using the LSB of a pixel has a disadvantage in that the restored cover image does not match the original cover image because the LSBs are changed. Data hiding based on wavelet analysis, especially multi-resolution analysis (MRA), has been widely...
used, and one of the problems with data hiding methods is the visibility of the information contained in the available distributed images [10–15]. Data hiding techniques using an additional reference matrix are proposed to increase embedding efficiency [16–19], and those using pixel value ordering (PVO) are also used [20–23]. Various reversible data hiding techniques have been recently proposed to hide confidential data by shifting the histogram of an image [24–38]. In particular, among these techniques, the techniques of [35–38] are reversible data hiding techniques using a prediction image. In the techniques of [35–38], a prediction image is generated by predicting pixel values using surface characteristics present in a cover image, and the prediction image is used to embed confidential data in the cover image.

Ni et al. proposed the NSAS technique using the histogram of an image [1]. This technique examines the peak and zero points in the histogram of a cover image, shifts the pixels between the points, and inserts confidential data into the pixels corresponding to the peak point. Therefore, the technique has a weakness; the maximum number of bits that can be embedded is limited to the frequency at the peak point in the histogram of the cover image.

Li et al. proposed the adjacent pixel difference (APD) technique [2] that improved the NSAS technique. It is a reversible data hiding technique that embeds confidential data using the histogram of an adjacent pixel difference sequence. In this technique, an adjacent pixel difference sequence composed of the differences between adjacent pixels in the cover image is generated, and then its histogram is generated. As adjacent pixels are generally similar to each other, many elements in the sequence have values close to zero. Therefore, the frequency at the peak point in the histogram of the sequence becomes very large, and the number of data bits that can be hidden in the cover image is greatly increased. This technique has a weakness; the number of data bits that can be embedded in the cover image is limited to the frequency at the peak point in the histogram of the adjacent pixel difference sequence.

To solve this issue and hide a large amount of confidential data, various reversible data hiding techniques have been proposed that effectively use the local similarity and the surface characteristics in images [35–38]. They are based on the APD technique and are effective techniques for hiding a large amount of confidential data in cover image at various levels. The APDLS (adjacent pixel difference using spatial locality and surface characteristics) technique [37,38] can hide more confidential data than the technique of [35,36]. It is a reversible data hiding technique that embeds confidential data using the histogram of a difference sequence generated using the cover image and the prediction image composed of predicted pixel values. The pixel values are precisely predicted in two steps to generate the prediction image. In the first step, locations that are expected to have high local similarity are selected using neighboring pixels. In the second step, the surface characteristics of the image are examined at the locations where local similarity is expected to be high, and pixel values are accurately predicted at the locations by reflecting the surface characteristics. If a difference sequence is generated using the differences between the pixel values of the cover and prediction images, the frequency at the peak point in the histogram of the difference sequence is greatly increased. Therefore, APDLS embeds more data bits in a cover image than the technique of [35,36].

We propose an advanced reversible data hiding algorithm (ARDHA) that performs even better than APDLS. In the ARDHA technique, a prediction image is generated after precisely predicting pixel values according to various curved surfaces and various edge characteristics of an image. Therefore, more pixel value prediction is performed than within the APDLS technique, and the pixel values are accurately predicted. The frequency at the peak point in the histogram of the difference sequence generated using the cover and prediction images is further increased, thereby increasing the number of hidden data bits.

This paper is organized as follows: In Section 2, the existing APD technique is described. In Section 3, we describe the ARDHA technique in detail, and we briefly describe the APDLS technique. In Section 4, the experimental results for evaluating the performance of the ARDHA technique are described and analyzed. The conclusions are described in Section 5, followed by discussion in Section 6.
2. Adjacent Pixel Difference (APD)

The histogram of Lena, a 512 × 512 gray scale image, is shown in Figure 1a. The pixel values with the highest and second-highest frequencies were PP\textsubscript{1} (peak point 1) and PP\textsubscript{2} (peak point 2), respectively. The pixel values with zero frequencies located closest to the left and right of PP\textsubscript{1} and PP\textsubscript{2} are CZP\textsubscript{1} (closest zero point 1) and CZP\textsubscript{2} (closest zero point 2), respectively. Figure 1a shows that CZP\textsubscript{1} = 33, PP\textsubscript{1} = 153, PP\textsubscript{2} = 154, and CZP\textsubscript{2} = 235.

![Histogram of Lena and adjacent pixel difference sequence (D). (a) The histogram of Lena and (b) the histogram of the adjacent pixel difference sequence (D)](image)

Ni, Shi, Ansari, and Su proposed the NSAS technique, a reversible data hiding algorithm using a histogram shift [1]. As mentioned previously, the NSAS technique has a weakness; the number of confidential data bits that can be embedded is limited to 5701 bits, which is the sum of h(PP\textsubscript{1}) and h(PP\textsubscript{2}). The variable h(PP\textsubscript{1}) represents the frequency of the pixel value PP\textsubscript{1}.

Li et al. proposed the adjacent pixel difference (APD) technique [2] that improved the NSAS technique. The APD technique greatly increases the number of confidential data bits that can be embedded in the cover image by effectively using the property that adjacent pixels have similar values. In the APD technique, the cover image is scanned from left to right and top to bottom in inverse s-order to generate the cover image sequence (C), a sequence of pixel values. Equation (1) is applied to C to generate an adjacent pixel difference sequence (D) consisting of pixel value differences between adjacent pixels. As adjacent pixels are generally similar to each other, many elements of D have values close to zero. Therefore, in the histogram of D, the frequency at zero and values close to zero increases significantly. This is due to the existence of local similarity in images.

\[
D_i = \begin{cases} 
C_i & \text{if } i = 0 \\
C_{i-1} - C_i & \text{if } 1 \leq i \leq n-1 
\end{cases} \quad \text{where } n = (\text{image height}) \cdot (\text{image width})
\] (1)

The histogram of D generated by applying Equation (1) to the cover image sequence of Lena is shown in Figure 1b where h(PP\textsubscript{1}) is 25,010 bits, and h(PP\textsubscript{2}) is 23,414 bits. These values were much larger than those in Figure 1a. In the APD technique, as the histogram of D was shifted to embed confidential data, the maximum number of data bits that could be embedded was h(PP\textsubscript{1}) + h(PP\textsubscript{2}) = 48,424 bits, which was the sum of the frequencies at peak points PP\textsubscript{1} and PP\textsubscript{2}. Therefore, the APD technique could embed 42,723 more bits than the NSAS technique, and the number of confidential data bits that could be embedded was 8.5 times more than the ones embedded in NSAS.

The data embedding procedure of the APD technique is shown in Figure 2. The maximum number of data bits that can be embedded in the cover image is h(PP\textsubscript{1}) + h(PP\textsubscript{2}), as shown in Step 5 of Figure 2. The symbol (PP\textsubscript{j}, CZP\textsubscript{j}) used in Equation (2) of Figure 2 represents a set of integers greater than PP\textsubscript{j} and less than CZP\textsubscript{j}. If PP\textsubscript{j} is 1 and CZP\textsubscript{j} is 5, then (1, 5) is {2, 3, 4}. The symbol [PP\textsubscript{j} + sd\textsubscript{j}, CZP\textsubscript{j}] used in
Equation (7) of Figure 3 represents a set of integers greater than or equal to PP\textsubscript{j} + sd\textsubscript{j} and less than or equal to CZP\textsubscript{j}. If PP\textsubscript{j} + sd\textsubscript{j} is 1 and CZP\textsubscript{j} is 5, then \{1, 5\} is \{1, 2, 3, 4, 5\}.

The procedure of extracting the confidential data from the stego-image and restoring the original cover image from the stego-image is shown in Figure 3. By following this procedure, the confidential data can be extracted from the stego-image without loss and the original cover image can be restored without distortion. Therefore, the APD technique is an excellent reversible data hiding technique that can embed large amounts of confidential data.

An example of confidential data embedding in APD is shown in Figure 4, and an example of the confidential data extraction and the original cover image restoration procedure is shown in Figure 5. By constructing the histogram of the adjacent pixel difference sequence (D) of the cover image in Figure 4, it can be seen that PP\textsubscript{1} = 0, PP\textsubscript{2} = 1, h(PP\textsubscript{1}) = 8, and h(PP\textsubscript{2}) = 7. Therefore, the number of data bits that can be embedded in the cover image is h(PP\textsubscript{1}) + h(PP\textsubscript{2}) = 15 bits. As shown in Figure 4, the stego-image generated by embedding confidential data in the cover image was slightly different from the original cover image, and the image quality of the stego-image was slightly degraded. As shown in Figure 5, the confidential data was extracted from the stego-image without loss, and the original cover image was recovered without distortion. Therefore, the APD technique could be used in digital image watermarking, military, and medical applications.

Step 1. Cover image sequence (C) is generated by scanning the pixel values of a cover image from left to right and top to bottom in inverse s-order.

Step 2. Equation (1) is applied to C to generate an adjacent pixel difference sequence (D) having pixel value differences between adjacent pixels.

Step 3. After generating the histogram of D, find PP\textsubscript{1}, PP\textsubscript{2}, CZP\textsubscript{1}, and CZP\textsubscript{2} in the histogram.

Step 4. Shifted adjacent pixel difference sequence (DS) is generated by shifting D according to Equations (2) and (3).

\[ DS_i = \begin{cases} D_i & \text{if } (i = 0 \text{ or } D_i \not\in (PP_j, CZP_j)) \\ D_i + sd_j & \text{if } (D_i \in (PP_j, CZP_j)) \end{cases} \]  

where \( j \in \{1, 2\} \)

\[ sd_j = \begin{cases} 1 & \text{if } (PP_j < CZP_j) \\ -1 & \text{if } (CZP_j < PP_j) \end{cases} \]  

Step 5. Embedded adjacent pixel difference sequence (DE) is generated by embedding confidential data in DS according to Equation (4).

\[ DE_i = \begin{cases} DS_i & \text{if } (i = 0 \text{ or } DS_i \neq PP_j \text{ or data } = 0) \\ DS_i + sd_j & \text{if } (DS_i = PP_j \text{ and data } = 1) \end{cases} \]  

Step 6. Stego-image sequence (S) is generated according to Equation (5).

\[ S_i = \begin{cases} DE_i & \text{if } i = 0 \\ C_{i-1} - DE_i & \text{if } 1 \leq i \leq n - 1 \end{cases} \]  

where \( n = (\text{image height}) \cdot (\text{image width}) \)

Step 7. Stego-image is generated in inverse s-order using S.

\[ Figure 2. \text{The data embedding procedure of APD.} \]
Step 1. Generate the stego-image sequence (S) by scanning the pixel values of the stego-image in inverse s-order.

Step 2. Restore the embedded adjacent pixel difference sequence (DE) and the cover image sequence (C) by repeatedly executing Step 2.1 and Step 2.2 while increasing i.

Step 2.1. Restore the embedded adjacent pixel difference sequence (DE) using Equation (6).

\[
DE_i = \begin{cases} 
C_i & \text{if } i = 0 \\
C_{i-1} - S_i & \text{otherwise}
\end{cases}
\]  

(6)

Step 2.2. Restore the cover image sequence (C) using Equation (7).

\[
C_i = \begin{cases} 
S_i + sd_j & \text{if } (1 \leq i \leq n - 1 \text{ and } C_{i-1} - S_i \in [PP_j + sd_j, CZP_j]) \\
S_i & \text{otherwise}
\end{cases}
\]

(7)

where \( j \in \{1, 2\} \)

Step 3. Extract the confidential data embedded in the stego-image by applying Equation (8) to each element DE\(_i\) of the embedded adjacent pixel difference sequence (DE).

\[
\text{Extracted bit} = \begin{cases} 
0 & \text{if } (DE_i = PP_j) \\
1 & \text{if } (DE_i = PP_j + sd_j)
\end{cases}
\]  

(8)

Step 4. The original cover image is generated in inverse s-order using C.

**Figure 3.** Data extraction and original cover image restoration procedure of APD.

**Figure 4.** An example of data embedding using the APD technique.

**Figure 5.** An example of data extraction and original cover image restoration using the APD technique.
3. Proposed Algorithm

In general, local similarity, curved surface characteristics, and edge characteristics exist in images. That is, adjacent pixel values of an image are very similar, and various curved surface characteristics such as flat surface, simple inclined surface, simple convex surface, and simple concave surface exist in particular locations of the image. Various edges also exist in specific portions of the image. In the proposed algorithm (ARDHA), the locations where local similarity is expected to be high were selected by using neighboring pixels. Next, it was examined whether a flat surface or a simple inclined surface or a simple convex surface or a simple concave surface exist at each location. At each location where local similarity was not expected to be high, it was examined whether a strong edge or a normal edge or a weak edge exist. Thereafter, the pixel values at each location were accurately predicted according to the investigated characteristics.

A prediction image was generated using the precisely predicted pixel values, and a difference sequence was generated using the cover and prediction images. If the prediction image was generated very close to the cover image, the frequencies at the highest and second-highest peak points in the histogram of the difference sequence increased. Thus, the number of confidential data bits that could be embedded in the cover image could be increased. The more accurate the pixel value prediction, the higher the frequencies at the highest and second-highest peak points in the histogram of the difference sequence, and the greater the number of confidential data bits that could be embedded in the cover image.

The flowchart for obtaining the predicted pixel values to generate the prediction image is shown in Figure 6. Figure 7a shows a sample cover image, and Figure 7b shows the prediction order of the pixel values in a predictable region (the region inside double line marked as Region1. 1 to 16).

![Figure 6. Flowchart of pixel value prediction to generate the prediction image.](image-url)
Pixels values are unpredictable in the region outside the double line (two rows on the top and two columns from the left and right sides) as shown in Figure 7b. Figure 7c,d show the neighboring pixels used to predict pixel values at locations (1), (5), (6), and (10) using a flat surface, simple inclined surface, simple convex surface, and simple concave surface characteristics. Figure 7e shows the neighboring pixels used to predict pixel values at locations (6) and (10) using strong edge, normal edge, and weak edge characteristics. As shown in Figure 7e, at (10), 21 pixels in three rows above the location and 3 pixels in the three columns to the right were used to predict the pixel value using the edge characteristics. At (10), 21 pixels in three rows above and 3 pixels in the three columns to the left were used to predict the pixel value using the edge characteristics. Therefore, the region where pixel values were predictable using edge characteristics was the remaining portion except three rows on the top and three columns on the left and right sides (the region inside the bold black square marked as Region1(E), (6), (7), (10), (11), (14), and (15)). The neighboring pixels used for prediction were the pixels of the cover image that had already been scanned in inverse s-order.

The prediction images generated by applying the APDLS and the proposed ARDHA techniques to the sample image are shown in Figure 7f,g, respectively. As shown in Figure 6, to predict the pixel value, it was necessary to determine whether each location in the inverse s-order was within the region where pixel values were predictable. If the current location is not within Region1, the pixel value of the cover image becomes the predicted pixel value. If the current location is within Region1 where pixel values are predictable, the predicted pixel value is calculated according to the procedure of the proposed technique. As shown in Figure 6, in case of Region1 where pixel values were predictable, the deviation 1 (dev1) obtained using $V_1$ calculated based on the local similarity was compared with the threshold ($\beta$). If dev1 is below $\beta$, the surface characteristic at the current location is examined. If any one characteristic of flat, simple inclined, simple convex, and simple concave surfaces exists, the
predicted value ($V_2$ or $V_{3V}$ or $V_{3C}$) is calculated according to the procedure specified to reflect the corresponding surface characteristic.

However, if the current location does not have any of the characteristics of the aforementioned surfaces, $V_1$ is used as the predicted value. The rounded value of the predicted value ($V_1$ or $V_2$ or $V_{3V}$ or $V_{3C}$) is used as the predicted pixel value.

If $dev_1$ is greater than or equal to $\beta$, and at the same time, the current location is within Region1(E) and edge characteristic exists at the current location, the predicted value ($V_{4S}$ or $V_{4N}$ or $V_{4W}$) is calculated according to the procedure specified to reflect the corresponding edge characteristic. The rounded value of the predicted value ($V_{4S}$ or $V_{4N}$ or $V_{4W}$) is used as the predicted pixel value.

The prediction image was generated using the predicted pixel values. The detailed procedure for predicting pixel values was as follows:

Step 1.1: If pixel values are predictable at the current location, Step 1.2 is executed. Otherwise, the pixel value of the cover image becomes the predicted pixel value.

Step 1.2: Based on local similarity, the predicted value $V_1$, which is calculated using the neighboring pixels and their locational weights, is obtained. At location $\mathbb{D}$, $V_1$ is calculated as shown in Equation (9). The coordinates of location $\mathbb{D}$ are $(x_i, y_j)$. The coordinates of pixels with pixel values $M$, $N$, $O$, and $R$ are $(x_{i+1}, y_{j-1})$, $(x_{i}, y_{j-1})$, $(x_{i-1}, y_{j-1})$, and $(x_{i-1}, y_{j})$, respectively. In Equation (9), $P(x, y)$ represents the pixel value at the coordinates $(x, y)$. As shown in Figure 7a, $M$, $N$, $O$, and $R$ are 155, 155, 158, and 160, respectively. As shown in Equation (9), the influence weight of adjacent pixels $(M, N, O, \text{and } R)$ on $V_1$ is set to 1. The influence weight of the remaining pixels $(A, B, C, D, E, L, P, \text{and } Q)$ on $V_1$ is represented by $\alpha$.

$$V_1(x_i, y_j) = \frac{P(x_{i+1}, y_{j-1}) + P(x_{i}, y_{j-1}) + P(x_{i-1}, y_{j-1}) + P(x_{i-1}, y_{j}) + P(x_{i-2}, y_{j-2})}{4 + 8\cdot\alpha}$$

Step 1.3: To determine the degree of local similarity, the deviation $1$ ($dev_1$) between $V_1$ and the neighboring pixel values is calculated as shown in Equation (10).

$$dev_1(x_i, y_j) = \frac{abs(V_1 - P(x_{i+1}, y_{j-1})) + abs(V_1 - P(x_{i}, y_{j-1})) + abs(V_1 - P(x_{i-1}, y_{j-1}))}{4 + 8\cdot\alpha}$$

Step 1.4: The obtained $dev_1$ is compared with the threshold ($\beta$).

Step 1.5: If $dev_1$ is less than $\beta$, it is determined that the image at the location has a small change in pixel values and a very high local similarity. In this case, Step 1.6 to 1.9 are executed to calculate the predicted value using flat surface, simple inclined surface, simple convex surface, and simple concave surface characteristics. If $dev_1$ is equal to or greater than $\beta$, the image at the location is determined as a boundary area with a large change in pixel values. In this case, Step 1.A to 1.E are executed to calculate the predicted value using edge characteristics.

Step 1.6: The surface characteristic of the image is estimated at that location. The process of estimating the surface characteristic of the image at location $\mathbb{D}$ is as follows: The equation of a straight line passing through any two points $(x_1, y_1)$ and $(x_2, y_2)$ is given by Equation (11). Using Equation (11), the linear equation between adjacent pixels may be determined. Between the pixel having the O value and the adjacent pixel having the N value, a linear equation can be obtained that takes into account only the X-axis direction for the pixel value. That is, the equation $E_2(x)$ of a straight line passing through $(x_{i-1}, O)$ and $(x_i, N)$ is expressed by Equation (12). In the same way, taking into account only the X-axis direction for the pixel values between N and
Step 1.8: If the values of the neighboring pixels satisfy Equations (18) to (21), it is assumed that location $O$ is not a simple inclined surface or flat surface, Step 1.7 is executed to check the presence of simple concave surface characteristic at location $O$. If all gradients of Equations (12) to (14) are greater than zero, the surface characteristic is estimated as a simple inclined surface where the pixel value on the right is greater than that on the left. If all gradients of Equations (12) to (14) are less than zero, the surface characteristic is estimated as a simple inclined surface whose left pixel value is greater than the right pixel value. If all gradients of Equations (12) to (14) are zero, the surface characteristic is estimated as a flat surface. If the surface characteristic in the X-axis direction at location $O$ is not a simple inclined surface or flat surface, Step 1.7 is executed to check the presence of simple convex surface characteristic. The predicted value $V_{2}(x, y)$ has a gently changing curved surface characteristic. If $\text{dev}_2(x, y)$ is used as the predicted pixel value at location $O$, the rounded value of $V_{2}(x, y)$ is used as the predicted pixel value at location $O$. If location $O$ is not a simple inclined surface or flat surface, Step 1.7 is executed to check the presence of curved surface characteristics at location $O$.

$$\begin{align*}
    y &= \left\{ (y_2 - y_1) / (x_2 - x_1) \right\} (x - x_1) + y_1 \\
    E_L(x) &= \left\{ (N - O) / (x_i - x_{i-1}) \right\} (x - x_{i-1}) + O \\
    E_R(x) &= \left\{ (M - N) / (x_{i+1} - x_i) \right\} (x - x_i) + N \\
    E_{LD}(x) &= \left\{ (N - R) / (x_i - x_{i-1}) \right\} (x - x_{i-1}) + R \\
    V_2(x, y) &= (M + N + O + R) / 4
\end{align*}$$

Step 1.7: The $\text{avg}(x, y)$, which is the average of the neighboring 11 pixels, is calculated using Equation (16). The $\text{dev}_2(x, y)$, which is the deviation between $\text{avg}(x, y)$ and the neighboring pixels, is calculated using Equation (17). If $\text{dev}_2(x, y)$ is equal to or less than the reference value ($\epsilon$), Step 1.8 is executed because location $O$ has a gently changing curved surface characteristic. If $\text{dev}_2(x, y)$ is larger than $\epsilon$, the rounded value of $V_1(x, y)$ is used as the predicted pixel value.

$$\begin{align*}
    \text{avg}(x, y) &= \frac{\left\{ P(x_{i-2}, y_{j-2}) + P(x_{i-1}, y_{j-2}) + P(x_i, y_{j-2}) + P(x_{i+1}, y_{j-2}) + P(x_{i+2}, y_{j-2}) \\
        + P(x_{i+1}, y_{j-1}) + P(x_i, y_{j-1}) + P(x_{i-1}, y_{j-1}) + P(x_{i-2}, y_{j-1}) + P(x_{i-2}, y_j) \right\}}{11} \\
    \text{dev}_2(x, y) &= \left[ |\text{avg} - P(x_{i-2}, y_{j-2})| + |\text{avg} - P(x_i, y_j)| + |\text{avg} - P(x_i-1, y_{j-1})| + |\text{avg} - P(x_{i+2}, y_{j-2})| + |\text{avg} - P(x_{i+1}, y_{j-1})| + |\text{avg} - P(x_{i+1}, y_{j-1})| + |\text{avg} - P(x_{i+2}, y_{j-1})| + |\text{avg} - P(x_{i+2}, y_{j-1})| + |\text{avg} - P(x_{i-1}, y_j)| \right]
\end{align*}$$

Step 1.8: If the values of the neighboring pixels satisfy Equations (18) to (21), it is assumed that location $O$ has simple convex surface characteristic. The predicted value $V_{3V}(x, y)$ at location $O$ is calculated as shown in Equation (26) using Equations (22) to (25). Taking into account only the X-axis direction for the pixel values between $Q$ and $R$, $A$ and $O$, and $E$ and $M$, the linear equations $\text{con}0[x]$, $\text{con}1[x]$, and $\text{con}3[x]$ are represented by Equations (22), (23), and (25), respectively. Similarly, taking into account only the Y-axis direction for the pixel values between $C$ and $N$, the linear equation is represented by Equation (24). The rounded value of $V_{3V}(x, y)$ is used as the predicted pixel value. If all Equations (18) to (21) are not satisfied, Step 1.9 is executed to check the presence of simple concave surface characteristic at location $O$.

$$P(x_{i-2}, y_j) < P(x_{i-1}, y_j)$$
The values of edge(0) to (3) are calculated using Equations (22) to (25), where pixel values can be predicted using Equations (26) to (30).

\[ P(x_{i-2}, y_{j-2}) < P(x_{i-1}, y_{j-1}) \] (19)
\[ P(x_{i}, y_{j-2}) < P(x_{i}, y_{j-1}) \] (20)
\[ P(x_{i+1}, y_{j-2}) < P(x_{i+1}, y_{j-1}) \] (21)

\[ \text{con0}[x] = \{(R - Q)/(x_{i-1} - x_{i-2})\}(x - x_{i-2}) + Q \] (22)
\[ \text{con1}[x] = \{(O - A)/(x_{i-1} - x_{i-2})\}(x - x_{i-2}) + A \] (23)
\[ \text{con2}[y] = \{(N - C)/(y_{j-1} - y_{j-2})\}(y - y_{j-2}) + C \] (24)
\[ \text{con3}[x] = \{(M - E)/(x_{i+1} - x_{i+2})\}(x - x_{i+2}) + E \] (25)

\[ V_{3\text{C}}(x_{i}, y_{j}) = \left[ \text{K-avg}(x_{i}, y_{j}) + \text{con0}[x] + \text{con1}[x] + \text{con2}[y] + \text{con3}[x] \right]/[K + 4] \text{ where } x \in \{v, c\} \] (26)

Step 1.9: If the values of the neighboring pixels satisfy Equations (27) to (30), it is assumed that location \( \odot \) has simple concave surface characteristic. The predicted value \( V_{3\text{C}}(x_{i}, y_{j}) \) at location \( \odot \) is calculated as shown in Equation (26) using Equations (22) to (25). The rounded value of \( V_{3\text{C}}(x_{i}, y_{j}) \) is used as the predicted pixel value at location \( \odot \). If all Equations (27) to (30) are not satisfied, location \( \odot \) is neither a simple convex surface nor a simple concave surface. Thus, the rounded value of \( V_{1}(x_{i}, y_{j}) \) is used as the predicted pixel value at location \( \odot \).

\[ P(x_{i-2}, y_{j}) > P(x_{i-1}, y_{j}) \] (27)
\[ P(x_{i-2}, y_{j-2}) > P(x_{i-1}, y_{j-1}) \] (28)
\[ P(x_{i}, y_{j-2}) > P(x_{i}, y_{j-1}) \] (29)
\[ P(x_{i+2}, y_{j-2}) > P(x_{i+1}, y_{j-1}) \] (30)

Step 1.A: As the first step of predicting pixel value using edge characteristics, it is determined whether pixel value is predictable at the current location using edge characteristics. If the current location is within Region1(E) (6, 7, 10, 11, 14, and 15), where pixel values can be predicted using the edge characteristics, Step 1.B is executed. If the current location is not within Region1(E), the pixel value of the cover image becomes the predicted pixel value.

Step 1.B: In this step, four edge values used to predict pixel values are determined. The four edges, which are used to predict pixel value at location (6), are shown in Figure 8. To determine the values of the edge(0) to (3), diff[0] to [3] are first calculated according to Equations (31) to (34). Diff[0] to [3] are the sum of the absolute values of the differences between adjacent pixels at 0\(^\circ\), 45\(^\circ\), 90\(^\circ\), and 135\(^\circ\) based on the scan direction at the current location. Equations (31) to (34) are expressed based on the coordinates \((x_{i}, y_{j})\) of location (6). The values of edge(0) to (3) are calculated using Equation (35). As shown in Equation (35), if the diff value is 0, the corresponding edge is strong and has the value strong_edge. If the diff value is 1, the corresponding edge is normal and has the value normal_edge. If the diff value is 2, the corresponding edge is weak and has the value weak_edge. If the diff value is 3, the corresponding edge is very weak and has the value very_weak_edge. If the diff value is greater than or equal to 4, the corresponding edge has zero value.

\[ \text{diff}[0] = |P(x_{i+1}, y_{j}) - P(x_{i+2}, y_{j})| + |P(x_{i+2}, y_{j}) - P(x_{i+3}, y_{j})| \] (31)
\[ \text{diff}[1] = |P(x_{i+1}, y_{j-1}) - P(x_{i+2}, y_{j-2})| + |P(x_{i+2}, y_{j-2}) - P(x_{i+3}, y_{j-3})| \] (32)
\[ \text{diff}[2] = |P(x_{i}, y_{j-1}) - P(x_{i}, y_{j-2})| + |P(x_{i}, y_{j-2}) - P(x_{i}, y_{j-3})| \] (33)
Step 1.C: At location (x, y), if edge(0), or edge(1), or edge(2), or edge(3) has the value strong_edge, the predicted value $V_{4S}(x, y)$ is calculated using Equation (36). Thereafter, as shown in Equation (37), if the absolute value of the difference between $V_2(x, y)$ and $V_{4S}(x, y)$ is equal to or less than the difference threshold ($\delta$), the rounded value of $V_{4S}(x, y)$ is used as the predicted pixel value. In Equation (37), the value of $V_2(x, y)$ is $(T + U + V + W)/4$. When the absolute value of the difference between $V_2(x, y)$ and $V_{4S}(x, y)$ exceeds $\delta$, the pixel value of the cover image becomes the predicted pixel value. In Equation (37), $V(x, y)$ represents the predicted pixel value at location $(x, y)$ and $C(x, y)$ represents the pixel value of the cover image at location $(x, y)$.

$$
V_{4S}(x, y) = \left\{ \begin{array}{ll}
R_S \cdot V_2(x, y) + P(x_{i+1}, y_j) \cdot \text{edge}(0) + P(x_{i+1}, y_{j-1}) \cdot \text{edge}(1) \\
+ P(x_i, y_{j-1}) \cdot \text{edge}(2) + P(x_{i-1}, y_{j-1}) \cdot \text{edge}(3) / |R_S + \text{edge}(0) \\
+ \text{edge}(1) + \text{edge}(2) + \text{edge}(3) \right\} 
\frac{|V_2(x, y) - V_{4S}(x, y)|}{\delta} \leq \delta \right\} 
C(x, y) \quad \text{otherwise}
\right. 
$$

where $x \in \{S, N, W\}$

Step 1.D: If strong_edge is absent in edge(0), edge(1), edge(2), and edge(3), and at least one of them has normal_edge, then the predicted value $V_{4N}(x, y)$ is calculated using Equation (38). The predicted pixel value $V(x, y)$ is obtained in the same manner as shown in Step 1.C by applying $V_{4N}(x, y)$ to Equation (37).

$$
V_{4N}(x, y) = \left\{ \begin{array}{ll}
R_N \cdot V_2(x, y) + P(x_{i+1}, y_j) \cdot \text{edge}(0) + P(x_{i+1}, y_{j-1}) \cdot \text{edge}(1) \\
+ P(x_i, y_{j-1}) \cdot \text{edge}(2) + P(x_{i-1}, y_{j-1}) \cdot \text{edge}(3) / |R_N + \text{edge}(0) \\
+ \text{edge}(1) + \text{edge}(2) + \text{edge}(3) \right\} 
\frac{|V_2(x, y) - V_{4N}(x, y)|}{\delta} \leq \delta \right\} 
C(x, y) \quad \text{otherwise}
\right. 
$$

Step 1.E: If there is no strong_edge nor normal_edge in edge(0) to (3), and at least one of them has weak_edge, the predicted value $V_{4W}(x, y)$ is calculated using Equation (39). The predicted pixel value $V(x, y)$ is obtained in the same manner as shown in Step 1.C by applying $V_{4W}(x, y)$ to Equation (37). If edge(0) to (3) are not strong edges, normal edges, and weak edges, the pixel value of the cover image becomes the predicted pixel value.

$$
V_{4W}(x, y) = \left\{ \begin{array}{ll}
R_W \cdot V_2(x, y) + P(x_{i+1}, y_j) \cdot \text{edge}(0) + P(x_{i+1}, y_{j-1}) \cdot \text{edge}(1) \\
+ P(x_i, y_{j-1}) \cdot \text{edge}(2) + P(x_{i-1}, y_{j-1}) \cdot \text{edge}(3) / |R_W + \text{edge}(0) \\
+ \text{edge}(1) + \text{edge}(2) + \text{edge}(3) \right\} 
\frac{|V_2(x, y) - V_{4W}(x, y)|}{\delta} \leq \delta \right\} 
C(x, y) \quad \text{otherwise}
\right. 
$$
Step 1. The cover image is scanned in inverse s-order. It is determined whether the pixel value is predictable at each location, and a prediction image is generated in one of two ways: If the current location is within Region1 where pixel values are predictable, Step 1.1 to 1.E are executed to generate a prediction image. If the current location is not within Region1, the pixel value is unpredictable at the current location. In this case, the prediction image is generated using the pixel value of the cover image.

Step 2. The cover and prediction images are scanned in inverse s-order to generate the cover image sequence (C) and the prediction image sequence (P) composed of the respective pixel values.

Step 3. Using C and P, the difference sequence (D) is generated according to Equation (40).

\[
D_i = \begin{cases} 
C_i & \text{if } (i = 0) \\
C_{i-1} - C_i & \text{Mode 1} \\
P_i - C_i & \text{Mode 2} \\
P_{i-1} - C_i & \text{Mode 3}
\end{cases}
\]  

(40)

Mode 1: When the predicted pixel value was not used at the current and previous locations of the prediction image.

Mode 2: When the predicted pixel value was used at the current location of the prediction image.

Mode 3: When the predicted pixel value was not used at the current location of the prediction image; however, the predicted pixel value was used at the previous location of the prediction image.

Step 4. After generating the histogram of D, the values PP1, CZP1, PP2, and CZP2 are determined.

Step 5. By shifting D according to Equations (2) and (3), the shifted difference sequence (DS) is generated.

Step 6. The embedded difference sequence (DE) is generated by embedding confidential data in DS according to Equation (4).

Step 7. Using C and DE, the stego-image sequence (S) is generated according to Equation (5).

Step 8. The stego-image is generated in inverse s-order using S.

Figure 9. Confidential data embedding procedure of ARDHA.
Step 1. The stego-image sequence (S) is generated by scanning the pixel values of the stego-image in inverse s-order.

Step 2. C0=S0, DE0=C0.

While increasing the value of i in S from 1 to n-1, the extraction mode at the i-th location is determined, and DE and C are restored according to the extraction mode. By executing step 1.1 to 1.5, it is determined whether the predicted pixel value (one of \(V_i\), \(V_{33}\), \(V_{SC}\), \(V_{SE}\), \(V_{EN}\), and \(V_{EN}\)) was used at the i-th location during the embedding process.

The extraction mode is divided into three modes depending on whether the predicted pixel value was used.

Mode 1: When the predicted pixel value was not used at the current and previous locations of the prediction image during the embedding process. The procedure for extracting the confidential data and restoring the original cover image is the same as APD.

Mode 2: When the predicted pixel value was used at the current location of the prediction image during the embedding process. Embedded difference sequence (DE) is restored using Equation (6). Cover image sequence (C) is restored using Equations (41) and (42). RP\(_i\) is the predicted pixel value calculated at the i-th location when the extraction mode is Mode 2. The predicted pixel value RP\(_i\) at the i-th location is calculated using the restored cover image pixel values obtained so far in inverse s-order. The associated value AV\(_i\) used in Equation (41) is obtained from Equation (42) during the restoration process when the extraction mode is Mode 2 or 3.

Mode 3: When the predicted pixel value was used at the previous location, not the current location. DE is restored using Equation (6), and C is restored using Equation (43). AV\(_i\) in Equation (43) is the value calculated using Equation (42).

\[
C_i = RP_i + AV_i \quad (41)
\]

\[
AV_i = \begin{cases} 
C_{i-1} - S_i - sd_i & \text{if } C_{i-1} - S_i \in \left[PP_i + sd_i, CZP_i\right] \\
C_{i-1} - S_i & \text{otherwise} 
\end{cases} \quad (42)
\]

\[
C_i = RP_{i-1} - AV_i \quad (43)
\]

Step 3. The confidential data is extracted by applying Equation (8) to each element of DE.

Step 4. The original cover image is restored in inverse s-order using C.

**Figure 10.** Confidential data extraction procedure and the original cover image restoration procedure of ARDHA.

An example of embedding confidential data using the ARDHA technique is shown in Figure 11. An example of extracting the confidential data and restoring the original cover image using the ARDHA technique is shown in Figure 12. The values of the variables used in the examples shown in Figures 11 and 12 are as follows: The alpha (\(\alpha\)) used in Equations (9) to (10) was 0.45, the threshold (\(\beta\)) used in Step 1.4 to 1.5 was 10, the reference value (\(\epsilon\)) used in Step 1.7 was 10, and the K used in Equation (26) was 4.0. The strong_edge, normal_edge, weak_edge, and very_weak_edge used in Equation (35) were 1.00, 0.75, 0.50, and 0.25, respectively. The difference threshold (\(\delta\)) used in Equation (37) was 1.6. The RS, RN, and RW used in Equations (36), (38), and (39) were 1.1, 0.9, and 0.7, respectively. The reasons for assigning specific values to the variables were explained in the next section.

The predicted values \(V_j\) obtained from Equation (9) at locations (1) to (16) in Figure 7b were 157.00, 156.30, 155.99, 156.40, 158.11, 156.87, 156.62, 157.29, 157.25, 156.26, 156.88, 158.37, 160.84, 158.75, 156.93, and 155.84. If \(V_j\) is applied to Equation (10), dev1 can be calculated at each location. The dev1 calculated at each location (1) to (16) were 13.40, 9.54, 6.62, 10.19, 19.89, 11.30, 9.96, 9.66, 15.30, 17.32, 12.21, 17.23, 25.15, 21.60, 20.12, and 18.99, correspondingly. If the surface characteristics of the
image are examined at locations ②, ③, ⑦, and ⑧, where dev1 is less than β, the locations with simple inclined surface characteristic or flat surface characteristic in Step 1.5 to 1.6 are ⑦ and ⑧. Therefore, if the predicted V2 values are obtained at locations ⑦ and ⑧ using Equation (15), they are going to be 158.00 and 156.50, respectively. The values 158 and 157, which are rounded values of V2, were used as the predicted pixel values at locations ⑦ and ⑧. In Figure 7g, the rounded value of V2 was indicated by an underlined italic number.

Although dev1 was less than β, the locations without a simple inclined surface or flat surface characteristics were ② and ③. The dev2 values obtained at locations ② and ③ using Equation (17) were 14.55 and 10.00, respectively. Therefore, as shown in Step 1.7, the rounded value of V1 was used as the predicted pixel value at location ② where dev2 was larger than ε. Therefore, the predicted pixel value at location ② was 156. In Figure 7g, the rounded value of V1 was indicated by an italic number. At location ③, dev1 was less than β, dev2 was less than or equal to ε, and Equations (27) to (30) were satisfied. Therefore, as shown in Step 1.9, location ③ was determined to have simple concave surface characteristic. Therefore, the predicted value V3C calculated according to Equations (22) to (26) was 155.38. The value 155, which is rounded value of V3C, was used as the predicted pixel value at location ③. In Figure 7g, the rounded value of V3C was indicated by an underlined bold italic number.

At locations ①, ④ to ⑥, and ⑨ to ⑪, where dev1 was greater than or equal to β, the predicted pixel value was obtained as follows: As shown in Step 1.1A, locations ①, ④, ⑤, ⑨, ⑫, ⑬, and ⑮ were not within Region1(E). Therefore, the pixel value of the cover image became the predicted pixel value.
Locations 6, 10, 11, 14, and 15 were within Region1(E). Thus, pixel values at these locations could be predicted using edge characteristics. At location 6, the values from edge(0) to (3) were 6, 5, 2, and 3. The values from edge(0) to (3) were 0, 0, 0.5 (weak_edge), and 0.25 (very_weak_edge), correspondingly. Therefore, the predicted value V_{4W} obtained from Step 1.E was 156.2. The rounded value of V_{4W} was used as the predicted pixel value at location 6.

In the same way, the predicted values V_{4N}, V_{4S}, and V_{4W} calculated at locations 10, 11, 14, and 15 were 156.1, 157.3, 160.0, and 156.5, respectively. The rounded values of V_{4N}, V_{4N}, V_{4S}, and V_{4W} were used as the predicted pixel values at locations 10, 11, 14, and 15, correspondingly. In Figure 7g, the bold number represents the rounded values of V_{4S}, the underlined bold number represents the rounded values of V_{4N}, and the double underlined bold number represents the rounded values of V_{4W}. If we constructed the histogram of the difference sequence (D) in Figure 11, we could see that PP_1 = 0, h(PP_1) = 16, PP_2 = 1, and h(PP_2) = 7. Therefore, the maximum number of data bits that could be embedded in the cover image was 23 bits; 8 bits more than the maximum obtained in APD.

The technique of [35,36] uses only the predicted value V_1, while APDLS [37,38] uses V_2 as well. The techniques [35–38] were proposed by our research team. The flowchart of pixel value prediction for prediction image generation using the APDLS technique was constructed as follows: In Figure 6, the parts related to V_{3V}, V_{3C}, V_{4S}, V_{4N}, and V_{4W} should be removed. If dev1 is greater than or equal to \( \beta \), the pixel value of the cover image becomes the predicted pixel value. If dev1 is less than \( \beta \), it is determined whether it is a simple inclined surface or flat surface. If it is a simple inclined surface or flat surface in the X-axis direction, the rounded value of V_2 is used as the predicted pixel value. Otherwise, the rounded value of V_1 is used as the predicted pixel value. By changing Figure 6 as described above, the flow chart of pixel value prediction using APDLS can be obtained.

The prediction image generated by applying the APDLS technique to the sample cover image in Figure 7a is shown in Figure 7f. As shown in Figure 7f, the APDLS technique used only the predicted values V_1 and V_2. The predicted pixel values at locations 2 and 3 were 156 and 156, which were the rounded values of V_1. Since locations 7 and 8 had simple inclined surface characteristics, the predicted pixel values at locations 7 and 8 were 157 and 158, respectively, which were the rounded values of V_2. Therefore, when the APDLS technique was applied, the prediction image was generated as shown in Figure 7f. After constructing the difference sequence using the cover image and the prediction image, and generating the histogram of the difference sequence, we could see that PP_1 = 0, h(PP_1) = 11, PP_2 = 1, and h(PP_2) = 7. Therefore, the maximum number of data bits that could be embedded in the cover image using the APDLS technique was h(PP_1) + h(PP_2) = 18 bits.

In the APDLS technique, if \( \beta \) is set to 15, and a prediction image is generated for the sample image as shown in Figure 7a, 20 bits can be embedded in the cover image. In the ARDHA technique, if \( \beta \) is set to 15, and a prediction image is generated for the sample image, 24 bits can be embedded in the cover image. If \( \beta \) is set to 20, and the APDLS and the ARDHA techniques are used to generate the prediction image for the sample image, the APDLS and ARDHA techniques can embed 22 and 25 bits in their cover images, respectively. These results occurred because the number of predictions increased as \( \beta \) increased, the frequencies at the highest and second-highest peak points in the histogram of the difference sequence increased, and the number of data bits that could be embedded increased. As the APD technique does not use the prediction image, the embedded bits were always equal to 15 bits, regardless of \( \beta \).

As shown in Figures 11 and 12, if the ARDHA technique was used to embed confidential data in the cover image, it could embed 8 and 5 more confidential data bits than the APD and APDLS techniques, respectively. In addition, the original confidential data could be extracted without data loss from the stego-image, and the original cover image could be restored without distortion from the stego-image. Therefore, the proposed ARDHA technique could be effectively used in digital watermarking, military, and medical applications.
4. Experimental Results

To evaluate the performance of the proposed ARDHA technique, experiments were carried out using Lena, blackb, Portofino, and ship, which were 512 images × 512 images shown in Figure 13, as cover images. The abstract in this paper was converted to ASCII code and used as confidential data. In the experiments, the confidential data were embedded in the cover image using proposed and existing techniques. The image generated by embedding confidential data in the cover image is called a stego-image.

Figure 13. Cont.
The values of the variables used in this paper were as follows: the alpha (\(\alpha\)) used in Equations (9) to (10) was set to 0.45. In general, local similarity exists in images. Therefore, the closer the pixels are, the more similar the pixel values. To reflect this characteristic, when calculating the predicted value \(V_1\) using Equation (9), the weight of the adjacent pixels affecting the calculation of \(V_1\) was set to 1. The \(\alpha\) is the weight that pixels, which are one pixel apart, contribute to the \(V_1\) calculation. To reflect local similarity, we set \(\alpha\) to 0.45, which is much smaller than 1.

The experiment was carried out while changing the threshold (\(\beta\)), which is used in Step 1.4 to 1.5, from 0 to 20. The reference value (\(\varepsilon\)) used in Step 1.7 was set to 10. The dev2 obtained by Equation (17) is the sum of the absolute values of the differences between the avg and the neighboring pixels. If dev2 is much larger than the avg, it is very unlikely that it will be a simple concave surface or a simple convex surface because the change in pixel values around the location is large. For this reason, \(\varepsilon\) was set to 10; thus, the deviation between the avg and each pixel is less than 1 on average.

The \(K\) used in Equation (26) was set to 4. In the cases of simple convex and simple concave surfaces, the pixel values change gently. Therefore, the average value can be used to more accurately predict the pixel values in a simple convex surface or a simple concave surface. \(K\) was set to 4 to reflect the avg with the same weight as cont0[\(x_i\)], con1[\(x_i\)], con2[\(x_i\)], and con3[\(x_i\)].

When predicting pixel values using edge characteristics, strong_edge was set to 1, normal_edge was set to 0.75, weak_edge was set to 0.5, very_weak_edge was set to 0.25, and non-edge was set to 0 in Equation (35). Specific values were set owing to the following reason: The stronger the edge characteristic, the more it should be reflected in the pixel value prediction. Therefore, by setting no_edge to 0, no_edge has no influence on the pixel value prediction. The normal_edge and weak_edge have values evenly distributed between 1 and 0.

The difference threshold (\(\delta\)) used in Equation (37) was set to 1.6. To satisfy local similarity, the predicted value \(V_{4X}\) must have a value close to \(V_2\). Therefore, the \(\delta\) should be set to a small value.

The \(R_S\), \(R_N\), and \(R_W\) used in Equations (36), (38), and (39) are 1.1, 0.9, and 0.7, respectively. \(R_S\) was determined based on the average of all edge values at all locations with one or more strong edges. \(R_N\) was determined based on the average of all edge values at all locations with no strong edges and at
least one normal edge. \( R_W \) was determined based on the average of all edge values at all locations with no strong or normal edges, and with at least one weak edge.

Lena, blackb, Portofino, and ship images used as cover images are shown in Figure 13a.1–d.1. The stego-images, which are the experimental result images generated by embedding confidential data in cover images using the APD, APDLS, and ARDHA techniques, are shown in Figure 13a.2–a.11,b.2–b.11,c.2–c.11,d.2–d.11. Figure 13a.2 is a stego-image, which is generated by embedding confidential data into a Lena image using the APD, APDLS(0), and ARDHA-(0) techniques. As explained on the next page, the APD, APDLS(0), and ARDHA-(0) techniques worked the same way. Therefore, as shown in Figure 13a.2, the stego-images generated by embedding confidential data in the cover image using APD, APDLS(0), and ARDHA-(0) techniques are identical. Figure 13b.2,c.2,d.2 are stego-images generated by embedding confidential data in blackb, Portofino, and ship using the APD, APDLS(0), and ARDHA-(0) techniques. Figure 13a.3–a.6 are stego-images generated by embedding confidential data into a Lena image using APDLS technique, where \( \beta \) is 5, 10, 15, and 20, respectively. Figure 13a.7–a.11 includes stego-images generated by embedding confidential data in Lena image using the ARDHA technique, where \( \beta \) is 0, 5, 10, 15, and 20, respectively. As shown in Figure 13, the visual quality of stego-image generated by applying ARDHA technique is very high. The cover image and the stego-image are visually identical. Thus, it is not possible to visually discern whether confidential data are embedded in the stego-image.

Table 1 shows the quantitative values that measured the performance of the proposed technique via experimentation. Table 1 consists of two parts. The performance test results obtained by performing experiments on each of the techniques for calculating the seven predicted pixel values (i.e., \( V_1 \), \( V_2 \), \( V_3 \), \( V_3C \), \( V_4S \), \( V_4N \), and \( V_4W \)) used in the ARDHA technique are shown on the right side of the table. The overall performance test results of the ARDHA technique using all seven predicted pixel values are shown on the left side.

The second to eighth columns of Table 1 show the overall results of experiments applying the ARDHA technique to embed confidential data in each cover image. The items of the experimental results from the second to eighth columns in Table 1 are number of hidden bits, PSNR (dB), prediction hit ratio (%), mean of prediction errors, number of predictions, and increase rate of hidden bits (%), respectively. Number of hidden bits represents the number of confidential data bits hidden in the cover image. PSNR (dB) is a measured value that represents the quality of a stego-image. Prediction hit ratio (%) is the number of accurate predictions divided by the total number of predictions, and subsequently multiplied by 100. Mean of the prediction errors represents the sum of the absolute values of the prediction errors divided by the total number of predictions. Number of predictions represents the total number of times the prediction was executed. Increase rate of hidden bits (%) represents the rate of increase in the number of hidden bits expressed as a percentage in the ARDHA and APDLS techniques compared to the APD technique. Compared with the APDLS technique, the increase rate of hidden bits in the ARDHA technique is shown in parentheses.

The last seven columns in Table 1 contain the experimental results of the prediction hit ratio, the mean of prediction errors, and the number of predictions measured when predicting \( V_1 \), \( V_2 \), \( V_3 \), \( V_3C \), \( V_4S \), \( V_4N \), and \( V_4W \). P.H.R (%) represents the prediction hit ratio, M.P.E represents the mean of the prediction errors, and N. P represents the number of predictions.
| Image | Algorithm | Number of Hidden Bits | PSNR (dB) | Prediction Hit Ratio (%) | Mean of Prediction Errors | Number of Predictions | Increase Rate of Hidden Bits (%) | V1 PHR(%) | V1 M.P.E | V1 N.P | V2 PHR(%) | V2 M.P.E | V2 N.P | V3 PHR(%) | V3 M.P.E | V3 N.P | V4 PHR(%) | V4 M.P.E | V4 N.P | V5 PHR(%) | V5 M.P.E | V5 N.P | V6 PHR(%) | V6 M.P.E | V6 N.P | V7 PHR(%) | V7 M.P.E | V7 N.P | V8 PHR(%) | V8 M.P.E | V8 N.P | V9 PHR(%) | V9 M.P.E | V9 N.P | V10 PHR(%) | V10 M.P.E | V10 N.P | V11 PHR(%) | V11 M.P.E | V11 N.P | V12 PHR(%) | V12 M.P.E | V12 N.P | V13 PHR(%) | V13 M.P.E | V13 N.P | V14 PHR(%) | V14 M.P.E | V14 N.P | V15 PHR(%) | V15 M.P.E | V15 N.P | V16 PHR(%) | V16 M.P.E | V16 N.P | V17 PHR(%) | V17 M.P.E | V17 N.P | V18 PHR(%) | V18 M.P.E | V18 N.P | V19 PHR(%) | V19 M.P.E | V19 N.P | V20 PHR(%) | V20 M.P.E | V20 N.P | V21 PHR(%) | V21 M.P.E | V21 N.P | V22 PHR(%) | V22 M.P.E | V22 N.P | V23 PHR(%) | V23 M.P.E | V23 N.P | V24 PHR(%) | V24 M.P.E | V24 N.P | V25 PHR(%) | V25 M.P.E | V25 N.P | V26 PHR(%) | V26 M.P.E | V26 N.P | V27 PHR(%) | V27 M.P.E | V27 N.P |
|-------|-----------|-----------------------|---------|-------------------------|--------------------------|------------------------|--------------------------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|-------|-----------|---------|------- |
| Image    | Algorithm | Number of Hidden Bits | PSNR (dB) | Prediction Hit Ratio (%) | Mean of Prediction Errors | Number of Predictions | Increase Rate of Hidden Bits (%) | $V_1$  | $V_2$  | $V_{SV}$ | $V_{SC}$ | $V_{SR}$ | $V_{SN}$ | $V_{IN}$ |
|----------|------------|----------------------|-----------|--------------------------|--------------------------|-----------------------|---------------------------------|--------|--------|----------|----------|----------|----------|----------|
| APDLS(10) | 66174      | 39.617               | 21.759    | 1.44                      | 67855                   | 15.9                  | 22.309                          | 20.631 | 4543   | 22132    | 1.41     | 1.51     | -        | -        |
| ARDHA(10) | 72033      | 35.426               | 19.556    | 1.72                      | 132258                  | 26.2                  | 22.288                          | 20.631 | 1.42   | 1.51     | 1.77     | 1.75     | 1.91     | 1.95     |
|           | (9.9)      |                      |           |                          |                          | 45167                  | 22132                | 16.197                          | 16.667 | 16.667  | 18.008   | 17.988   | 16.611   |
|           |            |                      |           |                          |                          | 21.013                  | 19.492                | 142                | 144     | 7541    | 24466    | 3266     |
|           | blackb     |                      |           |                          |                          | 21.000                  | 19.492                | 142                | 144     | 7541    | 24466    | 3266     |
|           | (9.9)      |                      |           |                          |                          | 99044                  | 37534                 | 142                | 144     | 7541    | 24466    | 3266     |
| APDLS(15) | 72885      | 36.200               | 20.595    | 1.55                      | 136578                  | 27.6                  | 21.000                          | 19.492 | 142     | 144      | 1963     | 6739     | 10647    |
| ARDHA(15) | 73463      | 34.813               | 19.628    | 1.70                      | 155927                  | 28.7                  | 21.000                          | 19.492 | 142     | 144      | 1963     | 6739     | 10647    |
|           |            | (0.8)                |           |                          |                          | 98758                  | 37534                 | 142                | 144     | 7541    | 24466    | 3266     |
|           |            |                      |           |                          |                          | 20.141                 | 18.893                | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 115062                 | 42465                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 20.128                 | 18.893                | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 1.63                    | 1.70                   | 1.77               | 1.75    | 3.32    | 3.28     | 3.38     |
|           |            |                      |           |                          |                          | 114776                 | 42465                 | 142                | 144     | 944     | 3534     | 5739     |
| ARDHA(20) | 73828      | 34.387               | 19.273    | 1.75                      | 167744                  | 29.3                  | 21.000                          | 19.492 | 20.631  | 4543     | 22132    | 1.41     | 1.51     | -        |
|           |            | (0.2)                |           |                          |                          | 21.000                  | 19.492                | 142                | 144     | 7541    | 24466    | 3266     |
|           |            |                      |           |                          |                          | 115062                 | 42465                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 20.128                 | 18.893                | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 1.63                    | 1.70                   | 1.77               | 1.75    | 3.32    | 3.28     | 3.38     |
|           |            |                      |           |                          |                          | 114776                 | 42465                 | 142                | 144     | 944     | 3534     | 5739     |
| ARDHA(0)  | 30859      | 35.438               | 9.67      | 3.51                      | 33879                   | 9.0                   | 0.000                           | 0.000  | 2.00   | 7.00     | -        | -        | -        | -        |
|           | (9.0)      |                      |           |                          |                          | 0.000                  | 2.00                 | 7.00               | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           | Portofino  | 30859                | 35.438    | 9.669                     | 33880                   | 9.0                   | 0.000                           | 0.000  | 2.00   | 7.00     | -        | -        | -        | -        |
|           | (9.0)      |                      |           |                          |                          | 0.000                  | 2.00                 | 7.00               | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
|           |            |                      |           |                          |                          | 2.00                   | 7.00                 | -                  | -       | -       | -        | -        |
| Image | Algorithm | Number of Hidden Bits | PSNR (dB) | Prediction Hit Ratio (%) | Mean of Prediction Errors | Number of Predictions | Increase Rate of Hidden Bits (%) | \( V_1 \) | \( V_2 \) | \( V_{AV} \) | \( V_{SC} \) | \( V_{AR} \) | \( V_{AN} \) | \( V_{AV} \) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Portofino | APDLS(20) | 31539 | 35.949 | 11.203 | 2.93 | 45621 | 11.4 | 11.264 | 10.886 | - | - | - | - | - |
| ARDHA(20) | 32468 | 33.627 | 10.375 | 3.26 | 62516 | 14.7 | (2.9) | 2.91 | 2.99 | - | 0.000 | 9.134 | 8.336 | 7.910 |
| APD | 46430 | 48.573 | - | - | - | - | - | 11.246 | 11.068 | 11.246 | 2.91 | 2.99 | 5.00 | 4.06 |
| APDLS(0) | 46430 | 48.573 | - | - | - | - | - | 11.246 | 11.068 | 11.246 | 2.91 | 2.99 | 5.00 | 4.06 |
| ARDHA-(0) | 46430 | 48.573 | - | - | - | - | - | 11.246 | 11.068 | 11.246 | 2.91 | 2.99 | 5.00 | 4.06 |
| ARDHA(0) | 54880 | 36.276 | 19.142 | 1.84 | 89578 | 18.2 | (18.2) | - | - | - | - | - | - | - |
| ARDHA(5) | 54880 | 36.273 | 19.142 | 1.84 | 89632 | 18.2 | (17.9) | - | - | - | - | - | - | - |
| ARDHA(10) | 55207 | 36.103 | 19.254 | 1.80 | 124825 | 20.8 | (1.6) | - | - | - | - | - | - | - |
| ARDHA(15) | 56110 | 35.590 | 19.254 | 1.80 | 113865 | 20.8 | (1.6) | - | - | - | - | - | - | - |
| ARDHA(20) | 56529 | 35.060 | 18.787 | 1.86 | 124825 | 21.8 | (0.6) | - | - | - | - | - | - | - |
In Figure 13 and Table 1, APDLS(\(v\)) represent the case where the experiments were carried out by setting \(\beta\) to \(v\) in the APDLS technique. ARDHA-(\(v\)) represents the ARDHA technique where \(V_{3V}, V_{3C}, V_{4S}, V_{4N}, \) and \(V_{4W}\) are not used and \(\beta\) is set to \(v\). Thus, only \(V_1\) and \(V_2\) are used in ARDHA-(\(v\)).

ARDHA(\(v\)) indicates the case where \(\beta\) is set to \(v\) in the ARDHA technique and uses \(V_1, V_2, V_{3V}, V_{3C}, V_{4S}, V_{4N}, \) and \(V_{4W}\). ARDHA(\(0\)) indicates the case where \(\beta\) is set to \(0\) in the ARDHA technique. As \(\beta\) is 0, only \(V_{3V}, V_{3C}, V_{4S}, V_{4N}, \) and \(V_{4W}\) are used in ARDHA(\(0\)), not \(V_1\) and \(V_2\).

For ARDHA-(\(0\)), as \(\beta\) is zero, \(V_1\) and \(V_2\) predictions are not carried out. Therefore, the pixel value of the cover image becomes the pixel value of the prediction image. Thus, the cover image and the prediction image are the same. Thus, as shown in Figure 13 and Table 1, APD and APDLS(\(0\)) and ARDHA-(\(0\)) worked the same way, and the stego-images generated by applying APD and APDLS(\(0\)) and ARDHA-(\(0\)) were all the same.

As shown in the last seven columns of Table 1, the larger the \(\beta\), the greater the number of pixel value predictions obtained using the characteristics of the flat, simple inclined, simple convex, and simple concave surfaces in the image. However, the instances of predicting the pixel values using edge characteristics are reduced. If \(\beta\) is a small value, the number of \(V_{3V}\) and \(V_{3C}\) predictions is greatly reduced. However, as \(\beta\) increases, the number of \(V_{3V}\) and \(V_{3C}\) predictions increases, and the accuracy of pixel value prediction using simple convex surface and simple concave surface characteristics approaches to a high value. These features of the proposed technique can also be easily seen in the flowchart of pixel value prediction in Figure 6. That is, as \(\beta\) increases, on one hand, the case where \(\beta\) is greater than dev1 in Figure 6 increases, and the prediction number for \(V_1, V_2, V_{3V}, \) and \(V_{3C}\) increases. On the other hand, the case where dev1 is greater than or equal to \(\beta\) decreases, and the number of predictions for \(V_{4S}, V_{4N}, \) and \(V_{4W}\) decreases. Additionally, as \(\beta\) increases, the number of predictions for \(V_{3V}\) and \(V_{3C}\) increases from a small value to a large value, and the prediction hit ratio converges to a constant value.

As shown in the second to eighth columns of Table 1, when predicting pixel values using the ARDHA technique, the number of predictions is very large, the prediction hit ratio is high, and the mean of prediction errors is very small. As the \(\beta\) increases in ARDHA technique, the number of predictions increases. Therefore, a large number of predicted pixel values are used in the prediction image, the prediction image sequence (P) includes a large number of predicted pixel values. Thus, the ARDHA technique increases the frequencies at the highest and second-highest peak points in the histogram of the difference sequence generated by using the cover image and the prediction image. Therefore, the number of confidential data bits that can be embedded in the cover image is increased. For this reason, as indicated in the third and eighth columns of Table 1, as \(\beta\) increases in the ARDHA technique, the number of hidden bits increases, increasing the rate of hidden bits compared to the APD technique.

As shown in Equation (40), when the pixel value prediction is accurate, \(P_i\) is the same as \(C_i\). Thus, \(D_i = C_i - P_i = 0\). The more accurate the prediction is, the closer the \(D_i\) value is to zero. Therefore, the more accurate the pixel value prediction, the greater the frequencies at the highest and second-highest peak points in the histogram of the difference sequence. Thus, the greater the number of confidential data bits that can be embedded.

When the ARDHA technique was applied, \(D_i\) was very small because the mean of prediction errors is very small as shown in Table 1. Thus, the ARDHA technique greatly increases the frequency at the highest and second-highest peak points in the histogram of a difference sequence. Therefore, the number of confidential data bits that can be embedded in the cover image is greatly increased.

The prediction hit ratio in the fifth column of Table 1 was the same as the value calculated using PH.R and N.P in the last seven columns. The number of predictions in the seventh column of Table 1 was equal to the sum of the N.Ps in the last seven columns. In addition, when the threshold (\(\beta\)) was 5 or more, the total number of prediction times for \(V_1\) and \(V_2\) in the APDLS technique was the same as the total number of prediction times for \(V_1, V_2, V_{3V}, \) and \(V_{3C}\) in the ARDHA technique. These points could be easily seen in the flowchart of pixel value prediction in Figure 6. That is, if \(\beta\) is greater
than 0 and dev1 is less than β, the predicted pixel value is determined between V_{1}, V_{2} in the APDLS technique, and the predicted pixel value is determined among V_{1}, V_{2}, V_{3V}, and V_{3C} in the ARDHA technique. Therefore, the number of predictions for V_{1} and V_{2} in the APDLS technique was the same as the number of predictions for V_{1}, V_{2}, V_{3V}, and V_{3C} in the ARDHA technique.

As shown in the fourth column of Table 1, in the ARDHA technique, increasing β from 0 to 20 decreased the PSNR value of the stego-image. However, we could observe in Figure 13 that the visual quality of the stego-image remained very good. Thus, it was not possible to visually discern whether confidential data are embedded in the stego-image. As shown in Figure 13, the cover image and the stego-image were visually identical. As shown in the eighth column of Table 1, when confidential data was embedded in the cover image using the ARDHA technique, the number of hidden bits increased by 29.3% and 24.5% at the maximum compared to the APD and APDLS techniques, respectively.

By applying the proposed technique, it is possible to hide confidential data in the cover image at various levels while maintaining excellent visual quality of the stego-image. The confidential data can be extracted without data loss from the stego-image, and the original cover image can be recovered without distortion from the stego-image. Therefore, the proposed technique can be very useful for digital image watermarking, military, medical applications.

5. Conclusions

In this paper, we proposed the ARDHA technique that can reversibly embed a large amount of confidential data in a cover image at various levels. In the proposed ARDHA technique, a prediction image was generated using seven kinds of predicted pixel values, and the prediction image was used to embed confidential data in the cover image. Due to the high accuracy of the seven kinds of pixel value predictions used in the ARDHA technique, a prediction image closely approximating the cover image was generated. Thus, by embedding confidential data in the cover image using a prediction image closely approximating the cover image, a large amount of confidential data could be embedded at various levels, while maintaining the excellent visual quality of the stego-image.

As the visual quality of a stego-image generated by applying ARDHA technique was very high, it was not possible to distinguish the stego-image from the cover image. Thus, it was not possible to visually discern whether confidential data are embedded in the stego-image. Using the ARDHA technique, the number of confidential data bits that could be embedded in the cover image was increased by 29.3% and 24.5% compared to the APD and APDLS techniques, respectively. In addition, the confidential data could be extracted without data loss from the stego-image and the original cover image could be restored without distortion from the stego-image. Therefore, the proposed ARDHA technique was an efficient reversible data hiding algorithm that is very useful in digital image watermarking, military, and medical applications.

6. Discussion

Compared to the previous research, we proposed an effective reversible data hiding technique that uses various curved surface characteristics and various edge characteristics such as simple convex surface, simple concave surface, strong edge, normal edge, and weak edge to hide large amounts of confidential data in the cover image at various levels. The experiments proved the superiority of the proposed technique. However, unlike the previous researches, the execution time of the proposed technique was increased because the parts that accurately predict pixel values were added using the aforementioned surface characteristics and edge characteristics. Therefore, in the proposed technique, there is a trade-off between the amount of hidden data and the execution time, which requires further research. In addition, the applicability of the proposed technique can be improved by conducting research on hiding confidential data in color images, text data, and voice data based on the proposed technique in this paper.

Author Contributions: All authors of this article discussed the contents of the manuscript and actively contributed in the processing of implementation. Conceptualization, S.-M.J.; methodology, S.-M.J.; software, S.-M.J.; validation,
B.-W.O.; formal analysis, S.-M.J.; investigation, B.-W.O.; resources, S.-M.J.; data curation, S.-M.J.; writing—original draft preparation, S.-M.J.; writing—review and editing, S.-M.J.; visualization, S.-M.J.; supervision, B.-W.O.; project administration, S.-M.J.; funding acquisition, S.-M.J. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by Sahmyook University Research Fund, 2017.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. Ni, Z.; Shi, Y.Q.; Ansari, N.; Su, W. Reversible data hiding. *IEEE Trans. Circuits Syst. Video Technol.* 2006, 16, 354–362.

2. Li, Y.C.; Yeh, C.M.; Chang, C.C. Data hiding based on the similarity between neighboring pixels with reversibility. *Digit. Signal Process.* 2010, 20, 1116–1128. [CrossRef]

3. Tsai, C.L.; Fan, K.C.; Chung, C.D.; Chuang, T.C. Reversible and lossless data hiding with application in digital library. In Proceedings of the 38th Annual 2004 International Canahan Conference on Security Technology, Albuquerque, NM, USA, 11–14 October 2004; pp. 226–232.

4. Chang, C.K.; Cheng, L.M. Hiding data in images by simple LSB substitution. *Pattern Recognit.* 2004, 37, 469–474. [CrossRef]

5. Mustafa, A.E.; ElGamal, A.M.F.; ElAlmi, M.E.; ElAlmi, M.E. A Proposed Algorithm for Steganography in Digital Image Based on Least Significant Bit. *Res. J. Specif. Educ.* 2011, 21, 144–150.

6. Zargar, A.J. Digital Image Watermarking using LSB Technique. *Int. J. Eng. Res. Gen. Sci.* 2015, 3, 1424–1433.

7. Chitradevi, B.; Thinaharan, N.; Vasanthi, M. Data Hiding Using Least Significant Bit Steganography in Digital Images. *Stat. Approaches Multidiscip. Res.* 2017, 1, 143–150.

8. Halder, T.; Karforo, S.; Mandal, R. A Block-Based Adaptive Data Hiding Approach Using Pixel Value Difference and LSB Substitution to Secure E-Governance Documents. *J. Inf. Process. Syst.* 2019, 15, 261–270.

10. Arai, K. Improvement of invisibility of secret images embedded in circulate images based on MRA with coordinate conversion and Principal Component Analysis: PCA. *J. Image Electron. Soc. Jpn.* 2007, 36, 665–673.

11. Arai, K.; Seto, K. Improvement of invisibility of secret images embedded in circulate images based on MRA with scanning scheme conversion. *Vis. Soc. Jpn.* 2009, 29, 167–170.

12. Nag, A.; Biswas, S.; Sarkar, D.; Sarkar, P.P. A novel technique for image steganography based on DWT and Huffman encoding. *Int. J. Comput. Sci. Secur.* 2011, 4, 561–570.

13. Arai, K. Improvement of Security and Invisibility of Secret Images Embedded in Circulate Images and Based on MRA; Report of RIMS-Research Institute for Mathematical Sciences Kyoto University; Kyoto University: Kyoto, Japan, 2010; pp. 93–113.

14. Arai, K. Data Hiding Method Replacing LSB on Hidden Portion for Secrete Image with Run-Length Coded Image. *Int. J. Adv. Res. Artif. Intell.* 2016, 5, 8–14. [CrossRef]

15. Golabi, S.; Helfroush, M.S.; Danyali, H. Reversible robust data hiding based on wavelet filters Modification. *Multimed. Tools Appl.* 2019, 78, 31847–31865. [CrossRef]

16. Liu, L.; Chang, C.; Wang, A. Data hiding based on extended turtle shell matrix construction method. *Multimed. Tools Appl.* 2017, 76, 12233–12250. [CrossRef]

17. Xie, X.Z.; Lin, C.C.; Chang, C.C. Data Hiding Based on a Two-Layer Turtle Shell Matrix. *Symmetry* 2018, 10, 47. [CrossRef]

18. Wang, F.; Guo, Y.; Yin, Z.; Zhou, X.; Zhao, Q. Data Hiding Method based on Reference Matrix. *Procedia Comput. Sci.* 2018, 131, 800–809. [CrossRef]

19. Bharti, J.; Pateriya, R.K.; Shukla, S. Block Wise Data Hiding with Auxiliary Matrix. *Int. J. Adv. Comput. Sci. Appl.* 2017, 8, 128–135.

20. Li, X.; Li, J.; Li, B.; Yang, B. High-fidelity reversible data hiding scheme based on pixel-value-ordering and prediction-error expansion. *Signal Process.* 2013, 93, 198–205. [CrossRef]

21. Peng, F.; Li, X.; Yang, B. Improved pvo-based reversible data hiding. *Digit. Signal Process.* 2014, 25, 55–265. [CrossRef]
22. Qu, X.; Kim, H.J. Pixel-based pixel value ordering predictor for high-fidelity reversible data hiding. *Signal Process.* **2015**, *111*, 249–260. [CrossRef]

23. Weng, S.; Shi, Y.Q.; Hong, W.; Yao, Y. Dynamic improved pixel value ordering reversible data hiding. *Inf. Sci.* **2019**, *489*, 136–154. [CrossRef]

24. Fallahpour, M.; Sedaaghi, M.H. High capacity lossless data hiding based on histogram modification. *IEICE Electro. Express* **2007**, *4*, 205–210. [CrossRef]

25. Tseng, H.W.; Hsieh, C.P. Reversible data hiding based on image histogram modification. *Imaging Sci. J.* **2008**, *56*, 271–278. [CrossRef]

26. Zhang, X. Separable reversible data hiding in encrypted image. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 826–832. [CrossRef]

27. Li, X.; Li, B.; Yang, B.; Zeng, T. General Framework to Histogram shifting based Reversible Data Hiding. *IEEE Trans. Image Process.* **2013**, *22*, 2181–2191. [CrossRef]

28. Lo, C.C.; Hu, Y.C.; Chen, W.L.; Wu, C.M. Reversible Data Hiding Scheme for BTC-compressed Images based on Histogram Shifting. *Int. J. Secur. Its Appl.* **2014**, *8*, 301–314. [CrossRef]

29. Sabeen Govind, P.V.; Wilscy, M. A New Reversible Data Hiding Scheme with Improved Capacity based on Directional Interpolation and Difference Expansion. *Procedia Comput. Sci.* **2015**, *46*, 491–498. [CrossRef]

30. Bonde, R.; Dahake, V.; Dhengre, S. Reversible data hiding with a general framework through histogram shifting method. *Int. J. Innov. Sci. Eng. Technol.* **2015**, *2*, 449–452.

31. Chandrasekaran, V.; Sevugan, P. Applying Reversible Data Hiding for Medical Images in Hybrid Domain Using Haar and Modified Histogram. *Int. J. Intell. Eng. Syst.* **2017**, *10*, 126–134. [CrossRef]

32. Arunkumar, S.; Subramaniyaswamy, V.; Sivaramakrishnan, N. Reversible Data Hiding scheme using modified Histogram Shifting in Encrypted Images for Bio-medical images. *Int. J. Pure Appl. Math.* **2018**, *119*, 13233–13240.

33. Sivasubramanian, N.; Konganathan, G.; Rao, Y.V.R. High capacity multi-bit data hiding based on modified histogram shifting technique. *ETRI J.* **2018**, *40*, 677–686. [CrossRef]

34. Chen, K.; Chang, C.C. High-capacity reversible data hiding in encrypted images based on two-phase histogram shifting. *Math. Biosci. Eng.* **2019**, *16*, 3947–3964. [CrossRef] [PubMed]

35. Jung, S.M. An advanced reversible data hiding algorithm based on the similarity between neighboring pixels. *J. Korea Soc. Comput. Inf.* **2016**, *21*, 33–42. [CrossRef]

36. Jung, S.M. Sahmyook University Industry-Academic Cooperation. A Method for Data Hiding Based on Pixel Value Predictions, a Method for Data Watermarking Using It, and an Apparatus for Data Hiding. Korea Patent- Registration Number 1017645300000, 27 July 2017. Available online: http://www.kipris.or.kr.

37. Jung, S.M.; On, B.W. Reversible data hiding algorithm using spatial locality and the surface characteristics of image. *J. Korea Soc. Comput. Inf.* **2016**, *21*, 1–12. [CrossRef]

38. Jung, S.M.; On, B.W. Sahmyook University Industry-Academic Cooperation. A Method for Reversible Data Hiding Based on Pixel Value Prediction According to Spatial Locality and Surface Characteristics, a Method for Reversible Watermarking Using It, and an Apparatus for Reversible Data Hiding, Reversible Watermarking. Korea Patent- Registration Number 1018754010000, 02 July 2018. Available online: http://www.kipris.or.kr.