

Chapter 5

Circular-Dotted Image --- A New Type of Image and Its Creation for Data Hiding by A Dot Overlapping Scheme

5.1 Overview of Proposed Method

In this chapter, we will investigate another form of art image. This new kind of art image has never been proposed. We call it *digital circular-dotted image*. A circular-dotted image is composed of numerous overlapping colorful circular-dots (denoted as *CDs*) with black borders. In Section 5.2, a method for automatic creation of digital circular-dotted images from an input digital image for data hiding applications will be proposed.

The proposed creation process is very similar to the digital pointillistic image creation process proposed in Section 4.2, but there are black borders around the *CDs* in a digital circular-dotted image. The size of the *CD* and the thickness of the black border (denoted as *TOB*) can be modified by users. Besides, we achieved the purpose of data hiding by changing the RGB values of the center of each colorful dot in a digital pointillistic image, but we achieve this purpose by applying a *dot overlapping* scheme to control the drawing order of the *CDs* in a digital circular-dotted image.

5.2 Proposed Digital Circular-Dotted Image Creation Process

5.2.1 Core Concept

The core concept of the proposed digital circular-dotted image creation process is shown in Figure 5.1. As mentioned before, the proposed image creation process is similar to that of a digital pointillistic image, therefore, we will refer to some of the parameters and algorithms which were defined and proposed in Chapter 4. While performing this process, we should apply the random array creation process discussed in Section 4.2.3 to disarrange the drawing order of the colorful circular-dots (denoted as CDs) of a digital circular-dotted image first.

After deriving a drawing order by performing the random array creation process, we start to perform a circular-dots drawing process. The proposed circular-dots drawing process is similar to the digital pointillistic image creation process proposed in Section 4.2.3, that is, we also utilize the derived drawing order and the CCA_i (meaning *Center-Color-Array*, which was defined in Chapter 4) of an input image to draw each CD_i in a digital circular-dotted image. However, before drawing each CD_i , we draw a black circular-dot (denoted as BD) first, and then draw the CD in it. The sizes of the CD and BD are decided by users. We utilize the “size difference” between the CD and BD to achieve the “black border effect.” The details of the proposed creation process will be discussed in the following section.

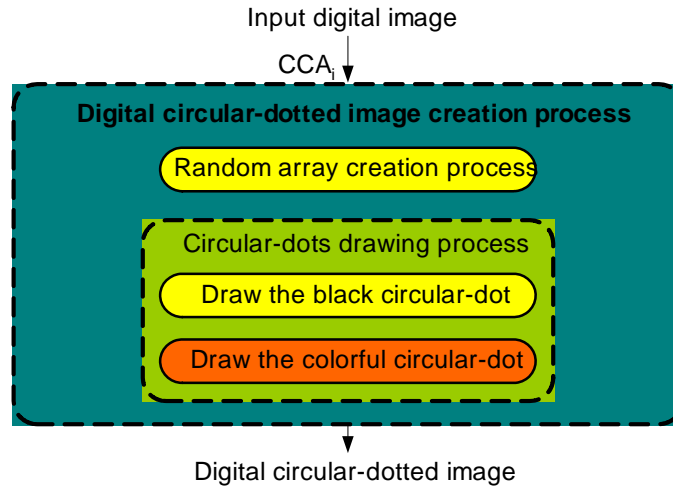


Figure 5.1 A flowchart of digital circular-dotted image creation process.

5.2.2 Proposed Creation Process

The parameter DAD defined in Chapter 4 is a fixed value of “5,” but the value of DAD in this chapter is decided by users. That is, the users can estimate the proper size of the CD , and then input the value of DAD while executing the program. Besides, the users should input a thickness value of the black order of each CD_i (denoted as TOB) while performing the proposed digital circular-dotted image creation process.

The detailed process is described as an algorithm below.

Algorithm 5.1: Digital circular-dotted image creation process.

Input: A digital image, a value of DAD , and a value of TOB .

Output: A digital circular-dotted image.

Steps:

- Step 1 Derive the CCA of an input image.
- Step 2 Figure out how many CDs will be drawn in a digital circular-dotted image by utilizing the input DAD , and denote the number of the CDs as NOD .

Step 3 Apply the random array creation process proposed in Section 4.2.3 to create a random array (denoted as RA), with the size of the RA being equal to the value of NOD .

$$\begin{cases} \text{if } DAD = \text{odd, then set} \\ BlackD = DAD + \frac{DAD+1}{2} \end{cases} ; \begin{cases} \text{if } DAD = \text{even, then set} \\ BlackD = DAD + \frac{DAD}{2} \end{cases} . \quad (5.1)$$

$$\text{set } ColorD = BlackD - TOB \times 2. \quad (5.2)$$

Step 4 Conduct the circular-dots drawing process by the following steps.

4.1 Figure out the diameters of BD_i and CD_i (denoted them as $BlackD$ and $ColorD$) by applying Formulas (5.1) and (5.2), respectively.

4.2 Decide a drawing order by applying the RA derived by Step 3. Draw each BD_i at its proper location, and successively draw CD_i on BD_i with its specific color (the color is got from CCA_i) determined in the following way:

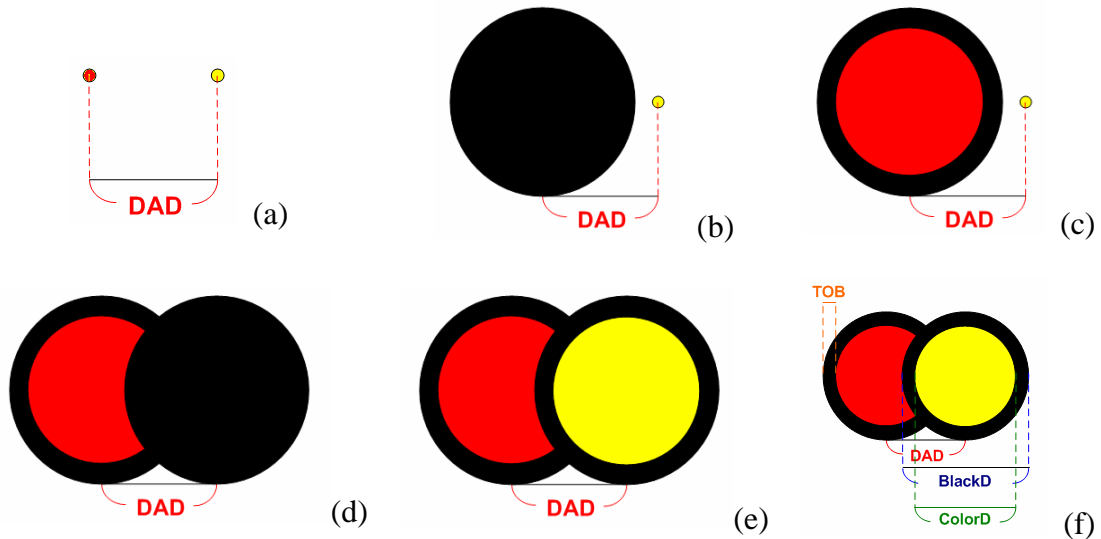


Figure 5.2 An illustration of the drawing steps (from (a) to (f)) of two circular-dots in a digital circular-dotted image.

5.2.3 Experimental Results and Discussions

Figure 5.3(a) is an input image. Figure 5.3(c) is the final digital circular-dotted image created from Figure 5.3(a) by applying the digital circular-dotted image creation process proposed in Section 5.2.2, and the input value of DAD is 11, the input value of TOB is 1. Figure 5.3(d) is another final digital circular-dotted image created from Figure 5.3(a), but the input value of DAD is 17, and the input value of TOB is 3.

Figure 5.3(b), (e), and (f) show some other experimental results of this section.

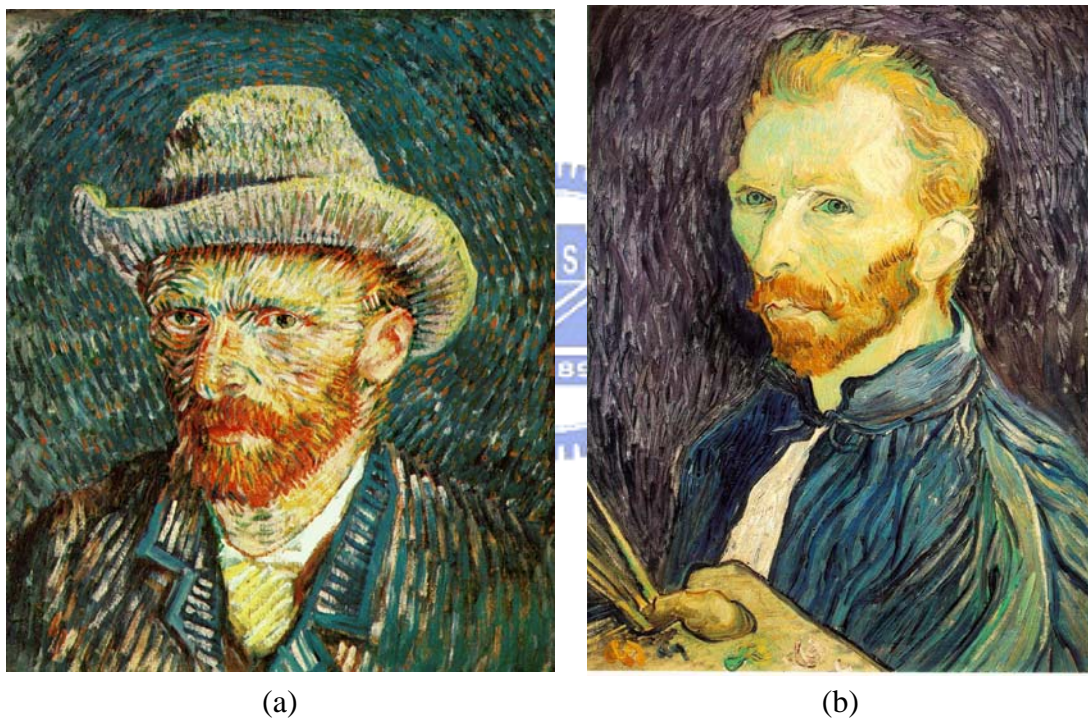


Figure 5.3 Experimental results. (a) An original image. (b) Another original image. (c) A digital circular-dotted image of (a) with inputs $DAD=11$ and $BOT=1$. (d) A digital circular-dotted image of (a) with inputs $DAD=17$ and $BOT=3$. (e) A digital circular-dotted image of (b) with inputs $DAD=11$ and $BOT=1$. (f) A digital circular-dotted image of (b) with inputs $DAD=17$ and $BOT=3$.



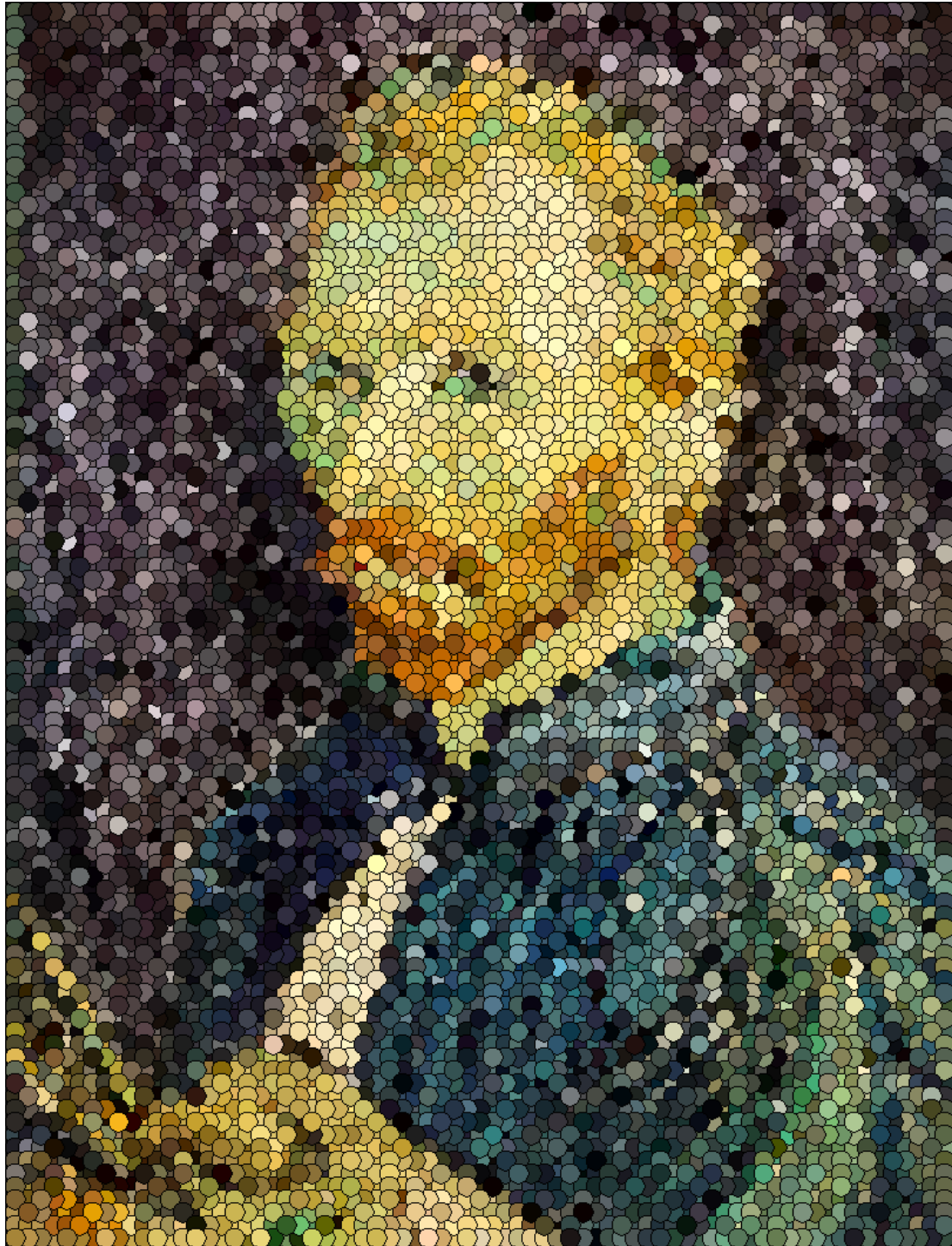
(c)

Figure 5.3 Experimental results. (a) An original image. (b) Another original image. (c) A digital circular-dotted image of (a) with inputs $DAD=11$ and $BOT=1$. (d) A digital circular-dotted image of (a) with inputs $DAD=17$ and $BOT=3$. (e) A digital circular-dotted image of (b) with inputs $DAD=11$ and $BOT=1$. (f) A digital circular-dotted image of (b) with inputs $DAD=17$ and $BOT=3$. (continued).



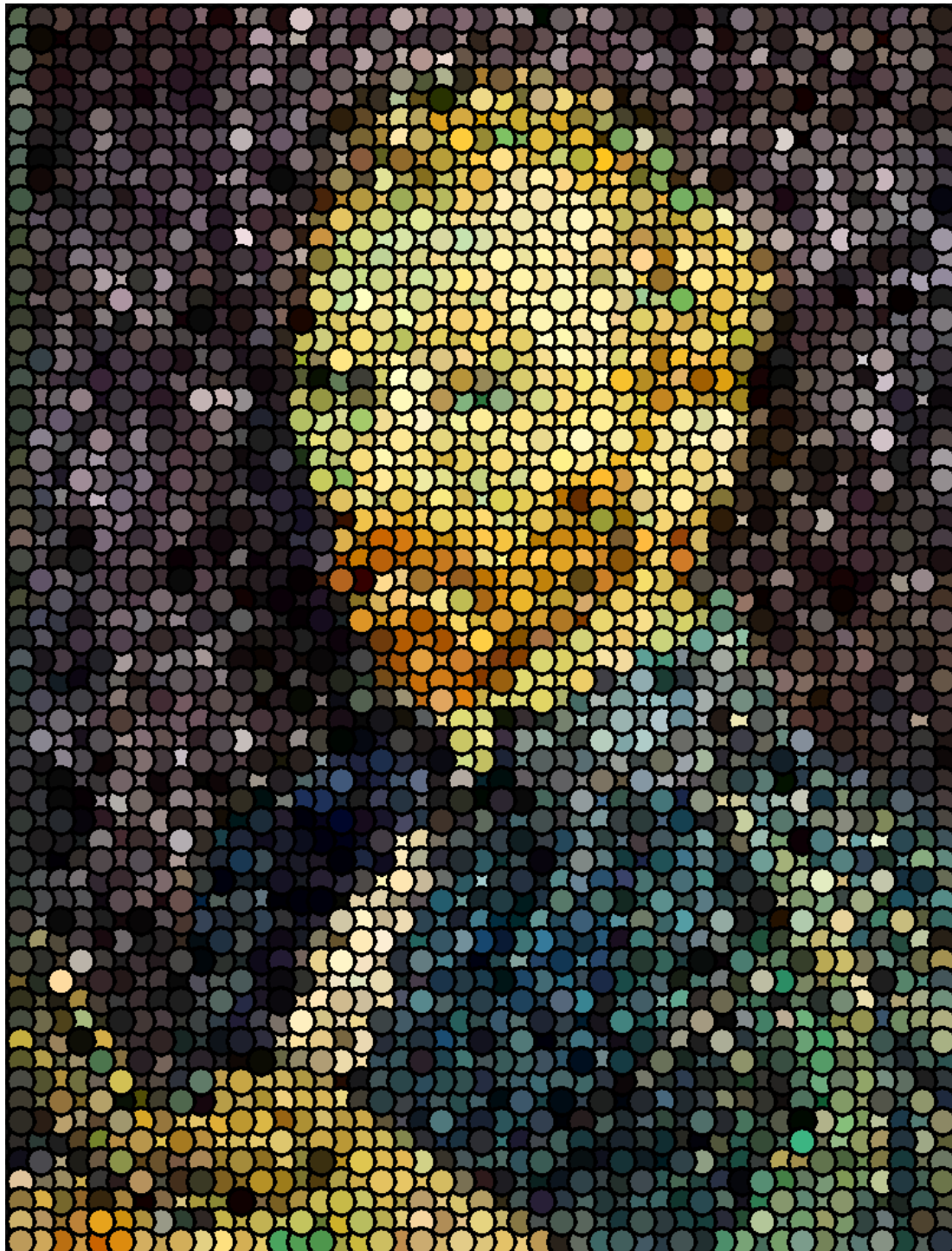
(d)

Figure 5.3 Experimental results. (a) An original image. (b) Another original image. (c) A digital circular-dotted image of (a) with inputs $DAD=11$ and $BOT=1$. (d) A digital circular-dotted image of (a) with inputs $DAD=17$ and $BOT=3$. (e) A digital circular-dotted image of (b) with inputs $DAD=11$ and $BOT=1$. (f) A digital circular-dotted image of (b) with inputs $DAD=17$ and $BOT=3$. (continued).



(e)

Figure 5.3 Experimental results. (a) An original image. (b) Another original image. (c) A digital circular-dotted image of (a) with inputs $DAD=11$ and $BOT=1$. (d) A digital circular-dotted image of (a) with inputs $DAD=17$ and $BOT=3$. (e) A digital circular-dotted image of (b) with inputs $DAD=11$ and $BOT=1$. (f) A digital circular-dotted image of (b) with inputs $DAD=17$ and $BOT=3$. (continued).



(f)

Figure 5.3 Experimental results. (a) An original image. (b) Another original image. (c) A digital circular-dotted image of (a) with inputs $DAD=11$ and $BOT=1$. (d) A digital circular-dotted image of (a) with inputs $DAD=17$ and $BOT=3$. (e) A digital circular-dotted image of (b) with inputs $DAD=11$ and $BOT=1$. (f) A digital circular-dotted image of (b) with inputs $DAD=17$ and $BOT=3$. (continued).

5.3 Proposed Data Hiding in Digital Circular-Dotted Images by Dot Overlapping Scheme

5.3.1 Core Concept

The main concept of data hiding in digital circular-dotted images by the dot overlapping scheme is to utilize the drawing order of the *CDs* in a digital circular-dotted image.

As shown in Figure 5.4(a), first, we input a bit code of secret information (denoted as Mes_i) and a bit code of a watermark (denoted as $Water_i$). By applying the information combination and disarrangement process proposed in Section 4.3.2, we can derive a bit code of disordered data (denoted as $DData_i$). Second, in order to derive the drawing order of the *CDs*, we perform a dot overlapping scheme, which will be discussed in Section 5.3.2 to deal with the $DData_i$. Finally, we utilize the derived drawing order to create a digital circular-dotted image by applying the image creation process proposed in Section 5.2.2.

Referring to the concept of data extracting shown in Figure 5.4(b), by applying a circular-dot detection process, which will be proposed in Section 5.3.3, we can extract a $DData_i$ from an input data-embedded digital circular-dotted image. Then, by applying the data recovering and separation process proposed in Section 4.3.5, we can finally derive the embedded Mes_i and $Water_i$.

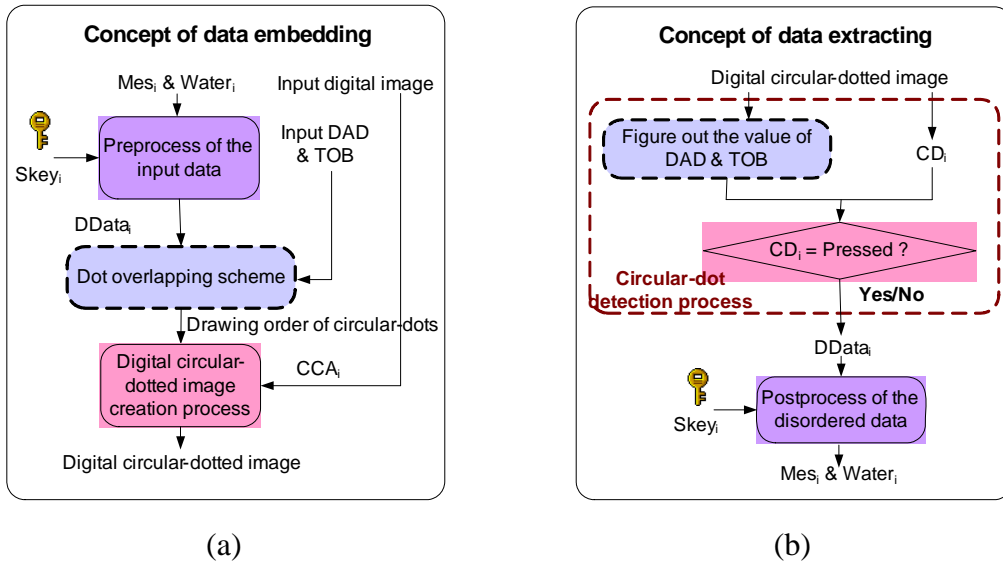


Figure 5.4 Core concepts of data embedding and extracting of a digital circular-dotted image.

5.3.2 Data Hiding Process

Before performing the proposed data hiding process of a digital circular-dotted image, we should preprocess the input data (Mes_i and $Water_i$) by applying the data combination and disarrangement process proposed in Section 4.3.2, and start up the random number generator using a secret key (denoted as $Skey_i$) to derive a bit code of disordered data (denoted as $DData_i$).

Besides, every pixel with the RGB value of (0, 0, 0) in the input digital image is replaced with one with the RGB value of (2, 2, 2), and we set the background color of an output digital circular-dotted image as black. Because of this pre-processing, there will be no pixel with the RGB value of (0, 0, 0) except the black borders of each CD_i , and the background color of the output digital circular-dotted image. This pre-processing is necessary in order to make the data extraction process smooth, and the details will be described in Section 5.3.3.

The details of the data hiding process are described as an algorithm below. Besides, an example of performing the proposed process is shown in Figure 5.5, and

the execution steps are illustrated in Figure 5.5(a) through (i).

Algorithm 5.2 : Data hiding in digital circular-dotted images by a dot overlapping scheme.

Input: An input digital image (denoted as I), a value of DAD , a value of TOB , and a bit sequence of a disordered data (denoted as $DData_i$).

Output: A two-dimensional array of bit sequences (denoted as $TCDA_{ij}$), and a drawing order of the CDs .

Steps:

Step 1 Create an empty two-dimensional array denoted as $TCDA_{ij}$.

1.1 Decide the size of the $TCDA_{ij}$.

1.1.1 Find out the height of I (denoted as $ImageHeight$), and the width of I (denoted as $ImageWidth$).

1.1.2 Derive the numbers of rows and columns of $TCDA_{ij}$ by $ImageHeight/DAD$ and $ImageWidth/DAD+1$, respectively.

1.2 Set the bit value of the first element of each row of $TCDA_{ij}$ as 1.

1.3 Set the bit value of the last element of each row of $TCDA_{ij}$ as 0.

1.4 Put each element of $DData_i$ into the empty space of $TCDA_{ij}$ in order.

1.5 After Step 1.4, if there still has empty space in $TCDA_{ij}$, randomly set the value of each redundant element in $TCDA_{ij}$ as 0 or 1.

Step 2 Decide the drawing order of the CDs in the following way.

2.1 For each row, if it satisfies the following conditions, utilize the “index of the found bit” in order, and apply the digital circular-dotted image creation process proposed in Section 5.2.2 to draw BDs and CDs :

2.1.1 From left to right : If the bit value of the element of the row is “0,” and the successive bit is “1.”

2.1.2 From left to right : If the bit value of the element of the row is “1,” and the successive bit is “1.”

2.1.3 From right to left : If the bit value of the element of the row is “0,” and the successive bit is “0.”

2.1.4 From left to right : If the bit value of the element of the row is “1,” and the successive bit is “0.”

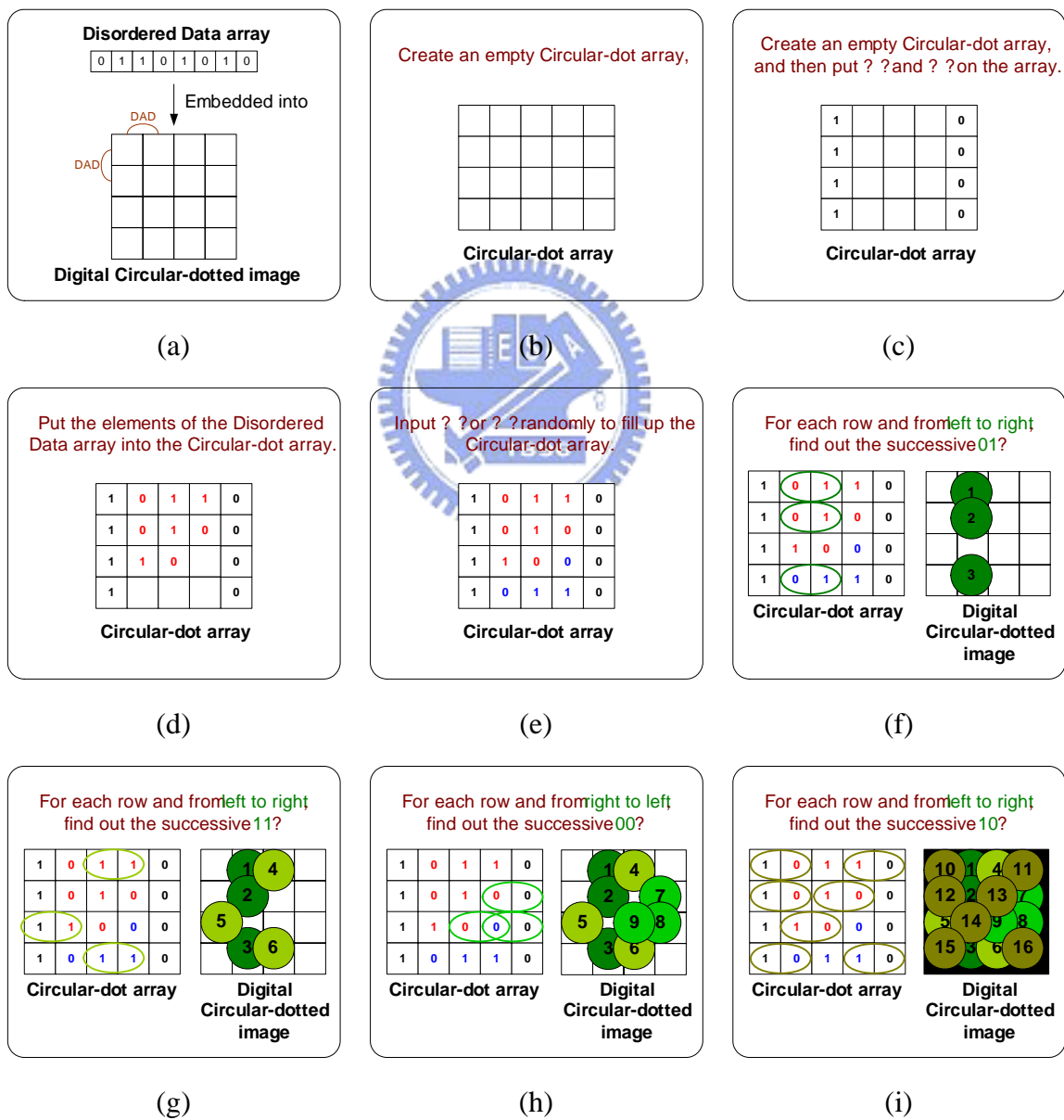


Figure 5.5 An example of performing data hiding in digital circular-dotted images by dot overlapping scheme. The colors of the CDs are get from the CCA of an original input digital image.

5.3.3 Data Extraction Process

The main concept of the data extraction process is to figure out whether a CD in a digital circular-dotted image is “*press on*” or “*pressed by*” a successive CD . If the CD *presses on* the successive CD , we know that the secret bit embedded between the two successive CDs is “0,” and if the CD is *pressed by* the successive CD , we know that the secret bit embedded between the two successive CDs is “1.”

Because every pixel with the RGB value of (0, 0, 0) in an input digital image is replaced with one with the RGB value of (2, 2, 2), and we set the background color of an output digital circular-dotted image as black, there will be no pixel with the RGB value of (0, 0, 0) except the black borders of each CD_i , and the background color of the output digital circular-dotted image.

After performing the circular-dot detection process of a digital circular-dotted image, we will derive a disordered bit sequence of the combined bit sequence of Mes_i and $Water_i$ (denoted as $DData_i$). Then, we apply the data recovering and separation process proposed in Section 4.3.5 to recover the embedded secret data (Mes_i and $Water_i$).

The details of the circular-dot detection process are described as an algorithm below.

Algorithm 5.3: Circular-dot detection process.

Input: A data-embedded digital circular-dotted image (denoted as I'), and a bit sequence of a secret key (denoted as $SKey_i$).

Output: A bit sequence of a disordered data (denoted as $DData_i$).

Steps:

Step 1 Find the coordinates of the center of the first CD (denoted as CC_1).

1.1 Perform a raster scan row by row to scan I' .

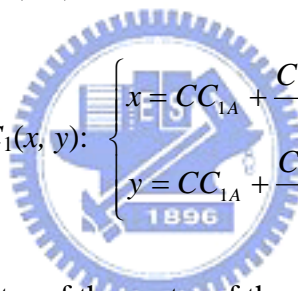
1.1.1 If a black pixel is detected, and the successive pixel is a non-black pixel:

1.1.1.1 Denote the position of the non-black pixel as CC_{1A} , and continue scanning.

1.1.2 If a non-black pixel is detected, and the successive pixel is a black pixel:

1.1.2.1 Denote the position of the non-black pixel as CC_{1B} .

1.1.3 Derive the x -coordinate and y -coordinate of CC_1 by applying Formula (5.3) below:


$$CC_1(x, y): \left\{ \begin{array}{l} x = CC_{1A} + \frac{CC_{1B} - CC_{1A}}{2} \\ y = CC_{1A} + \frac{CC_{1B} - CC_{1A}}{2} \end{array} \right\}. \quad (5.3)$$

Step 2 Find out the coordinates of the center of the second CD (denoted as CC_2).

2.1 Continue performing the raster scan mentioned in Step 1, and repeat Steps 1.1.1 through 1.1.3 to derive CC_2 .

Step 3 Figure out the distance between the centers of the two adjacent CDs (denoted as DAD) by applying Formula (5.4) below:

$$DAD = CC_{2.x} - CC_{1.x} \quad (5.4)$$

Step 4 Figure out the thickness of a black border (denoted as TOB).

4.1 Perform a raster scan from point $(CC_{1.x}, 0)$ to point $(CC_{1.x}, CC_{1.y})$.

4.2 Derive the TOB by calculating the black pixels between these two points.

Step 5 Figure out a *detection sample* (denoted as DS) by applying Formula (5.5) below:

$$DS = \frac{DAD - TOB}{2} \quad (5.5)$$

Step 6 Figure out a *detection value* of each CD_i (denoted as DV_i).

6.1 For CD_i in I' , perform a raster scan from point $(CC_{i,x}, CC_{i,y})$ to point $(CC_{i+1,x}, CC_{i+1,y})$.

6.1.1 Calculate the number of pixels on the scanning line until a pixel with the RGB value of $(0, 0, 0)$ is detected.

Step 7 Perform the following data extraction steps.

7.1 For each CD_i in I' , compare each DV_i with DS in the following way.

7.1.1 If $DV_i < DS$, set $DData_i = 1$.

7.1.2 If $DV_i > DS$, set $DData_i = 0$.

In Step 1, the first CD (denoted as CD_1) indicates the CD which is located in the most-west-north position of I' , and the second CD (denoted as CD_2) indicates the right adjacent CD of the CD_1 .

5.3.4 Experimental Results and Discussions

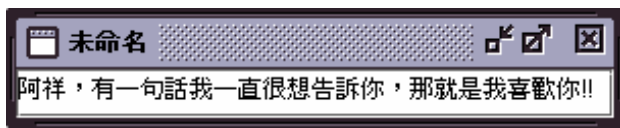
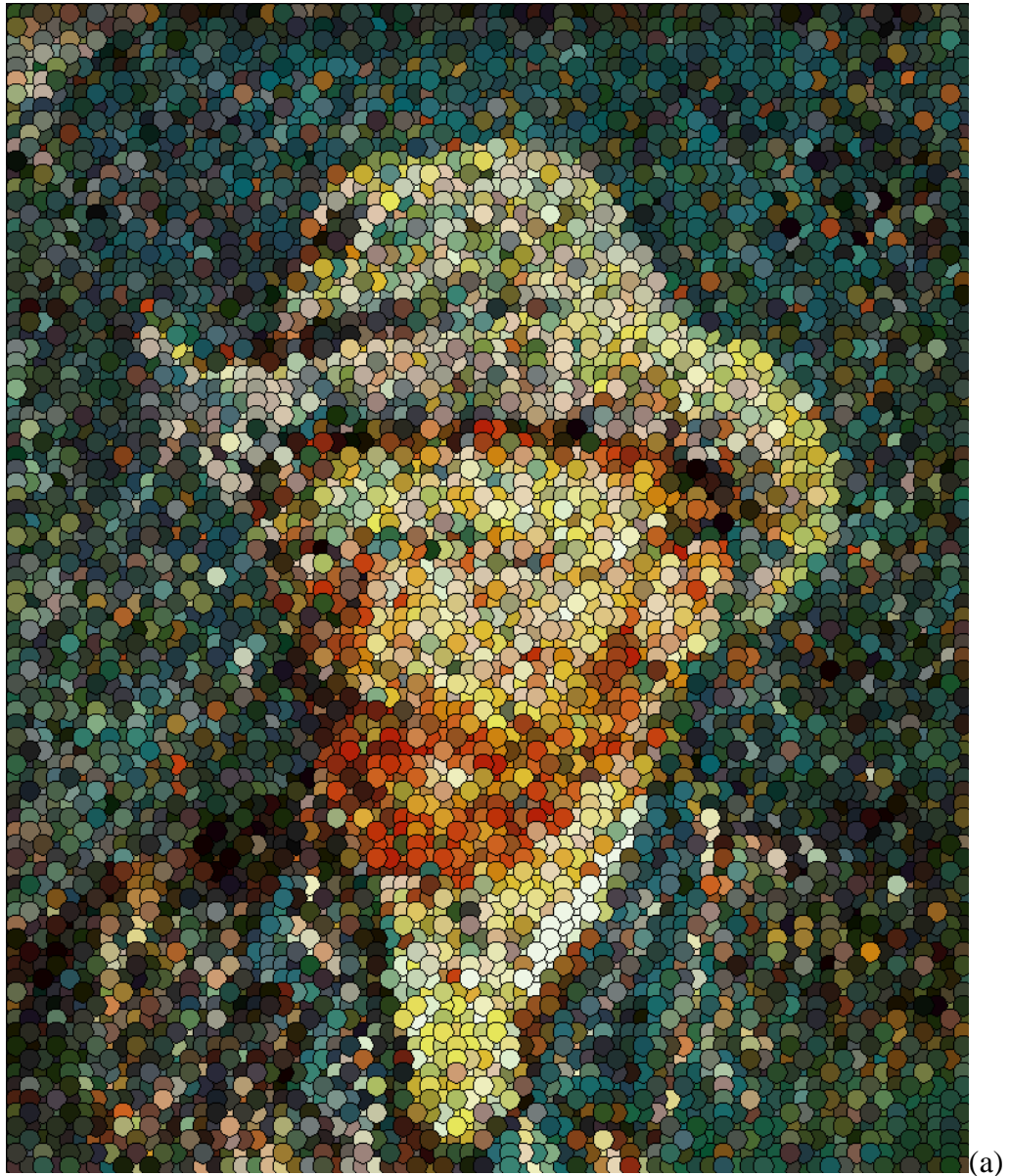
Figure 5.7 shows some experimental results of secret hiding in a digital circular-dotted image. Figure 5.7(a) is a circular-dotted image with the secret message, “阿祥，有一句話我一直很想告訴你，那就是我喜歡你!!,” and the watermark, Figure 5.7(c), embedded. Figure 5.7(b) is a secret message extracted from Figure 5.7(a) with a correct key, and the message is identical to the one we have embedded in Figure 5.7(a). Figure 5.7(d) is the watermark extracted from Figure 5.7(a) with a correct key.

Figure 5.7(e) shows the extraction result of Figure 5.7(a) with a wrong key. We can see that the text shown in Figure 5.7(e) is disordered and meaningless. It proves that the key we applied in the proposed data hiding process really works.

Figure 5.8 show some other experimental results. The secret data embedded in Figure 5.7(a) and Figure 5.8(a) are the same, but the input values (*DAD* and *TOB*) of them are different. It proves that although the sizes and the borders of the circular dots in Figure 5.7(a) and Figure 5.8(a) are different, the data extraction process proposed in this chapter still works.



Figure 5.6 The original image of Figure 5.6(a) and Figure 5.7(a).



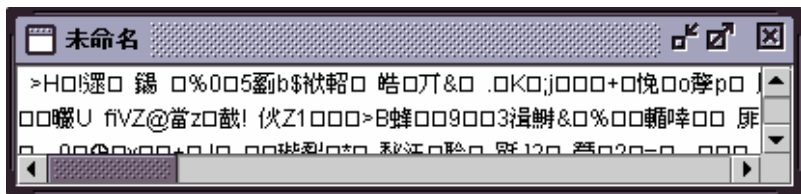
(b)



(c)

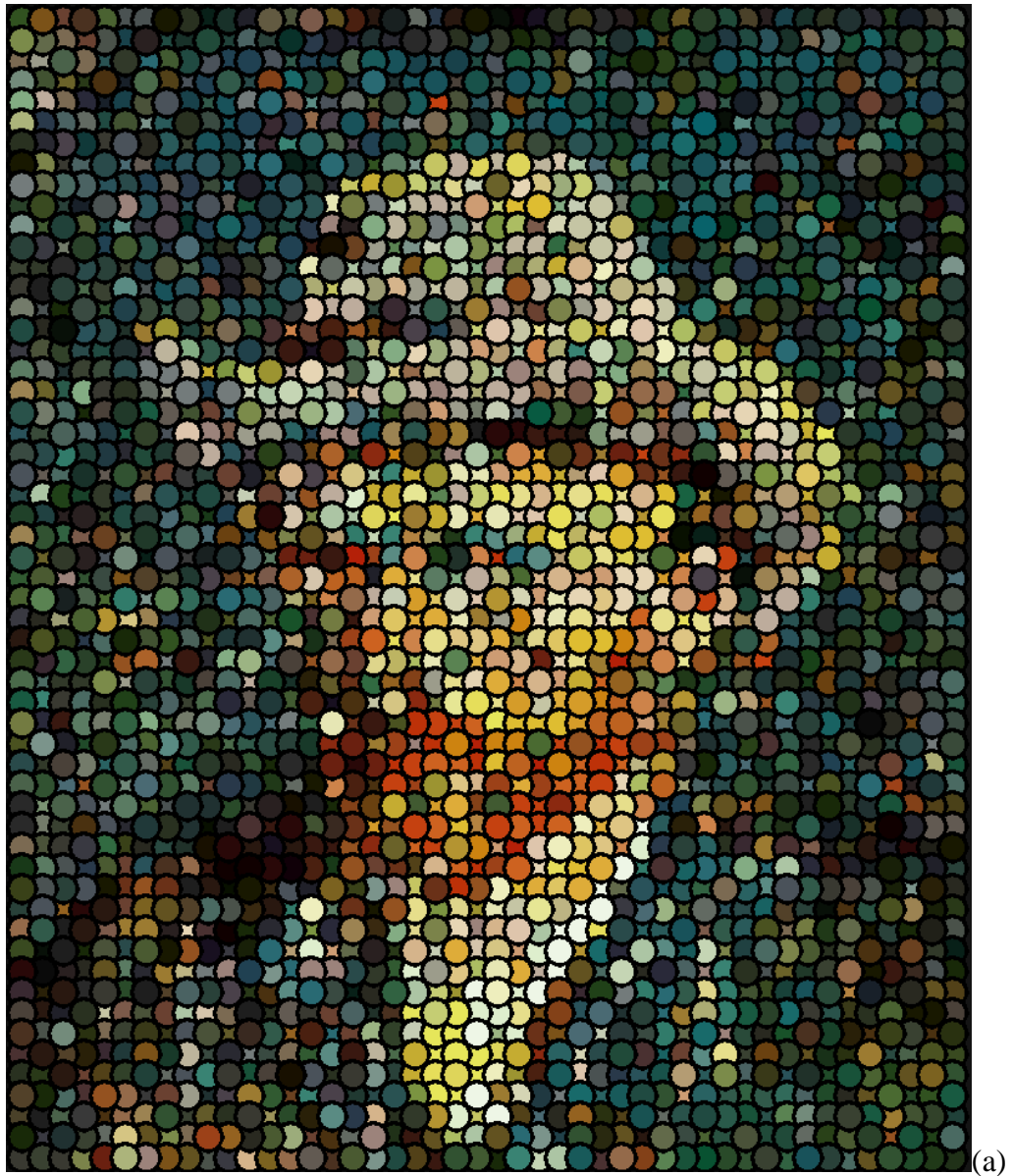


(d)

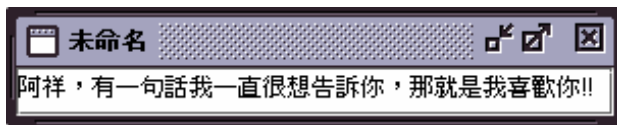


(e)

Figure 5.7 Experimental results of data hiding in a circular-dotted image with inputs $DAD=11$ and $BOT=1$. (a) A digital circular-dotted image with a secret message and the watermark (c) embedded. (b) The secret message extracted from (a) with a correct key. (d) A watermark extracted from (a) with a correct key. (e) The secret message extracted from (a) with an incorrect key.



(a)



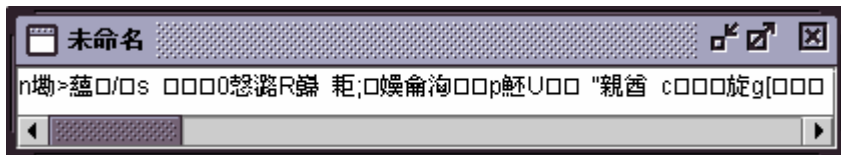
(b)



(c)



(d)



(e)

Figure 5.8 Experimental results of data hiding in a circular-dotted image with inputs $DAD=17$ and $BOT=3$. (a) A digital circular-dotted image with secret message and the watermark, (c), embedded. (b) The secret message extracted from (a) with a correct key. (d) A watermark extracted from (a) with a correct key.

Figure 4.18 shows the performing time of information embedding in and extracting from digital circular-dotted images.

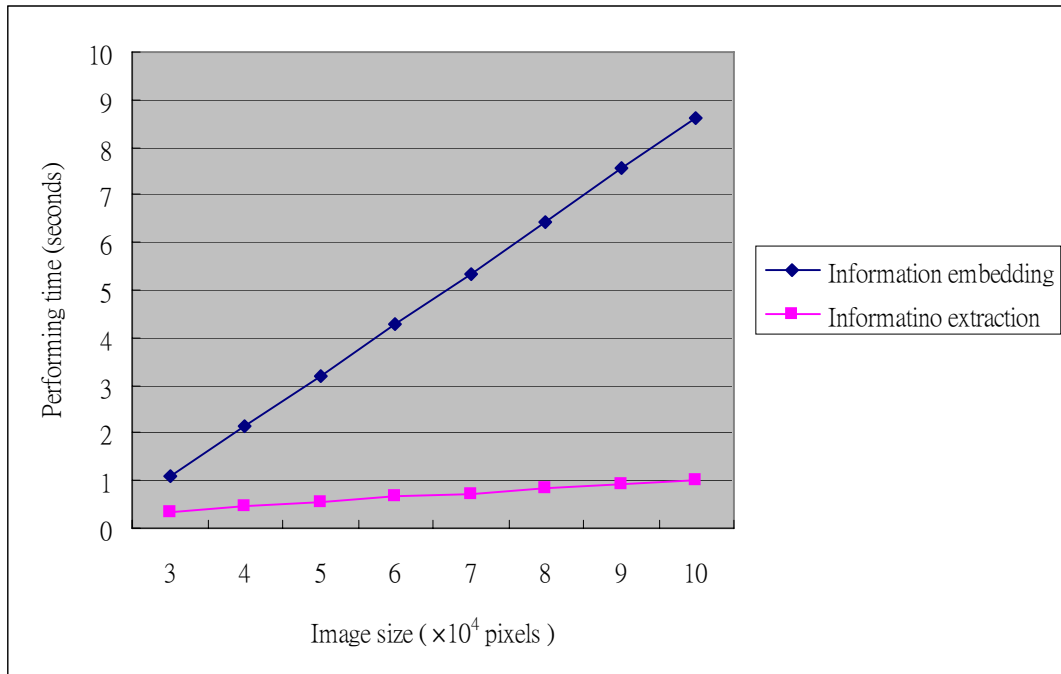


Figure 5.9 The performing time of experimental results.

