

# Evaluation of Visual Cryptography Halftoning Algorithms

Shital B Patel<sup>1</sup>, Dr. Vinod L Desai<sup>2</sup>

<sup>1</sup>Research Scholar , RK University, Kasturbadham, Rajkot, India.

<sup>2</sup>Assistant Professor, Department of Computer Science, Government Science College, Chikhali, Navsari

**Abstract**— Visual Cryptography (VC) encrypts a secret image into  $n$  halftone image shares. The secret image can be recovered simply by stacking these shares together without any complex computation involved. The stacking of less shares on transparencies is unable to extract any information about the secret. Halftoning is the key features of visual cryptography which provides security at the early stage of cryptography. A particular halftoning method that has been used extensively in VC scheme is called error diffusion method. This method provides simple and efficient algorithms for halftoning. In this paper we will make a review of halftoning techniques that are used in VC scheme and evaluate the respective algorithms and its parameters such as PSNR and perceived error. The simulation shows that the visual quality of the obtained halftone shares is observably better than that attained by any available visual cryptography method known to date.

**Keywords:** Visual cryptography, share, stacking, PSNR, perceived error.

## I. INTRODUCTION

Visual Cryptography (VC), proposed by Naor and Shamir<sup>[1]</sup>, is a method for protecting image-based secrets that has a computation-free decryption process. Each secret image is divided into two shares such that no information can be reconstructed from any single share. Each share is printed in transparencies. The decryption process is performed by stacking the two shares and the secret image can be visualized by naked eye without any complex cryptography computations.

In the above basic VC scheme each pixel 'p' of the secret image is encrypted into a pair of sub pixels in each of the two shares. If 'p' is white, one of the two columns under the white pixels in Fig. 1 is selected. If 'p' is black, one of the two columns under the black pixels are selected. The first two pairs of sub pixels in the selected columns are assigned to share 1 and share 2, respectively. Since in each shares, p is encrypted into black and white or white and black pair of sub pixels, an individual share gives no clue about the secret image. By stacking the two shares as shown in the last row of Fig.1.









Secret image	Share1	Share2	Stacked image
			
			

Fig.1 Sharing and Stacking scheme of Black and White Pixel

If 'p' is white it always output one black and one white sub pixel during encryption. If 'p' is black, it outputs two black sub pixels.

An example of the (2,2) – VCS scheme is shown in Fig.2, where the share images are  $2 \times 2$  times larger than the original secret image. The disadvantage of conventional visual secret sharing schemes is that it applied for binary image only.



Fig.2a Original binary image



Fig.2b Share1



Fig.2c Share2



Fig2d:Decrypted image obtained bystacking share1 & share2

The organization of the paper is as follows: Section II gives the discusses of related work on halftoning, Section III define the methodology use for halftoning algorithms , Section IV shows results and finding , finally section V discusses overall conclusion.

## II. RELATED WORK

Halftone images contain a series of dots in a specific pattern that simulate the look of a continuous tone image. Because printers cannot print continuous tones — whether it's the many shades of gray in a grayscale image or the millions of colors in a color photograph — you must convert these images to halftones. Another term for halftoning is dithering. Halftoning is an intentionally apply form of noise called as 'blue noise' used to randomize quantization error. It prevents large-scale patterns such as color banding in images. Halftoning is analogous to the dithering technique used in digital audio and digital video data processing such as digital photography, seismology, RADAR, weather forecasting systems and many more. It is often one of the last stages of audio production to compact disc. In the applications mentioned above, the quantization and re-quantization of digital data results into error. If this error is repeating and correlated to the signal, it results into a cyclical and mathematically determinable form. In some fields, especially where the receptor is sensitive to such artifacts, cyclical errors yield undesirable artifacts. To overcome this drawback a new technique was proposed by Zhou et al [7] called as 'Halftoning'. Halftoning result in less determinable artifacts. In image processing it is always required to print various images with a limited color palette. This results into loss of details of an image. In halftone image, colors not available in the palette are approximated by a diffusion of colored pixels from within the available palette. The human eye perceives this diffusion as a mixture of the colors within it. It has been shown that a blue noise halftone pattern is the least unsightly and distracting pattern [2]. The goal of all half toning technique is to generate an image with fewer amplitude levels that is perceptually similar to the original.

The visual qualities of shares are measured by two numerical quantities:

- a) PSNR
- b) MSE

The PSNR block computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is often used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed or reconstructed image.

The Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR) are the two error metrics used to compare image

compression quality. The MSE represents the cumulative squared error between the compressed and the original image, whereas PSNR represents a measure of the peak error. The lower the value of MSE, the lower the error.

To compute the PSNR, the block first calculates the mean-squared error using the following equation:

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N}$$

In the previous equation, M and N are the number of rows and columns in the input images, respectively. Then the block computes the PSNR using the following equation:

$$PSNR = 10 \log_{10} \left( \frac{R^2}{MSE} \right)$$

In the previous equation, R is the maximum fluctuation in the input image data type. For example, if the input image has a double-precision floating-point data type, then R is 1. If it has an 8-bit unsigned integer data type, R is 255, etc.

## III. METHODOLOGY

The proposed methodology is implemented using MATLAB, there are several algorithms to perform halftoning on images. In this paper we perform evaluation of two algorithms in Matlab and also find results of those algorithms. Halftoning method uses a two phased algorithms.

### A. Floyd-Steinberg halftoning algorithm

The Floyd-Steinberg dithering algorithm is based on error dispersion. The error dispersion technique is very simple to describe: for each point in the image, first find the closest color available. Calculate the difference between the value in the image and the color you have. Now divide up these error values and distribute them over the neighboring pixels which you have not visited yet. When you get to these later pixels, just add the errors distributed from the earlier ones, clip the values to the allowed range if needed, then continue as above.

If you are dithering a gray scale image for output to a black-and-white device, the "find closest color" is just a simple thresholding operation [6]. In color, it involves matching the input color to the closest available hardware color, which can be difficult depending on the hardware palette. There are many ways to distribute the errors and many ways to scan the image, but I will deal here with only a few. The two basic ways to scan the image are with a normal left-to-right, top-to-bottom raster, or with an alternating left-to-right then right-to-left raster. The latter method generally produces fewer artifacts and can be used with all the

error diffusion patterns discussed below. The different ways of dividing up the error can be expressed as patterns (called filters, for reasons too boring to go into here).

X 7 This is the Floyd and Steinberg  
3 5 1 error diffusion filter.

In this filter, the X represents the pixel you are currently scanning, and the numbers (called weights, for equally boring reasons) represent the proportion of the error distributed to the pixel in that position. Here, the pixel immediately to the right gets 7/16 of the error (the divisor is 16 because the weights add to 16), the pixel directly below gets 5/16 of the error, and the diagonally adjacent pixels get 3/16 and 1/16. When scanning a line right-to-left, this pattern is reversed. This pattern was chosen carefully so that it would produce a checkerboard pattern in areas with intensity of 1/2 128 in our image). It is also fairly easy to calculate when the division by 16 is replaced by shifts<sup>1</sup>. The matrix shown graphically in Figure 3(b) is an error-diffusion matrix proposed by Floyd and Steinberg [2][12].

**Algorithm 1.**

1. Procedure FLOYD-STEINBERG HALFTONING AN IMAGE
2. for i = 1 ,.....,n do
3. for j = 1 ,.....,m do
4. if  $j(i, j) < 128$  is found then  $j(i, j) = 0$  7.
5. else  $j(i, j) = 1$
6. error =  $j[i, j] - i[i, j] * 255$  8.
7. Distributed (3/8) error to the right pixel
8. Distributed (1/8) error to the right diagonal pixel
9. Distributed (1/8) error to the bottom pixel
10. Distributed (3/8) error to the left diagonal pixel
11. end for
12. end for
13. End procedure

AS observed from figure 3(b), halftoning is a very time consuming process.

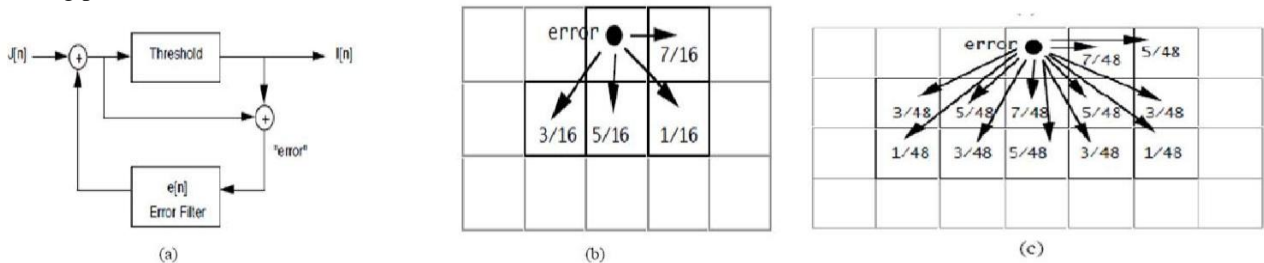


Fig.3. a) The Floyd-Steinberg Haftoning , b) Floyd-Steinberg Error-Diffusion Matrix, c) Jarvis Error-Diffusion Matrix

**B. Jarvis halftoning algorithm**

Another error diffusion algorithm has been proposed by Jarvis, Judice and Ninke. It diffuses the error in the 12 neighboring cells instead of 4 cell as in the Floyd-Steinberg algorithm. As a result, this algorithm is even slower, requiring at least 24\*n\*m floating point and memory access operations. Further, when printing images, the running time increase by a factor of four. A diffusion matrix of Jarvis algorithm is shown in figure 3(c) [11].

**Algorithm 2.**

1. Procedure JARVIS HALFTONING AN IMAGE
2. for i = 1 ,.....,n do
3. for j = 1 ,.....,m do (This algorithm goes through all pixels in the original image, normally starting from the pixel up to the left and then goes through all pixels from left to right and up down).
4. if  $f(i, j) < 128$  THEN
  - $b[i, j] = 1$
  - else  $b[i, j] = 0$
5. Since the pixel value in f, which is a real number between 0 and 255, has been replaced by 0 or 1 in b and "error" has been calculated.
  - $e = f(i, j) - b(i, j)$
  - The "error" is the difference between the pixel value in f and b at that position.
6. The error occurred at the position (i, j) is weighted by 7/48 and added to the pixel value at (i+1, j). The same error is weighted by 5/48 and added to the pixel at (i+1, j+1) and so on.
  - 7. end for
  - 8. end for
  - 9. End procedure

$$\tilde{f}(i, j) = f(i, j) + \sum_{k,l \in S} h(k, l)e(i - k, j - l)$$

After the error has been diffused the pixel value of the next position is compared to the threshold and the same process continues until all pixels have been met.



IV. RESULTS AND FINDING

The Floyd-Steinberg Halftoning and The Jarvis Halftoning algorithm are discussed above and also implemented using MATLAB R2012a and test performance of these algorithms and evaluate the result.

The result of Floyd-Steinberg halftoning algorithm applies in MATLAB is given below fig.4.

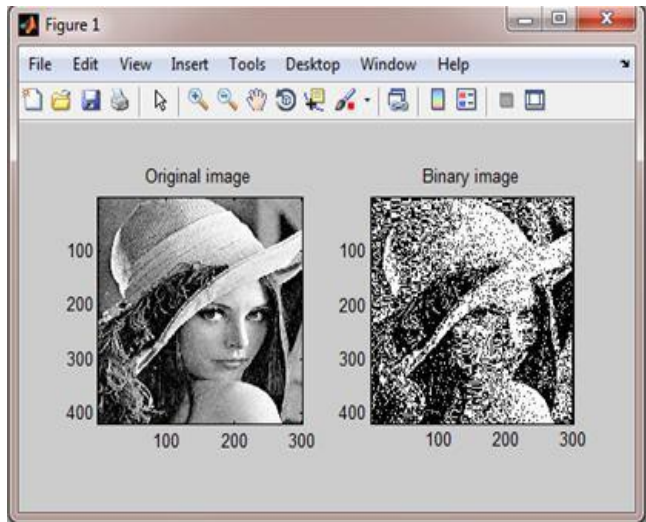


Fig.4 Original image with Floyd –Steinberg halftoning

Next, the result of Jarvis halftoning algorithm applies also in MATLAB is given below fig.5(a) and (b).

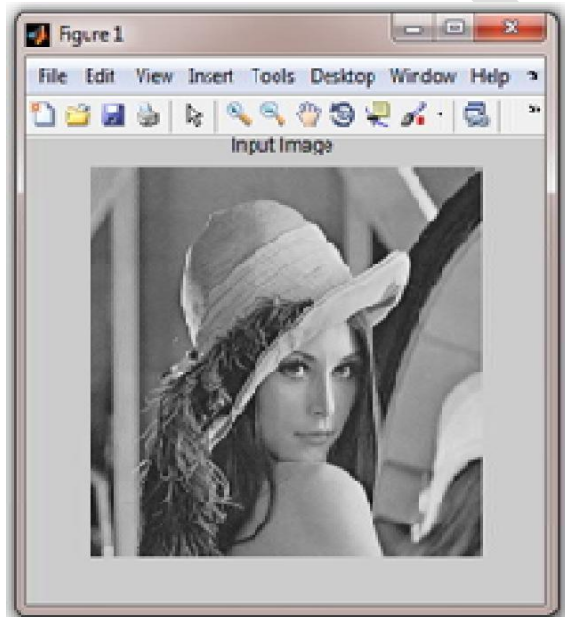


Fig.5 a) Original image (Input image)



Fig.5 b) Jarvis Halftone image

In this section, we will discuss the experimental results of the halftoning algorithms used in visual cryptography. Figure 4 and Figure 5 shows the result of halftone images, where we can observe the quality of all the images.

We can find that the visual quality of halftone images depends on the algorithm employed. Jarvis halftone method produce clearer image compared to Floyd and Steinberg halftone method. A typical problem that is seen in this Floyd-Steinberg halftoning technique is spectral whitening where the variation in average separation distance between minority pixels becomes so great that the pattern starts to resemble the halftone pattern created by white noise. Comparing the simulation results of both the algorithms, we can observe that increasing the coefficients that are distributed to the neighboring pixels , the contrast loss can be reduced.

The PSNR and Perceived values for these two algorithms are listed in below table 1.

Type of Halftoning Algorithm	Perceived Error	PSNR
Floyd-Steinberg algorithm	4213223	66.1038
Jarvis algorithm	4206273	67.4695

TABLE 1: PERCEIVED ERROR AND PSNR FOR TWO ALGORITHMS.

CONCLUSION

In this paper, two algorithms for halftoning are compared. The comparison is done on the basis of contract loss , perceived error between original and halftone image and the PSNR values . From the implementation of these algorithms , it observe that , visual quality of halftone image is higher

when Jarvis algorithm is used and also observe that if the error is diffused in larger areas it gives sharper details and reduces some of the artifacts.

So, in future as the further work it can be extended to implementing the faster algorithms with high visual quality.

#### REFERENCES

- [1]. M. Naor and A. Shamir (1994). "Visual Cryptography" In Proc. Eurocrypt, 1-12.
- [2]. Steinberg, R. W. (1976). "An Adaptive Algorithm For Spatial Grayscale". *17* (2), 75-77.
- [3]. B.Preneel, V. (1996). Efficient Color Visual encryption for shared colors of benetton. *Eurocrypto'96,Rump session*.
- [4]. G.Ateniese, C. (1996). Visual Cryptography for general access structure. *Information and Computation, Vol. 129*, 86-106.
- [5]. C.S.Laih, C. (2000). New Colored Visual Secret Sharing Scheme. *Design, Codes and Cryptography, Vol.20, No.3*, 325-335.
- [6]. Chang-Chou Lin, W.-H. T. (July 2002). Visual Cryptography for grey-level images by dithering techniques. *National Chiao Tung University*.
- [7]. Z.Zhou, G. a. (2006, Aug.). "Halftone Visual Cryptography". *18*, 2441-2453.
- [8]. Tzung-Her Chen, K.-H. T.-C. (2008). Multi-Secrets Visual Secret Sharing. *Proceeding of APCC*.
- [9]. Jena, D. &. (2009). A Novel Visual Cryptography Scheme. *In Advanced Computer Control, ICACC'09. International Conference on IEEE*, pp. 207-211.
- [10]. S.Chandramathi. (2010). Retrieved from [www.bioinfo.in/uploadfiles/13255803791\\_1\\_6\\_IJCI.pdf](http://www.bioinfo.in/uploadfiles/13255803791_1_6_IJCI.pdf).
- [11]. Patil, P. &. (2012, July). "VISUAL CRYPTOGRAPHY for COLOR IMAGES using ERROR DIFFUSION and PIXEL SYNCHRONIZATION."
- [12]. Ulichney, R. A. (2012). "The Void-And-Cluster Method for Dither Array Generation". 07-19.
- [13]. JKaur, N. &. (2013, October). "A Survey on Embedded Extended Visual Cryptography Scheme.". *International Journal of Engineering Sciences & Research Technology*, 2(10)