Histogram Uniformization for Digital Image Encryption

Juliano B. Lima
Department of Mathematics
Federal University of Pernambuco
Recife, Brazil
juliano@dmat.ufpe.br

Ricardo M. Campello de Souza
Department of Electronics and Systems
Federal University of Pernambuco
Recife, Brazil
ricardo@ufpe.br

***Abstract*—In this paper, histogram uniformization of digital images by means of the finite field cosine transform (FFCT) is examined. The approach consists in dividing the image into blocks and applying the FFCT, in a recursive manner, to each block. Simulations of the procedure show that the histogram of the transformed image exhibits a uniform shape and its pixels have low correlation with their neighbors. This result is achieved due to the modular arithmetic used in the application of the FFCT to the image blocks. The suitability of the proposed technique in the context of image encryption is discussed.**

***Keywords*-image encryption; histogram uniformization; finite field cosine transform.**

I. Introduction

Nowadays, the distribution of multimedia content on the Internet and other communication networks became a practice usually performed by users with different profiles. In this scenario, techniques devoted to protect this kind of information play an important role, providing confidential transmission and assuring the integrity of the received data. These are some of the reasons why the interest in studying watermarking, steganography and encryption for digital image, video and audio, has increased over the years [1], [2], [3].

Steganographic techniques have the main purpose of hiding a relevant information (a text or an image, for example) *behind* an apparently unimportant image. In a practical steganographic technique, an unauthorized person should not be able to realize the presence of any hidden information [4]. A digital watermark is a kind of fingerprint introduced without changing visual and statistical aspects of an image. Watermarking has application in scenarios where an information can be maliciously modified by an eavesdropper. The authorized recipient should be able to verify the presence of the referred fingerprint, ratifying the origin of the image (the copyright holder, for example) and/or determining the type of modification it may have suffered [5].

The present paper lies in the context of image encryption, where an image – the *plaintext* – is directly modified and transformed in another image – the *ciphertext* – with aspect completely different from the former, under some cryptographic assumptions [6]. A complete image encryption scheme requires the use of a key and the investigation of several security criteria. In this paper, we focus on a specific feature required by image encryption schemes: the resistance against statistical attacks. This is done by the introduction of a technique to uniformize histograms of digital images. The proposed approach causes a visual effect similar to pixel scrambling [7]; it should be included as a preliminary step of an image encryption system, with the main purpose of *hiding* the histogram of the original image and, therefore, avoiding the effectiveness of the mentioned attacks.

The uniformization technique proposed in this work is based on the finite field cosine transform (FFCT), the application of which requires operations involving modular arithmetic only [8], [9]. This means that rounding is not necessary and efficient computations can be carried out; once the values of the pixels of the transformed images are integer numbers, their coding and, consequently, their transmission or storage are easier.

After this introduction, in Section II, theoretical aspects of the FFCT are reviewed. In Section III, the proposed histogram uniformization technique is presented and metrics used to evaluate its performance are considered. In Section IV, we present simulations for the proposed technique and analyze its results; several points of the procedure are discussed and comparisons with other techniques are made. Also in this section, we give suggestions to extend the approach to a complete image encryption scheme. The paper closes with some concluding remarks on Section V.

II. Finite Field Cosine Transforms

In this section, the main theoretical aspects related to finite field cosine transforms are reviewed. We start by presenting some definitions concerning trigonometry in finite fields [10]. The most important results related to the eigenstructure of such transforms are also presented [11].

The FFCT was originally introduced in [8]. Actually, there are 8 types of finite field cosine transforms and also 8 types of finite field sine transforms. They constitute the family of finite field trigonometric transforms (FFTT), which is completely described in [9]. Such transforms employ the following definition related to trigonometry in finite fields.

Definition 1: Let ζ be a nonzero element in the finite field $\mathrm{GF}(p)$, p an odd prime. The finite field cosine function related to ζ is computed modulo p by

$$\cos_\zeta(x) := \frac{\zeta^x + \zeta^{-x}}{2}, \tag{1}$$

$x = 0, 1, \ldots, \mathrm{ord}(\zeta)$, where $\mathrm{ord}(\zeta)$ denotes the multiplicative order of ζ.

The definition of the finite field cosine function could contain additional details, such as the possibility of having the number ζ in the field $\mathrm{GI}(p)$, the integer Gaussians set modulo p, and the condition $p \equiv 3 (\mathrm{mod}\ 4)$ [8], [11]. It is remarkable to observe that the finite field cosine hold properties similar to those of the standard real-valued cosine function, such as *unit circle* and *addition of arcs*, for instance.

The finite field cosine transform adopted in this work is the FFCT of type 2. The FFCT of a vector with elements in a finite field is computed according to the following definition.

Definition 2: Let $\zeta \in \mathrm{GF}(p)$ be an element with multiplicative order $2N$. The finite field cosine transform of the vector $\mathbf{x} = [x_0, x_1, \ldots, x_{N-1}]$, $x_i \in \mathrm{GF}(p)$, is the vector $\mathbf{X} = [X_0, X_1, \ldots, X_{N-1}]$, $X_k \in \mathrm{GF}(p)$, of elements

$$X_k := \sqrt{\frac{2}{N}} \sum_{i=0}^{N-1} \beta_k x_i \cos_\zeta \left[k \left(i + \frac{1}{2} \right) \right], \tag{2}$$

where

$$\beta_r = \begin{cases} 1/\sqrt{2}, & r = 0, \\ 1, & r = 1, 2, \ldots, N-1. \end{cases}$$

It can be shown that the FFCT given in Definition 2 is invertible by the formula [8], [9]

$$x_i := \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} \beta_i X_k \cos_\zeta \left[\left(k + \frac{1}{2} \right) i \right]. \tag{3}$$

The computation of the FFCT of a vector $\mathbf{x}$ can be represented by the matrix equation

$$\mathbf{X} = \mathbf{C}\mathbf{x},$$

where $\mathbf{C}$ corresponds to the transform matrix, the elements of which are obtained directly from Equation (2). By comparing Equations (2) and (3), we verify that $\mathbf{C}\mathbf{C}^t = \mathbf{C}^t\mathbf{C} = \mathbf{I}$, where $\mathbf{C}^t$ and $\mathbf{I}$ denote, respectively, the matrix $\mathbf{C}$ transpose and the identity matrix. This means that the inverse FFCT is obtained by using the transform matrix $\mathbf{C}^t$ [11]. Using such a matrix notation, the FFCT can be extended to two dimensions; the transform of a matrix $\mathbf{m}$ with dimensions $N \times N$, that is, the two-dimensional version of the FFCT can be computed by

$$\mathbf{M} = \mathbf{C}\mathbf{m}\mathbf{C}.$$

The *period* of the FFCT matrix [11] is of great importance for the application described in this paper, since it requires the recursive computation of the FFCT, as described in the next section. Such a parameter corresponds to the least integer and positive power l giving $\mathbf{C}^l = \mathbf{I}$. The period is the multiplicative order of the matrix $\mathbf{C}$ while an element of the group $\mathrm{GL}(N, \mathrm{GF}(p))$. Differently from other transforms such as, for example, the discrete Fourier transform, whose transform matrix always has period equals to 4, the FFCT matrix defined in Equation (2) apparently does not present any regularity in its period [11].

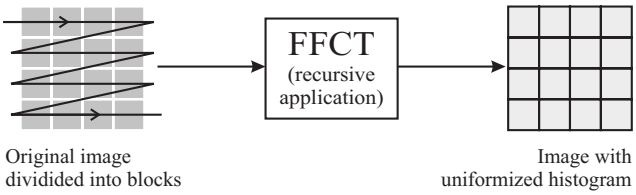


Figure 1: Schematic diagram for image histogram uniformization.

In order to investigate this question, we write the matrix $\mathbf{C}$ as $\mathbf{C} = \mathbf{U}\Lambda\mathbf{U}^t$, where $\mathbf{U}$ is a unitary matrix, the columns of which are the eigenvectors of $\mathbf{C}$, and Λ is a diagonal matrix, the elements of which are the eigenvalues of $\mathbf{C}$. Since $\mathbf{U}\mathbf{U}^t = \mathbf{I}$, then $\mathbf{C}^l = \mathbf{U}\Lambda^l\mathbf{U}^t$, where Λ^l is obtained by computing the l-th power of each element in the main diagonal of Λ. Therefore, l is the period of $\mathbf{C}$, if l is the least common multiple of the multiplicative orders of the eigenvalues of $\mathbf{C}$. It is conjectured that the eigenvalues of the matrix $\mathbf{C}$ are all distinct; they also can lie in extension fields [11]. This means that it is possible to obtain FFCT of type 2 matrices with larger periods[1], compared to the periods of other transforms. FFCT of types 1 and 4, for example, have period $l = 2$, i.e., such transforms are involutions [9].

III. Histogram Uniformization Based on the FFCT

In this section, the procedure for uniformizing the histogram of a digital image by using the finite field cosine transform is described. The basic idea is to divide the original image into blocks of fixed size and take such blocks sequentially, from left to right and from top to bottom, and applying the two-dimensional FFCT a certain number of times. The number of times in which the transform is recursively applied depends on the parameters of the FFCT being used. This procedure is illustrated in Figure 1. The original image is recovered by applying the inverse FFCT to the transformed image.

The description of the effects of the FFCT application on the histogram of an image can be made by considering some aspects of modular arithmetic. If each pixel of a gray scale (8 bits) image is multiplied modulo p by a constant K, the result is a displacement of the frequency of occurrence of this image symbols (integer numbers from 0 to 255).

[1]The existence of repeated eigenvalues would reduce the FFCT matrix period.

If $K = 2$ and $p = 257$, for example, after the modular multiplication, the frequency of occurrence of the symbol 200 will be associated to the symbol 143.

When the product between each symbol and a constant K is substituted by a linear combination which involves a block of symbols (transform application), the process becomes more complex. However, the aspect observed in the situation presented as an example is also verified. Although the difference among the frequency of occurrence of the symbols is large, the tendency is that the application of the FFCT produces transformed blocks composed by uniformly distributed symbols [12].

The use of finite field tools provide advantages with respect to computational complexity and accuracy, since rounding or truncation are not necessary [13]. In this context, the finite field $\mathrm{GF}(p)$, where p is a Fermat prime, i.e., a prime of the form $p = 2^m + 1$, is a structure where the algorithm for the computation of transforms are simplest [14].

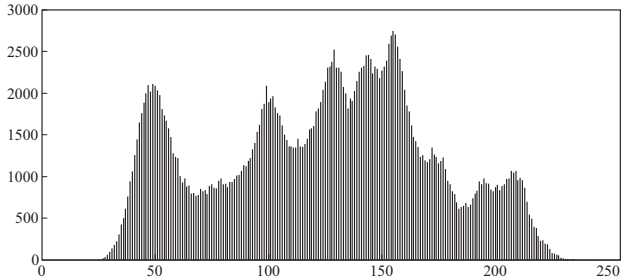
In this paper, gray scale images are considered and, therefore, their processing requires a transform such that $p \geq 257$. However, the greater the prime number used to define the transform is, more bits are needed to represent the image after applying the transform. In order to avoid this fact, the proposed technique uses the Fermat prime $p = 257$ to define an FFCT with $N = 8$; the number $\zeta = 128$, with $\mathrm{ord}(\zeta) = 16$, is chosen and the transform matrix is, from Equation (2),

$$\mathbf{C} = \begin{bmatrix} 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 \\ 137 & 163 & 98 & 106 & 151 & 159 & 94 & 120 \\ 160 & 6 & 251 & 97 & 97 & 251 & 6 & 160 \\ 163 & 151 & 120 & 159 & 98 & 137 & 106 & 94 \\ 242 & 15 & 15 & 242 & 242 & 15 & 15 & 242 \\ 98 & 120 & 106 & 163 & 94 & 151 & 137 & 159 \\ 6 & 97 & 160 & 251 & 251 & 160 & 97 & 6 \\ 106 & 159 & 163 & 120 & 137 & 94 & 98 & 151 \end{bmatrix}. \quad (4)$$

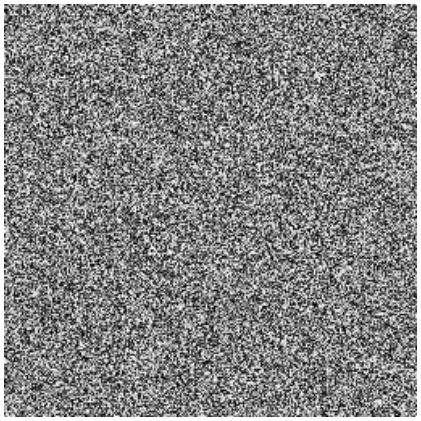
Powers of the above transform matrix were calculated using a computer program and the period $l = 16,974,594$ was encountered for such a matrix.

In Figure 2, the image *lena.bmp*, with dimensions 512×512 pixels, and its original histogram are presented; additionally, the image after the uniformization process illustrated in Figure 1, with each block transformed only once by the matrix in Equation (4), and its respective histogram are presented. In this figure, we verify that, after the application of the transform, the visual information of the image is completely damaged and a noisy aspect is observed. With respect to the new histogram, we observe that its distribution appears to be uniform, which suggests that a statistical analysis would not be effective for the evaluation of the original image content. It is important to emphasize that this illustrative result was obtained by applying the FFCT only once to each image block, which indicates that, differently

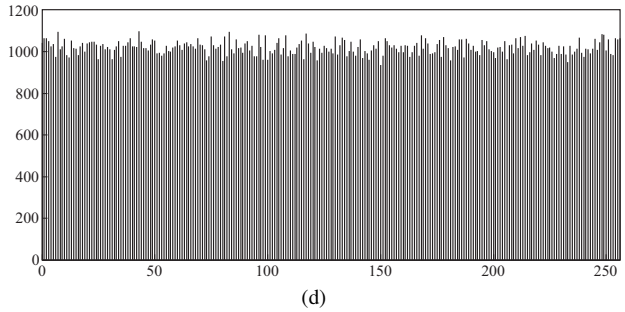
(a)



(b)



(c)



(d)

Figure 2: Example of FFCT-based histogram uniformization. (a) Original image *lena.bmp*; (b) *lena.bmp* histogram; (c) Image with uniformized histogram; (d) Uniformized histogram.

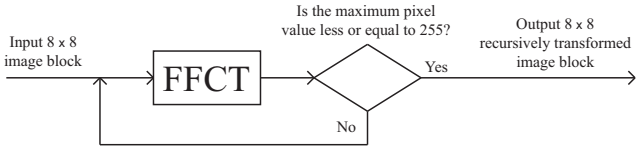


Figure 3: Recursive computation of the finite field cosine transform of an 8×8 image block.

from other techniques, our approach does not requires a large number of iterations [7].

A. Recursive application of the FFCT

As we have mentioned, the uniformization of the histogram of a gray scale image is done by using an FFCT with $p = 257$, such as that whose transform matrix is given by Equation (4). Although the application of this FFCT to each image block only once produces, from the point of view of histogram uniformization, results such as those presented in Figure 2, it brings implications related to image coding. In particular, the transformed image may contain pixels with values equals to 256, which are not adequate to an 8-bit coding.

In order to avoid this limitation, we apply the FFCT recursively to each image block until the resulting block has no pixels with value equals to 256. Such a criterion for the recursive application of the FFCT is illustrated in Figure 3. This process is invertible by computing the inverse transform also repeatedly until encountering the original image block, which does not contain any pixel with value equals 256.

Since there is no superposition among image blocks, the recursive application of the FFCT to each block can be done in parallel. Independently of this fact, such an operation represents an additional computational load to the proposed histogram uniformization technique. In order to evaluate the effect of this procedure, in Section IV, we count the number of blocks of a test image which need a determined number of FFCT applications until the resulting block has no pixels with value equals to 256. Observe that, if the number of recursive transform applications required by an image block (according to the criterion illustrated in Figure 3) is larger than the period of the transform matrix, the procedure fails. This ratifies the importance of using a transform with period as large as possible in the uniformization process.

B. Metrics to evaluate the proposed technique

In order to evaluate the performance of the proposed uniformization technique, we consider the metrics described as follows. The first metric is the *gray difference degree* [7], which requires, initially, the computation of the gray difference of a pixel with a neighbor pixel (GN). This parameter is given by

$$GN = \frac{\sum[G(x,y) - G(x',y')]^2}{4}, \tag{5}$$

where

$$(x',y') = \begin{cases} (x-1,y) \\ (x+1,y) \\ (x,y-1) \\ (x,y+1) \end{cases} \tag{6}$$

and $G(x,y)$ denotes the value of the pixel in position (x,y). The average neighborhood gray difference of the whole image (dimensions $M \times N$ pixels) is

$$AN(GN(x,y)) = \frac{\sum_{x=2}^{M-1}\sum_{y=2}^{N-1} GN(x,y)}{(M-2)\times(N-2)}. \tag{7}$$

The gray difference degree is then defined by

$$GVD = \frac{AN'(GN(x,y)) - AN(GN(x,y))}{AN'(GN(x,y)) + AN(GN(x,y))}, \tag{8}$$

where AN and AN' denote, respectively, the average neighborhood gray difference of the image before and after the histogram uniformization. The closer to 1 the value of *GVD*, the better the effect of the uniformization procedure.

Another metric we consider is the *correlation between two adjacent pixels* of an image (the adjacency can be horizontal, vertical or diagonal) [7]. By selecting randomly P pixels of the image, the correlation coefficient is computed by

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)D(y)}}, \tag{9}$$

where $cov(x,y) = \frac{1}{P}\sum_{i=1}^{P}(x_i - E(x))(y_i - E(y))$, $D(x) = \frac{1}{P}\sum_{i=1}^{P}(x_i - E(x))^2$ and $E(x) = \frac{1}{P}\sum_{i=1}^{P} x_i$; x_i is the value of the i-th selected pixel and y_i is the value of the correspondent adjacent pixel. It is expected that an image, before being submitted to the histogram uniformization, has correlation coefficient close to 1; it is desirable that the correlation coefficient of the image with uniformized histogram be as close to 0 as possible.

Finally, we can compute the entropy of an image after the histogram uniformization [15]. The entropy $H(m)$ of a message source m can be measured by

$$H(m) = \sum_{i=0}^{M-1} p(m_i)\log_2\frac{1}{p(m_i)}, \tag{10}$$

where M is the total number of symbols $m_i \in m$ and $p(m_i)$ represents the probability of occurrence of symbol m_i. For a random source emitting 256 symbols, one has $H(m) = 8$ bits.

IV. Simulations and Discussion

In this section, we describe the simulations carried out to evaluate the proposed method and discuss their results. We also suggest some strategies to extend the method and use it as the first step of an image encryption scheme.

Table I: Percentages of blocks of test images submitted to recursive applications of the FFCT, in the histogram uniformization procedure.

# App.	*peppers*	*mandril*	*lake*	*jetplane*	*lena*	*camera*
1	78.1494	78.2715	78.3447	77.7344	77.0264	78.1738
2	17.5293	17.3096	16.7236	17.2607	17.4561	17.3340
3	3.4180	3.3447	3.9307	3.9795	3.9795	3.6133
4	0.6348	0.7812	0.8301	0.8057	1.2939	0.8057
5	0.1953	0.2197	0.1221	0.1465	0.2197	0.0488
6	0.0488	0.0488	0.0488	0.0732	0.0244	0.0244
7	0.0000	0.0244	0.0000	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
9	0.0244	0.0000	0.0000	0.0000	0.0000	0.0000

Table II: Gray difference degrees (*GVD*) and correlation coefficients of the original test images (r_{xy}) and the test images with uniformized histogram ($\tilde{r}_{xy}$); (v), (h) and (d) are related to vertical, horizontal and diagonal correlation respectively.

Metric	*peppers*	*mandril*	*lake*	*jetplane*	*lena*	*camera*
GVD	0.9794	0.9563	0.9638	0.9779	0.9821	0.9813
$r_{xy}(h)$	0.9801	0.9134	0.9772	0.9697	0.9851	0.9898
$\tilde{r}_{xy}(h)$	0.0024	-0.0049	0.0010	-0.0009	0.0031	-0.0029
$r_{xy}(v)$	0.9798	0.9335	0.9764	0.9723	0.9710	0.9826
$\tilde{r}_{xy}(v)$	-0.0013	0.0015	0.0021	-0.0015	0.0046	0.0024
$r_{xy}(d)$	0.9702	0.8664	0.9626	0.9480	0.9584	0.9725
$\tilde{r}_{xy}(d)$	-0.0096	-0.0045	0.0026	0.0025	-0.0002	0.0018

A. Computer Simulations

The experiments run in Matlab®, where programs to implement the histogram uniformization technique, presented in Section III, and to compute the metrics, presented in Section III-B, were developed. In our tests, we consider the images presented in the first row of Figure 4. All images have dimensions 512×512 pixels and are in gray scale. The transform matrix employed is given by Equation (4). In case the image size has not an exact division by blocks zeroes can be appended.

The proposed histogram uniformization procedure was applied to each test image, which has 4096 image blocks with dimensions 8×8 pixels. The results are shown in the second row of Figure 4. For all images, the visual effects are similar to those observed in Figure 2; a *noisy* aspect is observed in the transformed image tests. In Figure 5, we present the histograms of the original and the transformed test images. We observe that the modified distributions appear to be uniform, which suggests that a statistical analysis would not be effective for the evaluation of the original image content.

In our simulations, we also compute the percentage of image blocks submitted to a given number of recursive applications of the FFCT. Such results are shown in Table I. In all images, almost 80% of the blocks had to be transformed only once. A rather small percentage of blocks had to be transformed more than twice. This indicates that the extra computational effort due to the recursive application of the FFCT is unrepresentative. We also observe that the largest number of rounds necessary for a block was 9. Once the transform matrix we have used has a large period, there is not any *risk* of returning the transformed block to its original block by the FFCT recursive computation.

In Table II, the values for *GVD* and vertical, horizontal and diagonal correlation coefficients for each test image are presented. We can see that the application of the FFCT to the images produces values for *GVD* considerably close to 1. The results we have obtained are better than those achieved by the well known Arnold method, which scrambling the pixels only and produces *GVD* around $0{,}9$ (this value may decay significantly, according to the number of iterations) [7], [16]. Also in [7], an encryption method based on chaotic sequences generated by secret keys is proposed. The *GVD* achieved by our technique are similar to those ones of that work. However, our approach involves less complexity (depending on the image size, the generation and the application of a chaotic sequence for scrambling may require a significant processing time).

The correlation coefficient of an image before and after the histogram uniformization are respectively denoted by r_{xy} and $\tilde{r}_{xy}$; $P = 4000$ pixels were selected by using a generator of uniformly distributed random numbers. As expected, while r_{xy} is considerably close to 1 for all images, $\tilde{r}_{xy}$ is close to 0. This indicates the low level of correlation between two adjacent pixels in the transformed images. Considering the correlation coefficients we have obtained, the proposed method has produced results that are better than those of the Arnold method and similar to those presented in [7]. With the purpose of illustrating this result, correlation distributions of original and transformed image *lena.bmp* are shown in Figure 6. In this figure, randomly chosen pixels and corresponding horizontal adjacent pixels are considered. Similar distributions are obtained for other images and directions in pixel adjacency.

For all images used in our simulations, after the uniformization procedure, the entropy has assumed values varying from 7.9992 to 7.9994. This means that the transformed images are close to a random source and the proposed technique is also secure against the entropy attack [17].

B. Extending the method

We have left clear that the proposed histogram uniformization procedure solely does not provide the encryption of an image. Although visual and statistical characteristics of the transformed images have a *random* aspect, there is not any key involved in the process and some important cryptographic features, such as confusion, are absent [18]. In order to implement a complete image encryption scheme, an extension of our method is considered.

Our suggestion is to use the histogram uniformization procedure as a first step and introduce a second processing

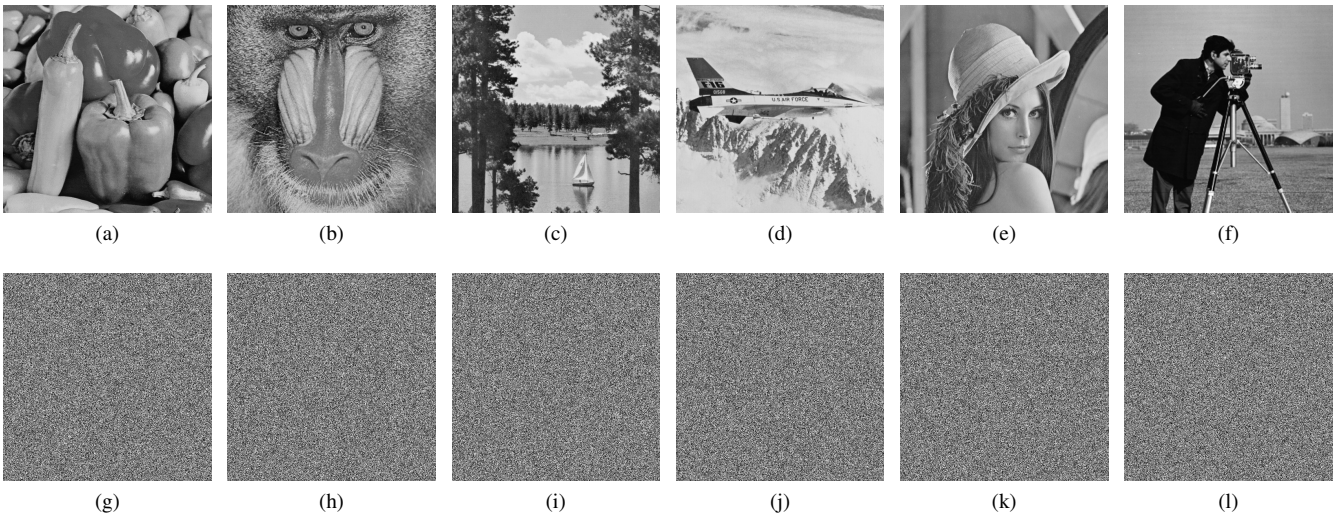


Figure 4: Original and uniformized gray scale images used in the simulations. (a),(g) *peppers.bmp*; (b),(h) *mandril.bmp*; (c),(i) *lake.bmp*; (d),(j) *jetplane.bmp*; (e),(k) *lena.bmp*; (f),(l) *camera.bmp*. All images have dimensions 512×512 pixels.

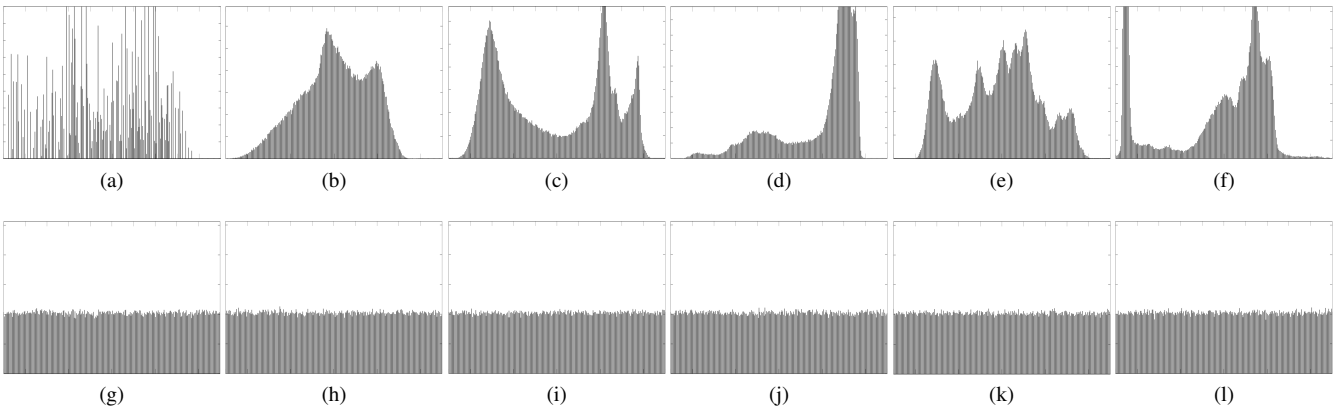


Figure 5: Histograms of original and transformed gray scale images used in the simulations. (a),(g) *peppers.bmp*; (b),(h) *mandril.bmp*; (c),(i) *lake.bmp*; (d),(j) *jetplane.bmp*; (e),(k) *lena.bmp*; (f),(l) *camera.bmp*.

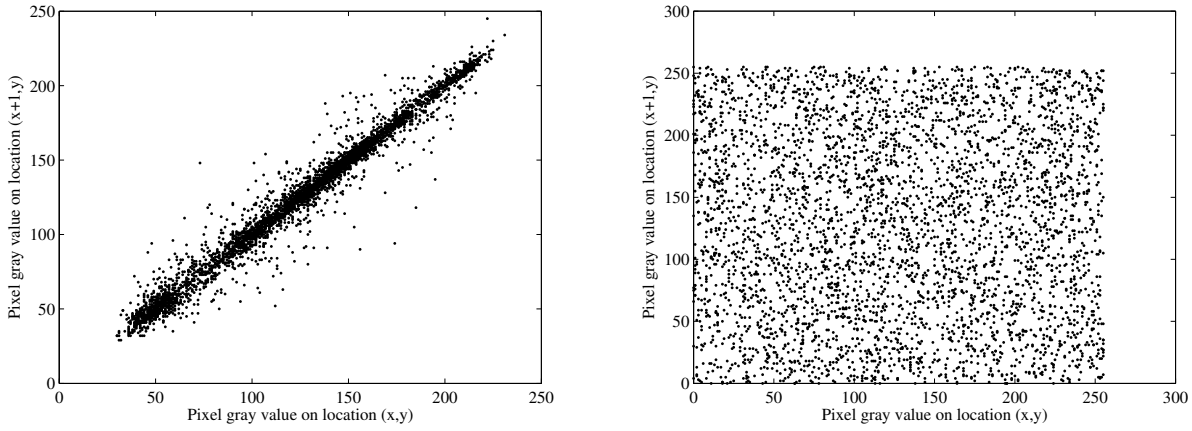


Figure 6: Correlation distributions in original and transformed image *lena.bmp*. Randomly chosen pixels and corresponding horizontal adjacent pixels are considered.

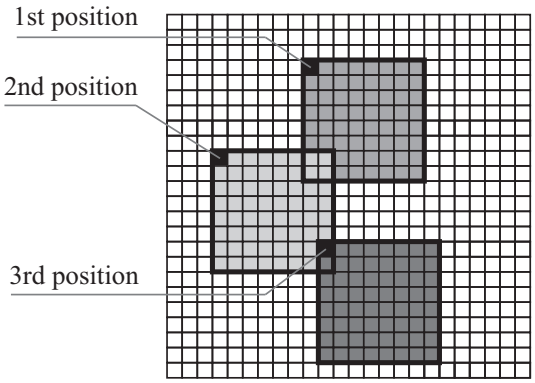


Figure 7: The positions of the black pixels in the image are determined by a secret key. Each selected pixel is taken as the top-left corner of an 8×8 image block to be recursively transformed using the FFCT.

step also based on the finite field cosine transform. Such a second step must be dependent on a secret key, which would be used to select blocks to be transformed by the FFCT. Such a transformation would also be recursive and more than one round could be applied. In Figure 7, the proposed second step is illustrated. The positions of the black pixels are determined by the key. Such pixels are then used as the top-left corner of 8×8 image blocks to be recursively transformed using the FFCT. More specifically, the positions of the black pixels can be defined by algorithms such as that one described in [19]. In this work, a key-dependent method for generating a sequence of pseudorandom points uniformly distributed in a two-dimensional region is given.

Differently from the uniformization procedure, in the second step, there is superposition among the blocks. This means that, if two keys lead to a difference in the determination of only one black pixel position, the resultant encrypted images can be substantially different; if we try to decrypt an image using a key which is minimally different from the correct key, the obtained image can also be significantly distinct from the original image. Analogously, if two minimally different images are encrypted using the same key, the correspondent ciphered images can be sufficiently distinct. These facts are related to the key sensitiveness of the method and to its resistance against the differential attack [15]. Naturally, details related to these aspects and also to the key-space of the method should be analyzed carefully.

Finally, we remember that, in the histogram uniformization, image blocks can be processed in parallel. On the other hand, in the second step, due to the superposition among the blocks, this is not possible. This makes important to reconsider the time delay involved in the entire system.

V. Concluding Remarks

In this paper, a procedure to uniformize image histograms, based on the finite field cosine transform, was introduced. As we emphasized throughout the text, the technique should be part of a complete image encryption system and its aim is to eliminate the effectiveness of attacks which explore the frequency of occurrence of the pixels values. This kind of attack may be useful against encryption techniques based on changes on pixels positions or transformations which do not alter significantly the histogram of an image. Simulations which indicate that the proposed scheme has potential applicability in practical scenarios were presented. The combination between the technique presented in this paper and procedures dependent on a key, with the purpose of implementing a complete image encryption scheme, is currently under investigation.

Acknowledgement

This research was supported by FACEPE, Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (APQ 1196-3.04/10).

References

[1] A. Cheddad, J. Condell, K. Curran, and P. M. Kevitt, "Digital image steganography: survey and analysis of current methods," *Signal Processing*, vol. 90, no. 3, pp. 727–752, March 2010.

[2] M. A. Suhail and M. S. Obaidat, "Digital watermarking-based DCT and JPEG model," *IEEE Trans. on Instrumentation and Measurement*, vol. 52, no. 5, pp. 1640–1647, October 2003.

[3] C.-P. Wu and C.-C. Jay Kuo, "Design of integrated multimedia compression and encryption systems," *IEEE Transactions on Multimedia*, vol. 7, no. 5, pp. 828–839, October 2005.

[4] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, Cambridge University Press, 1st edition, 2009.

[5] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking and Steganography*, Morgan Kaufmann, 2nd edition, 2007.

[6] S. Cimato and C. N. Yang, *Visual Cryptography and Secret Image Sharing*, CRC Press, 1st edition, 2011.

[7] G. Ye, "Image scrambling encryption algorithm of pixel bit based on chaos map," *Pattern Recognition Letters*, vol. 31, no. 5, pp. 347–354, April 2010.

[8] M. M. C. de Souza, H. M. de Oliveira, R. M. Campello de Souza, and M. M. Vasconcelos, "The discrete cosine transform over prime finite fields," in *International Conference on Telecommunications*, J. N. de Souza, P. Dini, and P. Lorenz, Eds., Berlin, 2004, Lecture Notes in Computer Science, pp. 482–487, Springer.

[9] J. B. Lima and R. M. Campello de Souza, "Finite field trigonometric tranforms," *Applicable Algebra in Engineering, Communication and Computing*, vol. 22, pp. 393–411, December 2011.

[10] R. M. Campello de Souza, H. M. de Oliveira, A.N. Kauffman, and A. J. A. Paschoal, "Trigonometry in finite fields and a new Hartley transform," in *Proc. IEEE Int. Symp. Information Theory (ISIT'98)*. IEEE, 1998, p. 293.

[11] J. B. Lima, R. M. Campello de Souza, and D. Panario, "The eigenstructure of finite field trigonometric transforms," *Linear Algebra and its Applications*, vol. 435, no. 8, pp. 1956–1971, October 2011.

[12] J. B. Lima, H. M. de Oliveira, and R. M. Campello de Souza, "Formatting probability distributions of integer numbers (in portuguese)," in *Proceedings of the XXV Brazilian Telecommunications Symposium*, Recife, Brasil, 2007.

[13] T. Toivonen and J. Heikkilä, "Video filtering with Fermat number theoretic transforms using residue number system," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 1, pp. 92–101, January 2006.

[14] R. E. Blahut, *Fast Algorithms for Signal Processing*, Cambridge University Press, 2010.

[15] A. Akhshani, S. Behnia, A. Akhavan, H. Abu Hassan, and Z. Hassan, "A novel scheme for image encryption based on 2D piecewise chaotic maps," *Optics Communications*, vol. 283, no. 17, pp. 3259–3266, September 2010.

[16] G. Ye, X. Huang, and C. Zhu, "Image encryption algorithm of double scrambling based on ASCII code of matrix element," in *Proc. 2007 International Conference on Computational Intelligence and Security*, Harbin, China, 2007, pp. 843–847.

[17] S. Luo and X. Chen, "Image encryption with control parameter chaotic map," in *Proc. 2nd International Conference on Computer Enginnering and Technology (ICCET)*, Chengdu, China, 2010, vol. 6, pp. 733–736.

[18] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, pp. 656–715, October 1948.

[19] E. A. Galperin and I. Galperin, "Small sample uniformity in random number generation," *Computers & Mathematics with Applications*, vol. 52, no. 1–2, pp. 95–108, July 2006.