PROJECTIVE AND AFFINE TRANSFORMATIONS IN THE MODIFICATION OF RSA ALGORITHM FOR DETACHED CLASSES OF IMAGES

Yuriy Y. Rashkevych, Dmytro D. Peleshko, Anatoly M. Kovalchuk, Zdzislaw Szymanski

IT Institute, Academy of Management, Lodz, Poland
zszymanski@swspiz.pl

Abstract

Based on the RSA algorithm, as the most common industrial standard of data encryption, modifications with projective and affine transformations for encryption of images that allow to strictly allocate outlines are proposed.

Key words: RSA algorithm, encryption, decryption, affine transformation, projective transformation.

1 Introduction

In today's world of questions development of information technology one of the most important questions is the security of information. One of the most common form of presentation of information in a digital form is a digital image.

One of the most common and secure encryption algorithms is RSA algorithm [1]. It refers to the most common group of public-key algorithms. The security of RSA algorithm is based on resource costly factorization of large numbers. The public- and private-keys are functions of two primes with 100-200 decimal digits and more.

RSA algorithm is a universal algorithm that can be applied to any signals. However, the drawback of this universality is that some classes of encrypted signals can be partially reproduced by other means of processing. One of these classes of signals are digital images. In this case, you need to implement special algorithms or modifications of existing ones.

2 Description of the RSA algorithm

The purpose of the RSA is resistant to unauthorized access encryption of signals. As signals for further research images with one byte pixel format [4] were chosen and those that allow to strictly allocate outlines.

Using algorithm RSA [1] as the most resistant to unauthorized decoding of encrypted signals, on images that allow to strictly allocate outlines, does not give satisfactory results. This is well illustrated by examples shown in Figure 1. As it can be seen from Figure 1, an encrypted image is "noise" set of pixel color values. However, it still basic contours of input images. That means the effect of incomplete image noise.

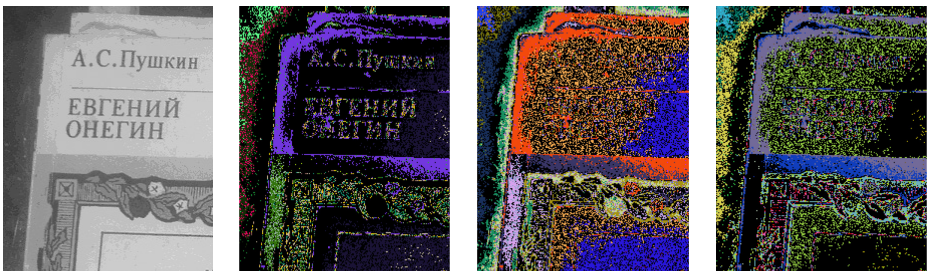
This effect is caused by the logic of mathematical transformation – encrypting operator of the RSA algorithm [1], which has the form

$$\tilde{c}_{i,j} = (c_{i,j})^e mod\ z, \quad (1)$$

where $c_{i,j}$ – value of pixel color (or brightness in case of image in "grayscale") with coordinates (i, j); $\tilde{c}_{i,j}$ encrypted value of pixel color (or brightness in case of image in "grayscale") with coordinates (i, j); mod – mathematical function of modulo division; z – second operand (encryption key) of mod function, which is defined as the product of two primes z1 and z2 [1] of same dimension; e – number, that is coprime with (z1 – 1)(z2 – 1).

"Noise" of the image in the RSA algorithm is provided by mathematical function mod. Mod function, with fixed second operand z, possesses significant homogeneous regions and the length of these regions depends on the ratio of z to $c_{i,j}$. That is why with close values of $c_{i,j}$ we get close values of $\tilde{c}_{i,j}$. This property provides "noise".

On the other hand, if the values $c_{i,j}$ are sharply fluctuational (typical for images that allow to strictly allocate outlines), we may have fluctuations of $\tilde{c}_{i,j}$ for fixed z. This causes a situation, when the outlines of original images can be seen on the encrypted image.



a) b) c) d)

Figure. 1. Samples of images, encrypted by RSA algorithm with different keys: a) input image; b) encrypted with z_1=53, z_2=103, e=33769; c) encrypted with z_1=73, z_2=97, e=6197; d) encrypted with z_1=67, z_2=167, e=1451

That is why the image, encrypted by RSA algorithm, cannot be considered as fully encrypted. This is because some information can be taken from the encrypted image by traditional image processing, such as filtering, reconstruction and others, without "breaking" the RSA algoritm (effective ways to "break" the RSA algorithm are currently unknown). Despite the fact that the second method does not fully reproduce the input image, it can provide some information out of encrypted image.

3 Presentation of image by the matrix of pixels colors

Image P of dimensions: h – height – number of pixels vertically, and l – length – number of pixels horizontally, can be considered as a matrix of pixels $P_{l,h}$, which can be transformed to a matrix of pixels colors $C_{P_{l,h}}$.

$$P = P_{l,h} = \left\lfloor pxl_{ij} \right\rfloor_{1\le i\le n(l),\,1\le j\le m(h)} \to C_{P_{l,h}} = \left\lfloor c_{ij} \right\rfloor_{1\le i\le n(l),\,1\le j\le m(h)}, \qquad (1)$$

where pxl_{ij} – pixel with coordinates (i, j) of discrete image P; n and m – a number of pixels in the directions l and h, respectively.

4 Modification of the RSA algorithm with projective transformations

Partial projective transformation described in [2] has the form:

$$\begin{cases} u = \dfrac{(p_2 - p_3)(x - p_1)}{(p_2 - p_1)(x - p_3)}, \\ v = \dfrac{(p_2 - p_3)(y - p_1)}{(p_2 - p_1)(y - p_3)}; \end{cases} \tag{2}$$

if

$$\delta = \begin{vmatrix} p_2 - p_3 & -u(p_2 - p_1) \\ v(p_2 - p_1) & -(p_2 - p_3) \end{vmatrix} \neq 0, \tag{3}$$

then there is an inverse transformation to (2)

$$\begin{cases} x = \dfrac{\delta_x}{\delta}, \\ y = \dfrac{\delta_y}{\delta}; \end{cases}, \tag{4}$$

where

$$\delta_x = \begin{vmatrix} p_1(p_2 - p_3) - p_3 u(p_2 - p_1) & -u(p_2 - p_1) \\ p_3 v(p_2 - p_1) - p_1(p_2 - p_3) & -(p_2 - p_3) \end{vmatrix}, \tag{5}$$

$$\delta_y = \begin{vmatrix} (p_2 - p_3) & p_1(p_2 - p_3) - p_3 u(p_2 - p_1) \\ v(p_2 - p_3) & p_3 v(p_2 - p_1) - p_1(p_2 - p_3) \end{vmatrix}. \tag{6}$$

Encryption:

- two adjacent elements in one row of input picture matrix (1) are taken
$$x = c_{i,j},\ y = \frac{c_{i,j+1} + c_{i,j}}{2},\ i = \overline{1,n},\ j = \overline{1,m-1};$$
- encrypted using formula (2) with
$$p_1 = z_2,\ p_2 = z_1 + 2(z_2 + d),\ p_3 = e - d;$$
- written to appropriate places of encrypted picture matrix.

Decryption:

- two adjacent elements in one row of the encrypted picture matrix are taken
  $x' = c'_{i,j},\ y' = c'_{i,j+1},\ i = \overline{1,n},\ j = \overline{1,m-1}$;
- decrypted using formula (4), taking into account (3), (5) and (6);
- written to appropriate places of the decrypted picture matrix in the following way:
  $c_{i,j} = x,\ c_{i,j+1} = 2y - c_{i,j},\ i = \overline{1,n},\ j = \overline{1,m-1}$.

Figure 2 shows samples of modified algorithm with different keys.

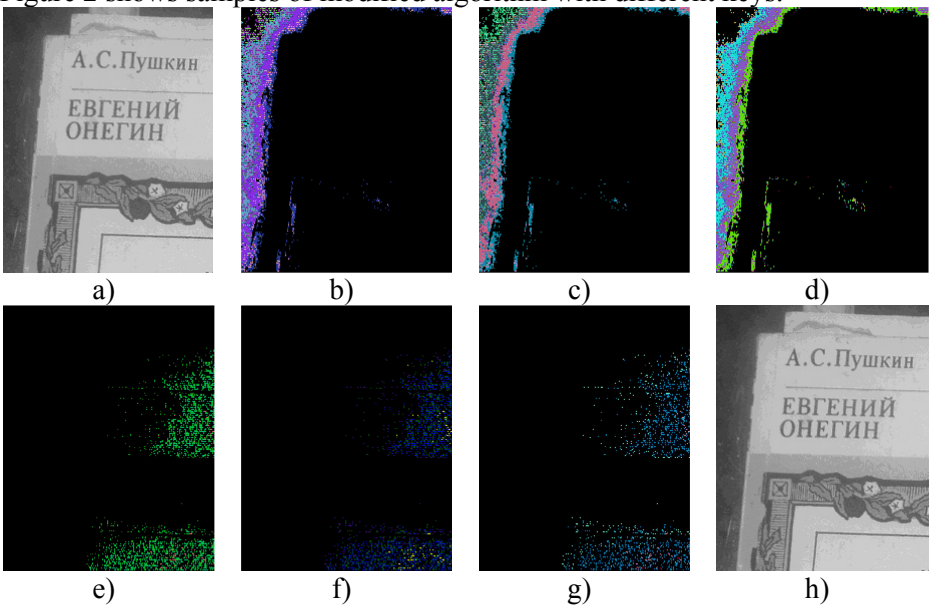


Figure. 2. Samples of images, encrypted and decrypted by the modified algorithm using projective transformations with different keys: a) input image; b) encrypted with z_1=53, z_2=53, e=2609, d=1537; c) encrypted with z_1=59, z_2=53, e=1181, d=2541; d) encrypted with z_1=97, z_2=59, e=569, d=137;e) encrypted with z_1=103, z_2=191, e=1979, d=18953; f) encrypted with z_1=109, z_2=191, e=4231, d=14671; g) encrypted with z_1=139, z_2=191, e=1979, d=7499; h) decrypted image.

5 Modification of RSA algoritm with affine transformations

Binary affine transformation [3] of plane in Cartesian coordinates has the form:

$$\begin{aligned} x' &= a_1 x + b_1 y + d_1, \\ y' &= a_2 x + b_2 y + d_2;' \end{aligned} \tag{7}$$

if

$$\delta = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} \neq 0\,, \tag{8}$$

then there is an inverse transformation to (7)

$$x = \frac{\Delta_x}{\delta},\ y = \frac{\Delta_y}{\delta}, \tag{9}$$

where

$$\Delta_x = \begin{vmatrix} x' - d_1 & b_1 \\ y' - d_2 & b_2 \end{vmatrix};\quad \Delta_y = \begin{vmatrix} a_1 & x' - d_1 \\ a_2 & y' - d_2 \end{vmatrix}, \tag{10}$$

Encryption and decryption can be handled in two ways: using two adjacent elements in one row or in one column of picture matrix.

1) encryption and decryption, using two adjacent elements in one row of picture matrix

Encryption:

- two adjacent elements in one row of input picture matrix (1) are taken
  $x = c_{i,j},\ y = \dfrac{c_{i,j+1} + c_{i,j}}{2},\ i = \overline{1,n},\ j = \overline{1,m-1}\,;$
- encrypted using formula (7) with
  $a_1 = b_2 = (z_1 + z_2)^e \pmod z, \qquad b_1 = a_2 = (z_1 - z_2)^d \pmod z,$
  $d_1 = j \cdot j,\ d_2 = j \cdot j \cdot j,\ j = \overline{1,m}\,;$
- written to appropriate places of the encrypted picture matrix.

Decryption:

- two adjacent elements in one row of the encrypted picture matrix are taken
  $x' = c'_{i,j},\ y' = c'_{i,j+1},\ i = \overline{1,n},\ j = \overline{1,m-1}$;
- decrypted using formula (9), taking into account (8) and (10);
- written to appropriate places of the decrypted picture matrix in the following way:
  $c_{i,j} = x,\ c_{i,j+1} = 2y - c_{i,j},\ i = \overline{1,n},\ j = \overline{1,m-1}$.

Figure 3 shows samples of the modified algorithm with different keys.

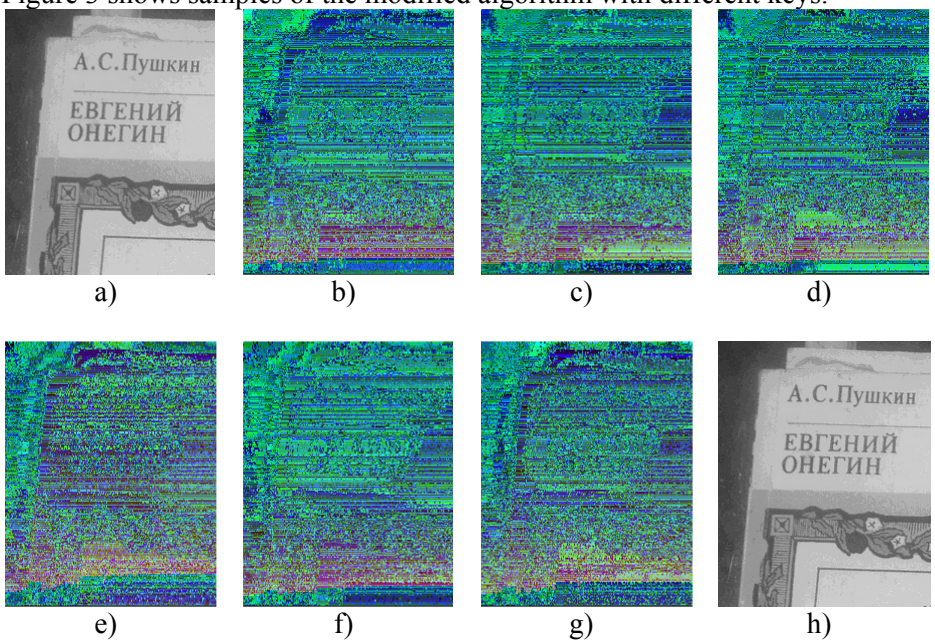


Figure. 3. Samples of images, encrypted and decrypted by the modified algorithm using affine transformations to two adjacent elements in one row of picture matrix with different keys: a) input image; b) encrypted with z_1=61, z_2=199, e=4019, d=10739; c) encrypted with z_1=67, z_2=89, e=4663, d=3079; d) encrypted with z_1=67, z_2=181, e=2557, d=11053;e) encrypted with z_1=83, z_2=191, e=8017, d=14173; f) encrypted with z_1=97, z_2=89, e=5147, d=4883; g) encrypted with z_1=97, z_2=167, e=6823, d=6103; h) decrypted image.

2) encryption and decryption, using two adjacent elements in one column of the picture matrix

Encryption:

- two adjacent elements in one column of the input picture matrix (1) are taken

$$x = c_{i,j},\ y = \frac{c_{i+1,j} + c_{i,j}}{2},\ i = \overline{1, n-1},\ j = \overline{1, m};$$

- encrypted using formula (7) with

$$a_1 = b_2 = (z_1 + z_2)^e \pmod z, \qquad b_1 = a_2 = (z_1 - z_2)^d \pmod z,$$
$$d_1 = -j \cdot j,\ d_2 = -j \cdot j \cdot j,\ j = \overline{1, m};$$

- written to appropriate places of the encrypted picture matrix.

Decryption:

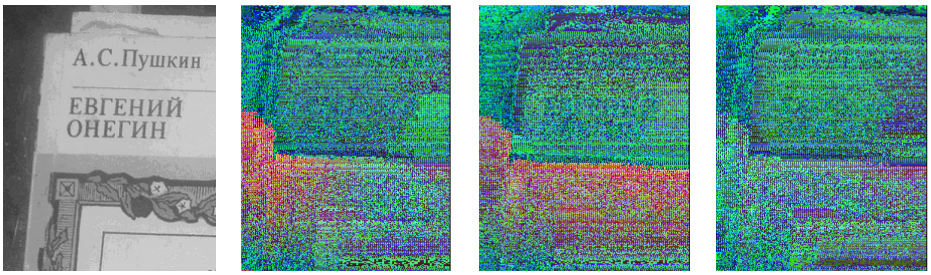
- two adjacent elements in one column of the encrypted picture matrix are taken

$$x' = c'_{i,j},\ y' = c'_{i+1,j},\ i = \overline{1, n} - 1,\ j = \overline{1, m};$$

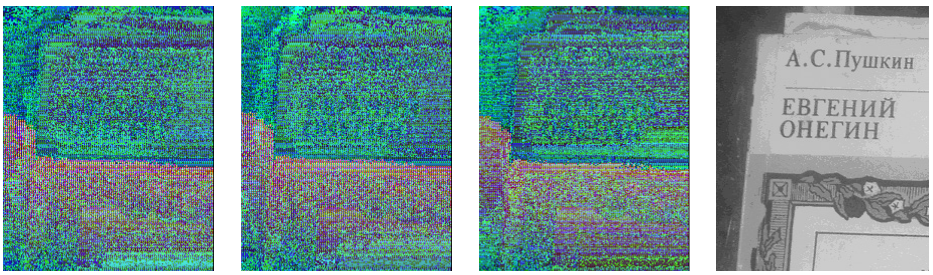
- decrypted using formula (9), taking into account (8) and (10);
- written to appropriate places of the decrypted picture matrix in the following way:

$$c_{i,j} = x,\ c_{i+1,j} = 2y - c_{i,j},\ i = \overline{1, n-1},\ j = \overline{1, m}.$$

Figure 4 shows samples of the modified algorithm with different keys.



a) b) c) d)



e) f) g) h)

Figure. 4. Samples of images, encrypted and decrypted by modified algorithm using affine transformations to two adjacent elements in one row of picture matrix with different keys: a) input image; b) encrypted with z1=73, z2=197, e=5507, d=8459; c) encrypted with z1=79, z2=191, e=7951, d=3931; d) encrypted with z1=97, z2=173, e=12197, d=3245;e) encrypted with z1=97 z2=181, e=6869, d=6269; f) encrypted with z1=101, z2=173, e=2131, d=5771; g) encrypted with z1=103, z2=193, e=7331, d=9227; h) decrypted image.

Conclusions

References

1. Б.Шнайдер. Прикладная криптография.– М.: Издательство Триумф, 2003.-816с.
2. Юрій Рашкевич, Анатолій Ковальчук , Дмитро Пелешко. Проективні відображення першого порядку в шифруванні і дешифруванні зображень з елементами алгоритму RSA. – Львів: Технічні вісті 2009/№1(29), 2(30). С. 41 – 44.
3. Ю.М. Рашкевич, А.М. ковальчук, Д.Д. Пелешко. Афінні перетворення в модифікаціях алгоритму RSA шифрування зображень. – Львів: ААЕКС, 2009/№2.
4. Р.Гонсалес, Р.Вудс. Цифровая обработка изображений.– М.: Техносфера, 2005.-1072с.