Performance Analysis of Binary Watermarking Algorithm Against Geometric Attacks

Zoran Veličković, Marko Veličković
Academy of Applied Technical and Preschool Studies
Section Niš
Niš, Serbia
zoran.velickovic@akademijanis.edu.rs

Zoran Milivojević
MB University
Department of Information technologies
Beograd, Serbia
zoran.milivojevic@akademijanis.edu.rs

Bojan Prlinčević
Kosovo and Metohija Academy of Applied Studies
Leposavić
Kosovo and Metohija
bojan.prlincevic@akademijakm.edu.rs

Abstract— **In this paper, the performance of the proposed Binary Watermarking Algorithm against geometric attacks is determined. A QR code encoding important security information was used as a watermark. The performance of the Binary Watermarking Algorithm was determined in the presence of geometric attacks performed by rotating, scaling, and cutting parts of the protected image. The intensity of the attack also varied according to the type of geometrical attack. The proposed Binary Watermarking Algorithm has the possibility of varying the insertion intensity, tests were performed for standard differential threshold values. To present all the obtained results, 3D graphics were created. The obtained results confirm the resistance of the proposed Binary Watermarking Algorithm to certain geometric attacks. Decoding the extracted watermark in the form of a QR code is possible even in case of errors due to the error correction capability of the QR code. The resistance of the Binary Watermarking Algorithm to geometric attacks recommends the use of the proposed algorithm for image protection purposes.**

Keywords - Binary watermark algorithm, YCbCr color model; Geometrics attacks; QR code; NCC; Bit error.

I. INTRODUCTION

Technological progress has enabled many new services for network users such as cloud computing, e-commerce, mobile banking, on-demand streaming, Internet television, Internet of Things, and the like [1]. Multimedia distribution companies even allow users to upload personal multimedia content to their servers. New online services have improved the user experience, but at the same time they have increased security risks.

On the other hand, modern users expect network services to be always available and secure. In fact, it is expected that users' assets will not be compromised online and protected from malicious attacks. We witness security threats happening online every day, which often lead to serious personal and business losses. In addition to malicious attacks that cause financial loss, there are also attacks that threaten users' personal data and then misuse it. The security threats related to the identities replacement of user are significant. This problem is related to copyright protection, that is, to the problem of illegal use of multimedia content. This type of security threat gains importance in the context of contemporary war events in Europe and the Middle East. Users are flooded daily with information whose exact origin is unknown. A handful of photos and videos are distributed through the media, for which it is difficult to determine the source, place, and time of the event. A similar phenomenon happened in the recent COVID-19 pandemic when the web was flooded with false and unreliable information. Fake news about the COVID-19 pandemic spread faster than the virus itself, so this phenomenon has been called "*infodemia*" [2].

From the mentioned events, a simple conclusion can be drawn that in crisis situations there is almost always a flood of information that, at the very least, confuses the user. Information is often tendentiously falsified to cause panic or realize political or other goals. Under such conditions, users have almost no way to distinguish true information from false information [3].

To prevent the illegal use of multimedia content, that is, to recognize the source of the content, numerous techniques have been developed. Although multimedia content can be protected by encryption, the classic technique based on PKI (Public Key Infrastructure) is not used for these purposes. For several reasons, which we will not go into here, this concept is not adequate for this type of multimedia content.

Inserting a watermark into the multimedia content itself (watermarking) is a technique that can solve problems related to the identification of content sources [4]. On the receiving side, an embedded watermark is extracted from the multimedia content itself which carries information about the creator and/or origin of the multimedia content. It is important to note that the embedded watermark should not be visible and should not cause visible degradation of the carrier multimedia content [5]. Also, the watermark should persist in multimedia content even after various types of attacks that can be applied to destroy the embedded watermark. A good feature of this concept

multimedia content protection is that attacks the multimedia content and the embedded watermark equally. Too high an intensity of attacks can of course destroy the watermark but also the multimedia content itself. Clearly, destroyed multimedia content has no value to attackers. Therefore, the basic idea of the attacker is to somehow destroy the inserted watermark, and to preserve the multimedia content and possibly place his own.

Geometric and non-geometric attacks are used to destroy watermarks in protected multimedia content (images in this paper). Non-geometric attacks are based on digital image processing and consist in adding interference that degrades protected images but also complicates watermark extraction. In previous works, the authors specifically considered attacks such as JPG compression, Gaussian noise, Speckle noise and Salt & Pepper [6]. Based on the results obtained during watermark extraction, the performance of the proposed Binary Watermarking Algorithm was determined.

To determine the performance of the applied Binary Watermarking Algorithm against geometric attacks, protected multimedia contents are exposed to different types of geometric attacks. A watermark is then extracted from the thus attacked multimedia content. Watermark extraction was evaluated for images with horizontal and vertical parts cut off. By increasing the size of the cut surface, the intensity of the attacks increases. The cropped area of the image is varied by the number of pixels in a certain direction. Also, the performance of the proposed binary watermarking algorithm was determined as a function of the image rotation angle. An attack test was also conducted in the form of scaling of the protected image for different values of the scaling factor.

The proposed algorithm enables the selection of the insertion strength of the watermark in the image, so tests were performed for several characteristic values of the differential threshold T, which determines the insertion strength. A large value of the differential threshold T causes a greater degradation of the protected multimedia content, but also a more reliable extraction of the inserted watermark. On the other hand, a smaller value of the differential threshold T causes a smaller degradation of the protected multimedia content, but also a less reliable extraction of the inserted watermark. This applies to all types of attacks.

Geometric attacks on protected multimedia content significantly reduce the probability of error-free watermark extraction. Since the intensity of the geometric attack can also be varied, tests were performed for different attack intensities.

In the paper, the applied algorithm was evaluated for rotating multimedia content - images. Rotating the image represents a mathematical transformation that results in the rounding of numerical values, so the extraction of the watermark for certain angles is difficult. Parts of the watermark may suffer minor or major degradation because of rotating the image. Also, in the paper, a test was performed in the form of an attack that is carried out by scaling the protected image. Testing was performed for several values of the scaling factor and the results are shown in 3D graphics as the attack was performed for different values of the insertion strength. The classification of watermarks was performed, and the application of the QR code as a watermark was considered. The usability of the watermark in solving the problem of identifying the creator, place and time of creation is shown. In the third chapter, the algorithms applied in this area are described. Insertion algorithms are considered, while extraction algorithms are considered as their inverse algorithms. In the fourth chapter, the proposed Binary Watermarking Algorithm is described in detail, while in the fifth, the obtained results of simulated geometric attacks on the protected image are presented. Through a series of 3D graphics, the obtained results and determined performance of the proposed binary watermarking algorithm in the case of geometric attacks are presented.

II. WATERMARK TECHNOLOGY

Watermark is a multimedia content protection technology based on inserting a secret - invisible image into the multimedia content itself. This image, which is invisible to users, can be extracted with appropriate algorithms and used as proof of the owner of the multimedia content. A watermark can be inserted into all types of multimedia content, from audio to images [5] to videos [7]. Multimedia content in which a watermark has been inserted is referred to as protected multimedia content in the following text, that is, in this paper it is a protected image.

A color image, a monochrome image (in shades of gray) or a binary image can be used as a watermark. Depending on the image used as a watermark, color images require the highest information capacity, followed by monochrome images and finally binary images. Also, the size of the watermark also plays a very important role considering the number of pixels that need to be inserted into the original multimedia content. The optimal choice of a watermark depends on many factors, but some of the most important are the needs of the user, the information capacity of the applied algorithm, the dimensions of the watermark and the like.

Most often, logos of companies that created multimedia content are used as watermarks. This type of watermark provides a good basis for proving ownership of multimedia content. However, the logo as a type of watermark does not provide additional information about the place and time of content creation. To somehow reconcile the conflicting requirements regarding the choice of a watermark, in this work a binary image is used as a watermark, which was created by encoding information using one of the QR (Quick Response) codes.

A QR code is a two-dimensional barcode that can be represented as a two-dimensional matrix whose elements can be binary values. By printing this matrix in a specific way, a 2D image is obtained. A QR code has sections that are important for decoding. The information capacity required from the QR code also determines its dimensions.

The smallest QR code measures 21×21 pixels with an information capacity of 152 bits. A QR code with a dimension of 177×177 pixels can store 23648 bits. In addition to data memorization, the QR code allows the selection of 4 levels of error correction: low, medium, quartile and high. The central area of the QR code is reserved for encoded data. In addition to storing data in the QR code, it provides an error correction mechanism, which is a significant advantage over the standard binary images used by the authors in previous works.

A watermark in the form of a QR code can encode and store arbitrary data. For this work, it is important to encode data about the creator of multimedia content as well as about the place and time of creation. Therefore, the previously described security network problems can be eliminated by using a QR code as a carrier of important information.

Therefore, the watermark is inserted into the carrier image, while the watermark is extracted from the protected image using the inverse algorithm, with known insertion parameters. By extracting the watermark and then decoding it, it is possible to prove the ownership of multimedia content, that is, to decode the true source of information.

III. WATERMARKING ALGORITHMS

Watermarking algorithms usually consist of two parts. One part is about inserting the watermark, while the other part is about extracting the watermark. The part of the algorithms related to inserting the watermark should not cause noticeable degradation of multimedia content (images). At the same time, the part of the algorithm related to watermark extraction should ensure reliable extraction and with as few errors as possible. In this work, the QR code is used as a watermark, which encodes information about the author, owner, place, time of creation of the content, etc. These are two parts of algorithms that pose conflicting technical requirements that quality watermarking algorithms need to successfully reconcile.

Watermark insertion can be done by algorithms based on the spatial domain of the image or in one of the transform domains of multimedia content (in our case, images). In algorithms for inserting watermarks in the spatial domain, the watermark itself is hidden in the values of the luminance components of the spatially distributed pixels of the image. On the other hand, with algorithms for inserting a watermark in the transformation domain, the watermark itself is hidden in the transformation coefficients of the image. Standard watermark insertion algorithms use a luminescent domain to insert the watermark. This results in visible degradation of the image as the human eye is very sensitive to the change of this component in the image. To eliminate this major drawback, the authors of this paper recommended using the color domain for watermark insertion. The idea comes from the characteristic of the eye, which is significantly less sensitive to the chrominance component of the image. To implement this idea, it is good to use encoders that separate the luminance component of the image from the chrominance component.

Insertion strength is a parameter of these algorithms that determines the strength of watermark inserting in multimedia content. Depending on the applied algorithm, the insertion strength is determined in various ways. Thus, for example, the strength of insertion in SVD algorithms is determined by the insertion factor α, while in this work a parameter called the differential threshold T is used for the same purpose. In general, it can be said that a higher insertion strength leads to an increased degradation of carrier image quality, but in return, a better watermark quality is obtained. The compromise that embedding algorithms need to satisfy is to perform watermark embedding with enough strength to extract the watermark without errors, but not to cause noticeable carrier image degradation. Another important property of embedding algorithms is to be resistant to attempts to destroy watermarks in multimedia content. Attacks on protected multimedia content can be different but can generally be classified into geometric and non-geometric attacks. The ability to extract the watermark after the attack also determines the performance of the algorithm itself. Another important feature required of watermarking algorithms is false-positive problem detection. Namely, with some algorithms, it was observed that it is possible to extract a watermark that was not even inserted. This phenomenon occurs with so-called blind watermarking algorithms, which require the original watermark and the original multimedia content in the extraction process. This phenomenon was observed in standard SVD watermarking algorithm. One of the solutions for false positive detection problem is to apply SVD transformation based on the principal component [8].

In this paper, the so-called block SVD transformation was used, which does not have the problem of false positive detection. Also, this algorithm is particularly suitable for inserting and extracting binary watermarks. The disadvantage that can be attributed to this algorithm is the limited information capacity. The details of the algorithm will be described in the following chapters. To improve the performance of certain insertion algorithms, hybrid techniques are applied. Hybrid techniques involve the application of several transformation domains such as YCrCb, DCT (Discrete Cosine Transform), SVD (Singular Value Decomposition) and DWT (Discrete Wavelet Transform).

Depending on the application, watermarking algorithms need to have some security. This component is sometimes provided by encryption watermarking algorithms. In one case, standard encryption algorithms are used, but as already mentioned, this is not always a good solution. On the other hand, some algorithms use the Arnold transform [9] to encrypt the watermark before insertion. To decrypts the watermark, it is necessary to know the parameters of the Arnold transformation. Most often, the information is encoded with a QR code and then encoded with an Arnold transformation before insertion. To decode the extracted watermark, it is necessary to apply the inverse Arnold transformation, for which it is necessary to know its parameters. This is an additional component that improves the security features of the watermarking algorithms.

IV. PROPOSED WATERMARKING ALGORITHM

Details of the recommended Binary Watermarking Algorithm will be presented below. First, the part related to inserting the watermark will be described, and then the part related to extracting the watermark. In the part of the algorithm related to insertion, it is assumed that the carrier image is given in RGB format. Therefore, before applying the watermark insertion algorithm, it is necessary to decompose the image into color and liminescence domains. This algorithm uses the YCrCb color model that satisfies the prerequisite [10]. The transformation from the RGB model to the YCbCr color model is lossless and is standardly used in image and video processing. Part of the insertion algorithm is based on the block SVD decomposition of the chrominance component of the carrier image [6]. The SVD decomposition coefficients of the Cb color

Fig. 1. Layouts of original carrier images with references used in the evaluation a) Lena b) Mandril c) Pepper d) Airplane e) Fruits.

component is modulated based on the individual bit value of the watermark. All the steps of the insertion algorithm carry the **IN** prefix, while the individual steps are marked with subscript number.

Step **IN₀**: Create a binary watermark using a QR code for the desired text data.

Step **IN₁**: Transform the carrier image I from RGB to YCrCb color model (optional).

$$\boldsymbol{YCbCr} = rgb2ycbcr(\boldsymbol{I}) \tag{2}$$

Step **IN₂**: Divide the color component **Cr** into non-overlapping $\mathbf{H}_{i,j}$ blocks of size 4×4 pixels. Individual blocks are identified by index *i* and *j*.

Step **IN₃**: Perform SVD decomposition for each block $\mathbf{H}_{i,j}$ from the **Cr** component:

$$\boldsymbol{H}_{i,j} = \boldsymbol{U}_{i,j} \times \boldsymbol{S}_{i,j} \times \boldsymbol{V}_{i,j}^{T}. \tag{3}$$

Step **IN₄**: Modify the elements from the second and third rows of the first column of each $\mathbf{U}_{i,j}$ matrix (elements $u_{2,1}$ and $u_{3,1}$) based on the value of each individual bit *w* from the QR code image:

$$if\ \ w = 1, \begin{cases} u_{2,1}^{*} = sign(u_{2,1}) \times \left(U_{avg} + \frac{T}{2}\right) \\ u_{3,1}^{*} = sign(u_{3,1}) \times \left(U_{avg} - \frac{T}{2}\right) \end{cases} \tag{4}$$

$$\text{else } if\ \ w = 0, \begin{cases} u_{2,1}^{*} = sign(u_{2,1}) \times \left(U_{avg} - \frac{T}{2}\right) \\ u_{3,1}^{*} = sign(u_{3,1}) \times \left(U_{avg} + \frac{T}{2}\right) \end{cases} \tag{5}$$

$$U_{avg} = \frac{(|u_{2,1}| + |u_{3,1}|)}{2} \tag{6}$$

where T is the differential threshold, and it is parameter of the algorithm used to determine the insertion strength. The modified matrix is denoted by $\mathbf{U}^{*}_{i,j}$.

Step **IN₅**: Perform the inverse SVD transformation of each $\mathbf{H}_{i,j}$ block to obtain $\mathbf{H}^{*}_{i,j}$ block with the inserted bit.

$$\boldsymbol{H}_{i,j}^{*} = \boldsymbol{U}_{i,j}^{*} \times \boldsymbol{S}_{i,j} \times \boldsymbol{V}_{i,j}^{T} \tag{7}$$

Step **IN₆**: Place the block with the inserted QR code bit in the appropriate place in the protected carrier image. Repeat steps **IN₄** and **IN₅** for all bits from the QR code.

$$\boldsymbol{RGB}^{*} = ycbcr2rgb(\boldsymbol{I}^{*}) \tag{8}$$

In this way, all bits from the QR code are inserted into the carrier image. In order to extract the watermark inserted in this way, it is not necessary to have the originals of either the image or the watermark, so this algorithm belongs to the class of BLIND algorithms. The algorithm for extracting a watermark from a protected image is shown in a series of **EX** steps with step number in subscript.

Below is a part of the watermarking algorithm related to watermark extraction.

Step **EX₁**: Perform transformation of protected carrier image I' from RGB to YCrCb color model (optional).

$$\boldsymbol{YCbCr} = rgb2ycbcr(\boldsymbol{I}') \tag{9}$$

Step **EX₂**: Divide the **Cb** component of the protected image into non-overlapping blocks $\mathbf{H'}_{i,j}$ of dimensions 4×4 pixels. Individual blocks are identified by index i and j.

Step **EX₃**: Perform SVD decomposition over all blocks $\mathbf{H'}_{i,j}$ carrying image.

$$\boldsymbol{H}_{i,j}' = \boldsymbol{U}_{i,j}' \times \boldsymbol{S}_{i,j}' \times \boldsymbol{V}_{i,j}'^{T} \tag{10}$$

Step **E₄**: The value of the corresponding extracted watermark bit *w'* is obtained by applying the following expressions:

$$w' = 1, \begin{cases} 0, & if\ u_{2,1}' > u_{3,1}' \\ 1, & if\ u_{2,1}' \le u_{3,1}' \end{cases} \tag{11}$$

Step **EX₅**: Set the extracted bit value to the appropriate location in the watermark.

Step **EX₆**: Repeat steps **EX₄** and **EX₅** for all blocks of the protected carrier image and form the complete binary image of the QR code. The resulting binary image represents the extracted QR code.

V. Experimental Results

To determine the performance of the proposed Binary Watermarking Algorithm for inserting a in the form of a QR code, the known color images of Lena, Mandril, Pepper, Airplane and Fruits with a resolution of 512×512 pixels [11] were used as carriers. Layouts of the carrier images for algorithm evaluation are shown in Fig. 1. A watermark created based on the text "Lena Forsen (Sjooblom), Swedish model" with a QR code was inserted into all the carrier images by the proposed binary algorithm. The resulting watermark is a binary image with dimensions of 64×64 pixels and is shown in the third row in Fig. 2a. The proposed binary image insertion algorithm allows the selection of the insertion strength by varying the differential threshold parameter T. For each carrier image, a series of

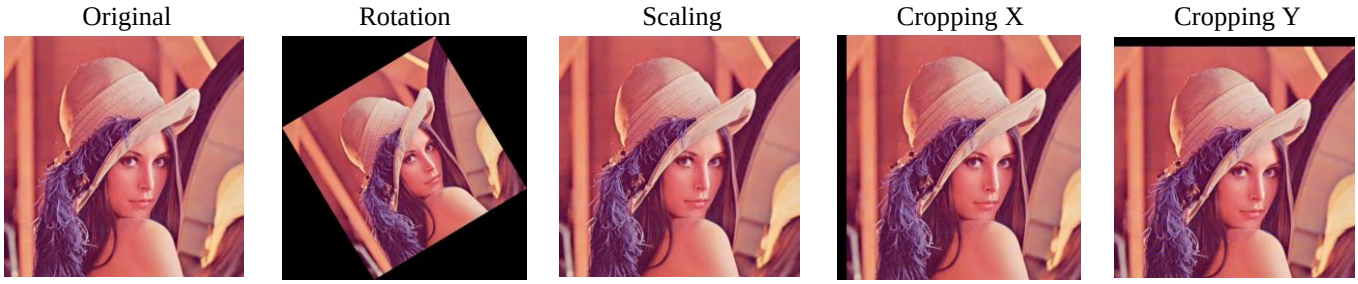

Fig. 2. Layouts a) unprotected carrier image "Lena" and original watermark QR code created for text "Lena Forsen (Sjooblom), Swedish model" b) protected image "Lena" with attack in the form of Rotation, 60^0 c) protected image "Lena" with attack in the form Scaling with 1.8 scale d) protected image "Lena" with attack in the form of horizontal cropping with 20 pixels d) protected image "Lena" with attack in the form of vertical cropping with 16 pixels.

protected images with different values of the differential threshold T were created. In the experiment, 6 protected images were generated for all carrier images with differential insertion threshold T = [0.1 0.15 0.2 0.25 0.3 0.35]. The proposed binary algorithm defines that the watermark is inserted only in the Cr chrominance channel of the carrier image. In total, 30 different protected images were generated, on which different geometric attacks are applied. Watermarks of different quality are extracted from such attacked protected images.

In previous works, the problem of choosing the differential threshold T was considered and its optimal value was determined without the presence of attacks. In the conditions of non-geometric attacks, a higher value of the differential threshold T should be chosen because it ensures the extraction of the watermark with fewer errors. In the experimental part of this work, geometrical attacks were analyzed, which include rotating, scaling and cropping the image. The performance of the proposed binary algorithm is determined based on objective parameters based on the absolute number of bit errors in the extracted watermark - ERROR and the normalized correlation coefficients NCC of the extracted watermark in relation to the original watermark [12]. The number of errors in the extracted watermark, which is considered a bit image, is determined according to the expression (12)

$$ERROR(w, w') = \sum_{k=1}^{K} \sum_{l=1}^{L} [w(k,l) \otimes w'(k,l)] \quad (12)$$

where $w(k,l)$ refers to the original, and $w'(k,l)$ refers to the extracted watermark. The variables K and L are the watermark dimensions, and their value is 64, while $\otimes$ is the XOR operator. The normalized correlation coefficient NCC is determined according to expression (13).

$$NCC(w, w') = \frac{\sum_{k=1}^{K} \sum_{l=1}^{L} [w(k,l) \times w\prime(k,l)]}{\sqrt{\sum_{k=1}^{K} \sum_{l=1}^{L} w^2(k,l)} \times \sqrt{\sum_{k=1}^{K} \sum_{l=1}^{L} w^2(k,l)}} \quad (13)$$

where $w(k,l)$ refers to the original, and $w'(k,l)$ refers to the extracted watermark. The results of the experiment are shown by a series of graphs in Fig. 3 related to geometrical attacks: Rotation and Cropping of the image. The 3D plots in these figures show the performance of the binary algorithm based on the number of errors in the extracted watermarks (ERROR) and the correlation coefficients between the original and extracted watermarks (NCC). Due to lack of space, Fig. 3 shows the results for only one carrier image - Lena. Graphics related to other carrier images are similar in form but differ in some details.

From Fig. 3a, in the attacks performed by rotating the image by the appropriate angle Rotate angle = [20 60 100], the number of errors in the extracted binary watermark decreases with the increase of the insertion strength, which is regulated by the value of the differential threshold T, which was expected. For the rotation angles shown, the insertion strength has a more significant influence on the number of extracted errors in relation to the rotation angle. As for the NCC correlation coefficients, they are very high even for lower values of the parameter T. With increasing insertion strength, the NCC coefficients grow almost independently of the rotation angle. It has been observed that for a small number of specific angles, an unusually large number of errors occur, which are the result of rounding due to the raster nature of the image.

Fig. 3b refers to the results obtained by the geometric attack, which is realized by cutting the protected image horizontally (x direction). From these graphs, the number of errors and NCC coefficients decrease with the increase of the cut area. The size of the cropped area is given as a function of the number of pixels. These results are expected. Fig. 3c refers to the results obtained by the geometric attack, which is realized by cutting the protected image vertically (y direction). It can be seen from the graph that the number of errors directly depends on the cut surface, and that the strength of the insertion does not affect the number of errors. It is interesting to note that with the increase of the cut part, the number of errors oscillates and decreases for the examples shown. This phenomenon is significantly different from the phenomenon of clipping in the horizontal direction and can be attributed to the raster in which the pixel values are observed. The values of the NCC coefficients, which alternately change their sign as a function of the number of vertically cut pixels, behave in a similar way. It can be concluded that the proposed algorithm has a lower performance

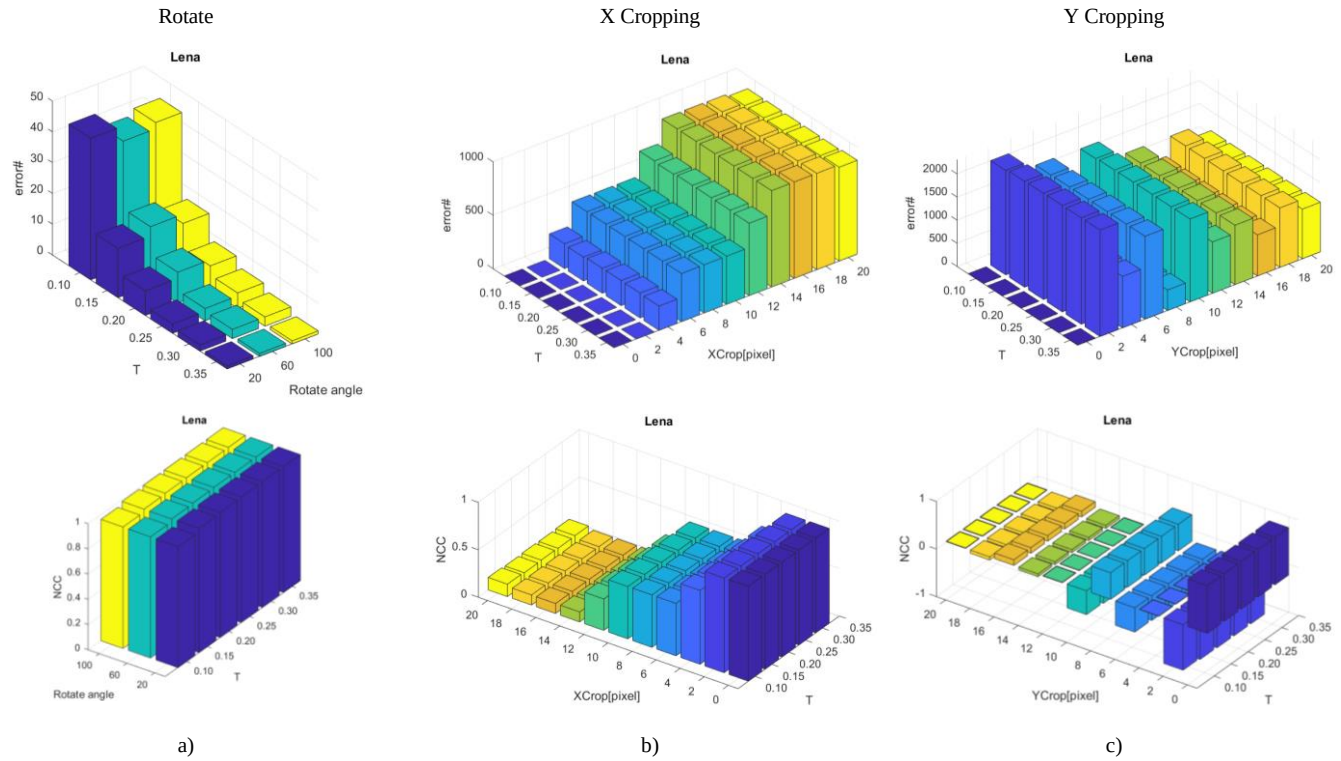

Fig. 3. Number of errors and NCC coefficients in the extracted watermark for image Lena for a) rotation attack as a function of differential threshold T and attack intensity - rotate angle b) horizontal cropping attack as a function of differential threshold T and attack intensity in pixels b) vertical cropping attack as a function of differential threshold T and attack intensity in pixels.

in the case of the attack of cutting in the vertical direction than the attack of cutting the image in the horizontal direction. Graphs related to the geometric attack that is realized by scaling are not shown because no degradation was observed in the extracted watermarks when the scaling factor is in the range from 1 to 2.

VI. CONCLUSION

This paper analyzes the performance of the recommended Binary Watermarking Algorithm for image protection with a watermark in the form of a QR code. The performance of the algorithm is determined in relation to geometric attacks that are realized by rotating, scaling, and cropping the image. To determine the performance of the binary insertion algorithm in detail, both the watermark insertion strength and the attack intensity were varied. Performance was determined using objective parameters for evaluating the quality of the extracted watermark. One objective parameter is the number of bit errors in the extracted watermark, and the other is the determination of the correlation coefficients between the original and the extracted watermark. In addition to the security component that a watermark in the form of a QR code carries, an additional advantage is that it can be decoded from an extracted QR code with errors.

The results of the experiment confirm that the binary watermark insertion algorithm recommended in this paper can be successfully used in the protection of images in case of geometric attacks. Additional security components related to QR code encryption will be discussed in the continuation of the research.

REFERENCES

[1] Cisco Annual Internet Report (2018–2023), White paper, 2020.

[2] Z. Veličković, Z. Milivojević, M. Veličković, „ASP.NET MVC aplikacija za potraživanje i publikovanje podataka COVID-19 Data API-a“, INFOTEH-JAHORINA, pp. 109 - 114, 17-19 March 2021.

[3] https://ethicaljournalismnetwork.org/fake-news-business-democracy

[4] O. Evsutin, K. Dzhanashia, “Watermarking schemes for digital images: Robustness overview”, Signal Processing: Image Communication, Vol. 100, January 2022.

[5] H. Hu, L. Hsu, H. Chou, “An improved SVD-based blind color image watermarking algorithm with mixed modulation incorporated”, Information Sciences, Vol. 519, Pages 161-182, May 2020.

[6] Z. Veličković, D. Blagojević, M. Veličković, „Performance Analysis of Binary Watermarking Algorithm for Color Image Protection Against Nongeometric Attacks“, International Journal of Electrical Engineering and Computing - IJEEC, Vol. 7, No. 1 (2023). DOI 10.7251/IJEEC2301042V.

[7] Z. Veličković, Z. Milivojević, M. Veličković, „Video protection by color watermark using a modified cyclic insertion scheme“, 24th International Conference on Information Technology - IT, pp. 1 - 4, Žabljak, Montenegro, 2020, doi: 10.1109/IT48810.2020.9070430.

[8] D. Rajani, P. Rajesh Kumar, “An optimized blind watermarking scheme based on principal component analysis in redundant discrete wavelet domain”, Sig. Proc. Vol. 172, 2020.

[9] https://www.mdpi.com/1099-4300/24/8/1103 (01.2024.)

[10] C. Patvardhan, P. Kumar, C. V. Lakshmi, “Effective color image watermarking scheme using YCbCr color space and QR code”, Multimed Tools Appl (2018) 77:12655–12677.

[11] https://links.uwaterloo.ca/Repository.html

[12] B. Mahbuba, M. S. Uddin, “Digital Image Watermarking Techniques: A Review”, Journal Information. 11. 110. 10.3390/info11020110. 2020.