Evaluation Tool For Generating Attacks on Watermarking Algorithms

Ahmed Al-Gindy

Al-Dar University College, Dubai
United Arab Emirates

Summary
A new evaluation tool for generating attacks on watermarking algorithms is proposed. The tool is implemented for digital still images but additional attacks for audio or video can be added in the test environment continuously. The implemented tool is comprised of 23 different attacks using a graphical user interface (GUI) to open, select, attack, evaluate, and display the results of both the original and attacked image in ease. It permits user to select any desirable image and perform all aforementioned attacks simultaneously allowing the user to enter or select attack strength parameters. User can also select individual modified images and perform any required changes to the attacked image by adjusting parameters without affecting other attacked images in the scheme. The execution process is performed at a high speed. The tool is much easier for end-users to understand and learn than other traditional tools that need commands to be known or memorized.

Keywords:
Image Processing; Watermarking; Attacks; GUI;

1. Introduction

Digital watermarking technology is now frequently used to embed different sorts of information (information may be composed of a numeric data or figures such as a signature or any form of text) into a digital content to preserve security in the watermarking community. The watermarking community is exposed to illegal violations raising the need for a benchmarking system to carry out an extensive evaluation and test procedures of most watermarking algorithms. It has been possible to determine the robustness or fragility of a watermarking algorithm against attacks via the benchmarking system.
Digital image watermarking, like any new area of research, has many drawbacks and challenges. Every researcher aims to solve some of the problems related to watermarking. Hundreds of points of views and approaches are introduced and submitted to literature with several stories of success. However, none of them uses the same robustness evaluation criteria, this may include but not limited to amount of embedded information, watermark embedding strength, size and nature of the host and the watermark. This is not practical at all for comparison and slows down progress in this area. Benchmarking tools are used to evaluate the robustness of a watermarking technique against attacks. Several tools are popular in the market such as, Checkmark, Optimark, and Stirmark. Checkmark was developed by Shelby Pereira [1]. It is a benchmarking tool for digital watermarking. It can run on Matlab under UNIX and Windows. Optimark is another benchmarking tool for still image watermarking algorithms developed by the Artificial Intelligence and Information Analysis Laboratory at the Department of Informatics, Aristotle University of Thessaloniki, Greece [2]. In November 1997, the first version of Stirmark was introduced as a tool for robustness testing of image watermarking algorithms [3]. Stirmark was developed by Fabien Petitcolas during his Ph.D. at Cambridge University, UK. Stirmark has gained large interest from the watermarking community and it is currently the most widely used benchmarking suite for digital watermarking technologies. Given a watermarked input image, Stirmark generates a number of modified images (attacked images) which can then be used to verify the performance and test if the embedded watermark can still be extracted. Watermark robustness is essential issue for copyright protection [4, 5]. An original digital image can be modified to improve quality, compress data, and so on. Protecting copyrights while maintaining sufficient quality is desirable [4]. There are a number of well-known attacks carried out on images [6]. There are also tools such as StirMark and unZign used to generate watermarking attacks

2. Needs for Evaluation

With a common evaluation tool, watermarking techniques providers would need to present a table of results, to summarize the reliability and performance of the proposed scheme. Therefore, end-users can check whether their basic requirements are satisfied or meet. In addition, researchers can evaluate different algorithms and see how a method can be improved. The industry can also evaluate risks associated to the use of a particular solution by knowing which level of trustworthiness can be achieved by each contender.

3. Some Significant know Attacks on Watermarking

JPEG Compression: JPEG is currently one of the most widely used compression algorithms for images and any watermarking system should be resilient to some degree of compression. In digital images, the original source

Manuscript received June 5, 2016
Manuscript revised June 20, 2016

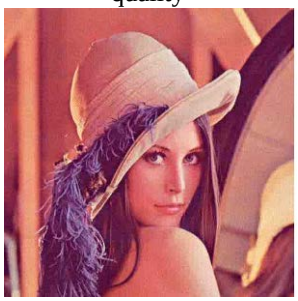
material may be compressed for more efficient storage or transmission. Therefore, it is important to examine whether the proposed watermarking algorithms can survive JPEG compression. The quality rates for JPEG compression can be set to different values. Higher compression ratios yield coarse quantisation for DCT coefficients. Hence, the watermark will be destroyed and become unclear. However, in this situation, the quality of the JPEG compressed image (without being watermarked) will be degraded severely so that the processes of digital watermarking become less meaningful. JPEG compression attacks with different quality are shown in Table I.

TABLE I. JPEG Compression Attack

(a): Original uncompressed watermarked image

(b): Compressed with 75% quality

(c): Compressed with 50% quality

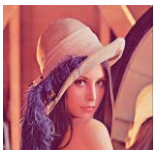


(d): Compressed with 25% quality

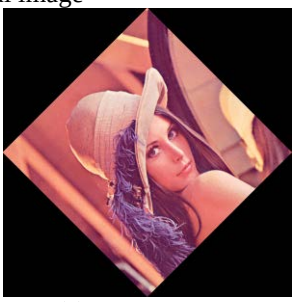
Geometric Distortion: this includes several types of attacks such as flipping, scaling, rotation and cropping. Flipping is usually straightforward to implement, however very few watermarking algorithms do survive it. Rotation is usually combined with cropping or scaling. Usually rotation is the first alignment applied to an image after it has been scanned. Applications that uses image segmentation might involve cropping. In some cases, attackers are interested to crop different parts like central part of an image to remove the copyright information. Applications like web publishing apply scaling to high resolution images. Scaling can be divided into two groups, uniform and non-uniform scaling. Under uniform scaling we understand scaling which is the same in horizontal and vertical direction. Non-uniform scaling uses different scaling factors in horizontal and vertical direction (change of aspect ratio). Very often digital watermarking methods are resilient only to uniform scaling. Geometric attacks do not actually remove the embedded watermark itself, but aim to change the synchronization of the embedded information. The detector would recover the embedded watermark information when perfect synchronization is regained. Different types of geometric distortions are shown in table II.

TABLE II. Geometric Distortion Attacks

(a): Original image



(b): resized image to 256×256



(c): Rotation 45º

(d): Cropping

(e): Affine

Image enhancement operations: digital cameras have been widely used to capture images in digital format. As a result, captured images can be more easily processed. This includes but not limited to filtering, sharpening, histogram modification, gamma correction, color quantization and restoration. Commonly used filters include median, Gaussian, and standard average filter. The contrast of an image is usually adjusted to enhance the subjective quality. Image quality of different contrast enhancement are shown in Table III.

TABLE III. Image Enhancement

(a): Original image

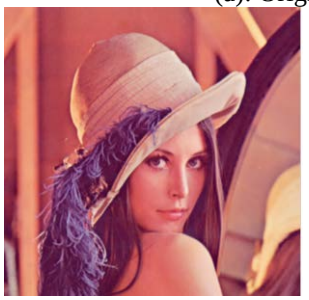
(b): Contrast Adjustment I

(c): Contrast Adjustment II

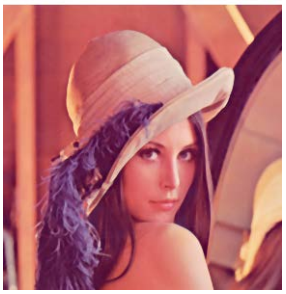
Filtering: The robustness of watermarking algorithms is usually examined against low-pass and median filtering. The watermarked images may still being recognizable undergoing filtering levels of 3×3 mask size, but higher levels of filtering using 5×5 mask size, would spoil the quality of the watermarked image which in result spoil the watermark itself. Low-Pass and Median filtered images are shown in table IV.

TABLE IV. Filtering Attack

(a): Original image



(b): 3×3 Low-pass filtered image



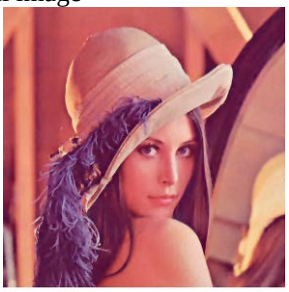
(c): 3×3 Median filtered image

Removal attacks: aim at the complete removal of the watermark information from the watermarked data. Weiner filter try to severely impair the embedded watermark while maintaining the quality of the attacked image.

TABLE V. Removal Attack

(a): Original image

(b): 3×3 Wiener filtered image



(c): 5×5 Wiener filtered image

Additive Noise: Some watermarking techniques are introducing high frequency noise. Sharpening filter can be an effective tool of detecting high frequency noise. histogram modification, gamma correction, color quantization and restoration are applied respectively on the condition like poor lighting conditions, scanning an image for web publishing, converting an image for GIF format, restoration after degradation process. All the image enhancements attacks can be used to test the robustness of a watermarked image without prior knowledge of the noise introduced by the watermarking algorithm. Additive noise can result from certain applications such as the use of A/D and D/A converters or from transmission errors. Authors often claim that their copyright marking techniques survive this kind of attack, but many forget to mention the maximum level of acceptable noise that can be handled by these techniques [7]. Examples of additive noise are shown in table VI.

TABLE VI. Additive Noise Attack

(a): Original image

(b): Gaussian noise

(c): Salt & pepper noise

The Proposed Attackmark System

The introduced benchmark tool is referred to as the Attackmark. The Attackmark is comprised of 23 different attacks; namely Gaussian Filter, Average Filter, Median Filter, Wiener Filter, Disk Filter, Laplacian, Motion, Prewitt, Sobel, Unsharp Filter, Dither, Salt and Pepper, Speckle, Gaussian Noise, Resize, JPEG Compression, Rotation, Color Enhancement, Cropping, Cutting, Translation, Affine Transform, and Scaling & Translation. The proposed benchmark is implemented using a graphical user interface (GUI) to open, select, attack, evaluate and display the results of both the original image and attacked/modified image. The implemented GUI is much easier than other traditional tools for end-users, since it doesn't require memorization of numerous programming or syntax commands.

The Stirmark system was limited in its potential for generating attacks on watermarking algorithms because several attacks were applied to the host image at one go, using discrete or default parameters. However, in Attackmark tool the user can tailor the attacks to their choice overcoming the limitations in the Stirmark tool. The user can also blend a range of attacks in succession to a single host image. Stirmark can be compiled using Microsoft visual studio express which might be difficult to integrate with the watermarking algorithms as most of the researchers using MATLAB to implement their watermarking techniques. On the other hand, Attackmark tool implemented in MATLAB which make it easily to be used as a part of a programmed watermarking algorithm.

Furthermore, Attackmark has extra features such as; viewing image's original size, display the RGB and x-y coordinates of any image by pointing the cursor anywhere on the image, users also have the option to refresh all the axis along with the input fields to load a new image, finally user can save all the images in a folder automatically created in the specific folder named "Attackmark" when the "Save-All" button is clicked. Individual images can be saved through the push button positioned next to each image.

4. Attackmark features Description

Attackmark user friendly interface provides users with ease to evaluate any watermarking algorithm. User can apply different sorts of default attacks or by adjusting their parameters. Attackmark factor dashboard is a panel containing input fields or pull-down menus for all the implemented attacks. There are 24 axis in the menu, each axis corresponds to a type of attack applied to the host image; the first axis represents the host image where the user can load the selected image and then apply different attacks by first filling then execute the "Attackmark factor panel" as shown in Figure 1. All attacks will be applied on the host image and attacked images will appear in their respective axis.

User can use the default or standard attack parameters without having to utilize the Attackmark factor panel via the Standard key. After selecting the host image, the Standard key is pressed applying all attacks to the host using set parameters. The modified images will appear in their respective axis. The user can choose a particular attack by ticking the radio button of the axis corresponding to that particular attacked image.

Attackmark allows the user to amend any attacked image after the execution process is complete without filling in the Attackmark factor panel again and without altering the rest of the modified images by selecting that particular attack from "Amendments to specific attack" menu. Once the specific attack is chosen and attack parameters set, only that particular axis will transform and others axis remain unchanged.

Another important feature in Attackmark permits users to blend multiple attacks or all attacks in succession onto a single image resulting in an image enormously attacked. Ticking radio buttons over each axis displays a separate GUI menu from which the user can select an image, choose any attack and then save the attacked image. User then clicks on "Choose another attack" button repeating same process on the same image. The same procedure can be carried out to blend many more attacks to the same image.

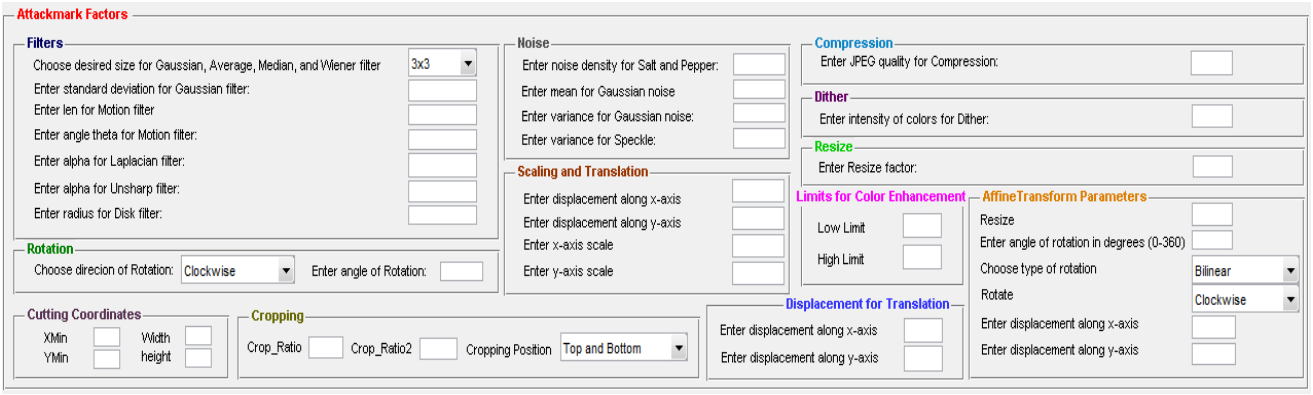


Figure 1. Attackmark Factor Panel

5. Attacks on Attackmark

- Filters panel contains parameters of various filters, namely; Gaussian, Average, Median, Wiener, Motion, Laplacian, Unsharp and Disk.
- Rotation attack may demolish the synchronization of the watermark detector and the watermarking. Even a slight rotating effect can nullify the detection of watermarks. However, from the Rotation panel, the user can choose the direction of rotation and enter the desired angle of rotation in the provided input field.
- Additive Noise can result from certain applications such as the use of A/D and D/A converters or from transmission errors [8]. The Noise panel includes three noise attacks, namely; Salt and Pepper, Gaussian Noise and Speckle.
- Affine attack is a combination of three distinct attacks, specifically; Rotation, translation and Resize Attack.
- JPEG Compression Attack, JPEG Joint Photographic Experts Group, is the most common image data compression standard [9], handles grey-scale and colour images of different resolutions. In digital images, the original source material may be compressed for more efficient storage or transmission. The quality rates for JPEG compression attack can be set to different values.
- In the colour enhancement panel, the user can input the lower and upper limits of Color Enhancement attack in their corresponding fields. Color Enhancement plays a big role in controlling the contrast and the brightness of the image. The lower limit is used to change the brightness, whereas the upper limit is responsible for altering the contrast.
- Dither attack involves increasing the apparent color resolution of the any/or watermarked image.
- Resizing attack reduces accuracy detection of watermarks [4]. This attack does not actually remove the embedded watermark itself, but aim to change the synchronization of the embedded information. To test the robustness of any watermarking algorithms against resizing attacks, different images have been resized to different scales.
- Scaling and Translation panel consists of all the parameters responsible for scaling and translating the image simultaneously. Cutting attack is more or less similar to cropping but is achieved without zero padding; instead the image will be truly cut by eliminating pixels. In translation attack the image is shifted in coordinate space by adding a specified value to the x- and y- coordinates. Cropping is another way to detach the desired portion can be achieved by covering the unwanted region with black bands.

6. Performance Evaluation and Results

This section demonstrates some obtained results from the proposed attackmark tool. Figure 2 illustrates the original Lena image being attacked by all aforementioned attacks, using parameters inserted by user in the Attackmark factor panel; whereas, Figure 3 shows the original Lena image being attacked using default parameters. Pressing the standard pushbutton applies all the attacks on the loaded image using default parameters. Figure 2 and Figure 3 show the advantages of the Attackmark system over the Stirmark tool. The blending of attacks was successfully tested on Lena image.

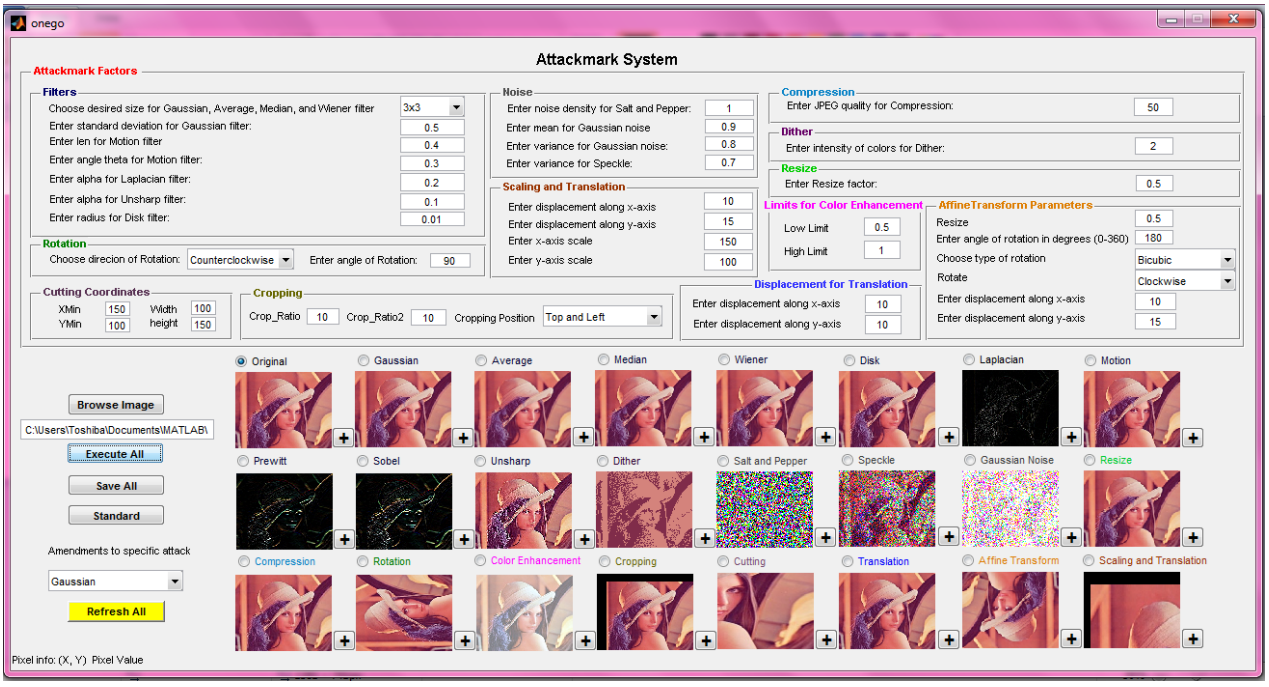


Figure 2. User defined parameters result.

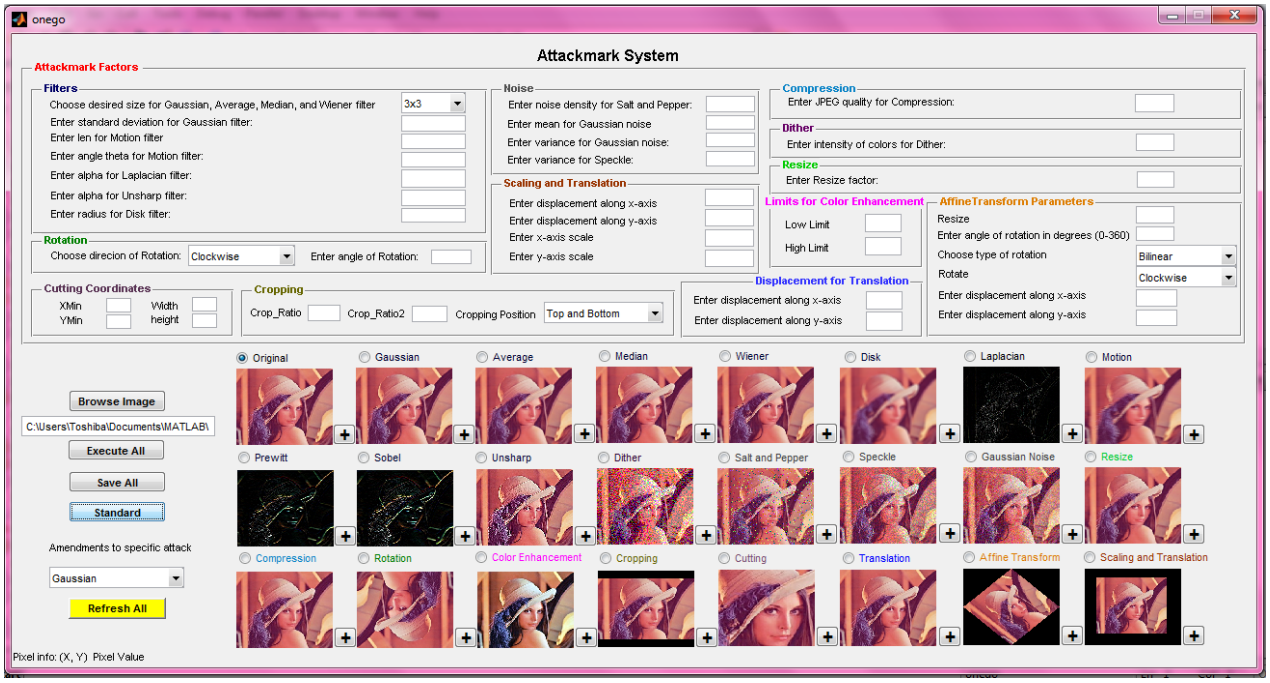


Figure 3. Defualt parameters result.

7. Conclusion

The proposed benchmark system, consisting of 23 attacks, was mainly implemented for the purpose of evaluating and testing any watermarking technique and its robustness and/or fragility against attacks. Various features has been added to the GUIs, to make them more comprehensible. This includes the option to inflict all attacks, using default and/or desired parameters, simultaneously, on the watermarked image. The implemented attackmark system is more advanced than other available systems such as

Stirmark system, as it grants control over the selection of attack parameters, unlike the latter, which uses only default parameters. However, based on the preceding results, the system is very reliable to be tested on any watermarking algorithm/technique. More attacks such as audio attacks and video Attacks can be integrated easily to the proposed system.

References

[1] Shelby Pereira, S. Voloshynovskiy, M. Madueño, S. Marchand-Maillet, and T. Pun, "Second generation benchmarking and application oriented evaluation," in In Information Hiding Workshop III, Pittsburgh, PA, USA, 2001.

[2] V. Solachidis, A. Tefas, N. Nikolaidis, S. Tsekeridou, A. Nikolaidis, and I.Pitas, "A benchmarking protocol for watermarking methods," in IEEE Int. Conf. on Image Processing (ICIP'01), Thessaloniki, Greece, October, 2001 pp. 1023-1026.

[3] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in David Aucsmith (Ed), Information Hiding, Second International Workshop, Portland, Oregon, U.S.A., 1998, pp. 219-239.

[4] G. Voyatzis and I. Pitas, "Protecting Digital Image Copyrights: A Frame work," in IEEE Computer Graphics and Applications, 1999, pp. 18-24.

[5] I. J. Cox, J. Kilian, F. Leighton, and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia," in IEEE Transactions on Image Processing, 1997, pp. 1673 -1687.

[6] S. Voloshynovskiy, S. Pereira, and T. Pun, "Attacks on Digital Watermarks: Classification, Estimation-Based Attacks, and Benchmarks",," in IEEE Communications Magazine, 2001, pp. 118-26.

[7] M. Kutter and F. A. P. Petitcolas, "A Fair benchmark for image watermarking systems " in Electronic Imaging '99. Security and Watermarking of Multimedia Contents, The international Society for optical Engineering, , Sans Jose, USA, 1999

[8] A. M. B. Sewaif, "Digital Image Watermarking Using Walsh Coded Handwritten Signatures," in Department of Electronic Engineering. vol. M.Sc. Sharjah: Etisalat University College, 2005, p. 111.

[9] M. Prasad and S. Koliwad, "A Comprehensive Survey of Contemporary Researches in Watermarking for Copyright Protection of Digital Images," in International Journal of Computer Science and Network Security IJCSNS, April 2009, pp. 91-107.