BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 59, No. 1, 2011
DOI: 10.2478/v10175-011-0004-3

Watermarking software in practical applications

P. LIPIŃSKI*

Institute of Information Technology, Technical University of Lodz, 215 Wólczańska St., 90-924 Łódź, Poland

Abstract. In the recent years, several digital watermarking applications have been developed for copyright protection of digital images. In this article we have tested how they perform in practical applications. We have identified the most common operations performed by professional photographers and web developers, and tested the robustness of watermarks embedded using the applications for copyright watermarking. Our aim was to prove that commercially available software does not meet the requirements of photography and web industry. We have also identified areas in which the software should be improved in order to meet current and future requirements of the industry.

Key words: watermarking, digital watermarking, robust watermarking, steganography.

1. Introduction

In today's age of the Internet, most information, documents etc. are stored and processed in digital form on a computer. As a result, digital content can be easily copied and illegally distributed over the Internet. Hence, there is a strong need of developing the techniques which can help to identify intellectual property of digital documents, see [1]. Digital Watermarking has emerged as a solution for protecting the intellectual property of digital data [1–4].

Digital watermarking is the process of embedding information into digital data, such as files, images, texts, audio or video. It can be divided into two categories: visible watermarking (when the information is visible or audible in the picture, video or sound) and invisible watermarking (information is added as digital data, but it cannot be perceived as such; it is possible to detect hidden information performing some operations on watermarked data) [5]. The existence of the digital watermark can prove an author's ownership of some original data, or trace the illegal use of the data. The author's identity or ownership rights can be protected. Unfortunately, even small manipulations performed on digitally watermarked data can destroy watermark. This leads us to the topic of attacks, that is operations which result in removing digital watermark from digitally watermarked data. Robust watermarks are designed to survive some attacks performed on digitally watermarked content. Still, the question to be answered is what operations should watermarks survive and how to design reliable test for watermark testing [6]. Here we have developed our own test basing on survey results and compared the robustness of the digital watermarks which can be added to digital images using available commercial software for digital images watermarking. We focus only on those applications, which are able to embed digital watermark in the form of text into digital images, because such watermarks can easily be processed using automated tools.

2. Digital watermarking

In digital watermarking, the information to be embedded is called *digital watermark*. The data where the watermark is to be embedded is called *host data*. For the purpose of this publication we use a simplified model of watermarking shown in Fig. 1. We consider three distinct operations in a watermarking system: embed, attack and extract. Embedding means changing some parameters of *host data* in order to hide *digital watermark* in the *host data*. As a result of embedding we obtain *watermark data*. The *digital watermark* should be imperceptible in the *watermarked data*, as we focus on invisible watermarking here. A watermark is called imperceptible if the *host data* and *watermarked data* are indistinguishable with respect to an appropriate perceptual metric. The *watermarked data* is usually transmitted to another user. Modifications of the watermarked image made by the user, is called *attack*. As a result of the *attack* we obtain *attacked data*. The *attacks* can be intentional (when someone wants to remove the watermark from an image) or unintentional (as a result of lossy compression, cropping, rotating, etc.). Detection (also called extraction) is an algorithm which is applied to the *attacked data* to extract the watermark from it. If the signal is unmodified (has not been attacked), the *digital watermark* can easily be extracted from the *watermarked data*, but when *watermarked data* has been attacked, extraction is more difficult, sometimes even impossible. In robust watermarking, the extraction algorithm should be able to correctly reproduce the *digital watermark* from the *attacked data*. If the *digital watermark* can be extracted after more severe modifications (attack) we call the system more robust. There is a trade-off between robustness and imperceptibility and it is challenging to create a watermark which is robust and imperceptible at the same time [1].

There are two possible sorts of extraction algorithms: blind and non-blind. The extraction algorithm is blind when *host data* is not required in order to extract the *digital watermark*. In non blind extraction algorithm the original host

*e-mail: piotr.kazimierz.lipinski@gmail.com

data is necessary to extract the watermark. Non-blind watermarking is inconvenient for automated detection of watermark when the number of images is large; therefore it is desired to use blind watermarking when processing large numbers of images.

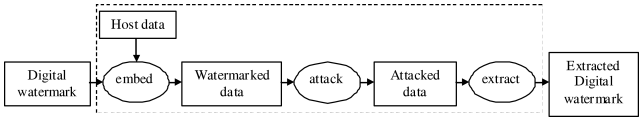

Fig. 1. The model of digital watermarkind system

The length of the *digital watermark* (number of bits) determines two different classes of watermarking schemes: zero-bit long (sometimes called one-bit watermark) which is designed in order to detect the presence or absence of the watermark in the watermarked signal, and n-bit long (sometimes called multiple bit watermarking or non zero-bit watermarking) in which the n-bit stream is modulated in the *digital watermark*. Here we focus on n-bit long, invisible, blind, robust watermarking of images.

3. Watermarking attacks

We assume that digital watermarking software should at least meet the requirements of professionals, such as photographers and web developers. Therefore, we have conducted a survey. In the survey we asked 25 web developers and 25 professional photographers to indicate modifications they usually perform on digital images. We had made a list of all possible image modifications which can be performed using professional photo editing software (Photoshop) and asked them to indicate the most frequently used ones. The results of the survey are given in Table 1.

Table 1
The operations the most frequently performed on pictures (the survey results)

Operation	Range
Scaling	8–1000%
Streching	±20%
Rotation	0–180°
Cropping	30% of the original image
Jpeg compression ratio	70–100%
Color depth reduction	8–48 bit

Table 1 indicates that digital pictures are processed using many different algorithms. The modifications are performed from various reasons, for example, scaling is frequently used to reduce image size when exporting the image from the professional camera (21 mega pixels 5616×3744) to the internet (typical image published in picture agencies 450×300). Stretching is often used to make people look slimmer, rotation is used in "paparazzi pictures" or sport pictures. Photographers crop images to fit them to standard photo sizes (which is usually more than 30% of the original image), while web developers may sometimes need only small elements from a large image (1% of the image). Brightness, contrast, sharpening, blurring, noise are used to enhance image quality and usually do not exceed 20%–30%. Jpeg compression ratio and color depth is changed in order to reduce the image file size, especially when images are to be published in the Internet. Color depth is also used as a result of hardware or software limitations. Professional photo camera can use up to 48 bit color depth, while software can usually operate only on 24 bit color depth, and printers can print only 8 bit colors. Lower color depth are not used any more. According to our survey, all abovementioned operations are performed very often, therefore, ideal watermark should survive all operations from Table 1. Basing on the survey results we have selected test operations. The attacks which were selected to perform tests are given in

Table 2
Test operations selected based on survey results

Operation	Range
resize	10%, 25%, 50%, 75%, 150%, 200%, 300%
rotation	1°, 2°,5°,10°,45°,90°,180°
crop center	10%, 25%, 50%, 75%
crop non center	10%, 25%, 50%, 75%
jpeg compression	10%, 25%, 50%
jpeg 2000 compression	10%, 25%, 50%, 75%
color depth conversion to	16 bit, 8 bit, 7 bit, 6 bit, 5 bit, 4 bit, 3 bit, 2 bit
file conversion	raw, jpeg, tiff, jp2
printscreen	
print-> scan	

We have also inquired the professionals, which standards they use in digital imaging. The results were quite surprising because only three standard were indicated in all surveys: jpg, tiff and raw. Which is more surprising such forthcoming standards as jpeg 2000 were completely unknown. Nevetheless, we have included jpeg 2000 in our tests because we expect this standard to be more popular in the future.

Color depth conversion has been extended when compared to survey results just because of curiosity. We were saving files using 16 bit, 8 bit, 7 bit, 6 bit, 5 bit, 4 bit, 3 bit, 2 bit color depths. We have also added print screen and print scan tests to test the robustness of the watermarking applications against geometrical distortions. In print screen test we were making print screen of the watermarked image, cropping the image to its original size and testing if it was possible to read the watermark. In print-scan test the image was printed using Canon i4000 printer in high quality on high quality paper and than the image was scanned using HP Scanjet G2710 scanner, cropped to its original size and the existence of the watermark was tested.

The digital image test set includes 212 well known test pictures available in The USC-SIPI database (http://sipi.usc.edu/database/). Picture sizes ranged from 256×256 pixels to 1024×1024 pixels, 16 bit color or 8 bit gray. Pictures were selected from textures, aerials, miscellaneus and sequences. They reflect various picture characteristics (color, frequency, sharpness etc.). The sample selected

digital test images are shown in Fig. 2 in grayscale due to limitations of this publication. Tests were performed on color images as well.

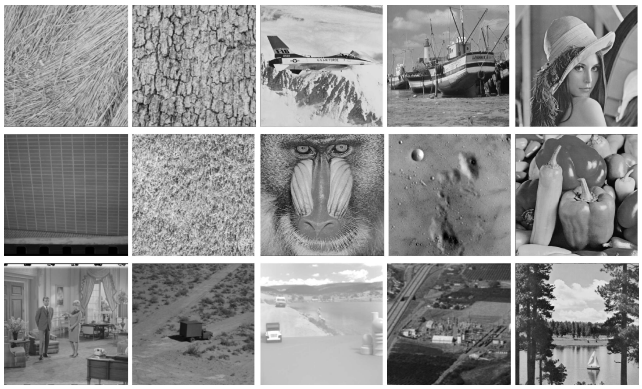

Fig. 2. Sample test pictures

4. Watermarking software

We have performed a detailed search of watermarking software available in the Internet. There is a variety of software which can embed hidden information in digital data, see Steganography Tools (http://www.jjtc.com/Security/stegtools.htm). This software can serve many purposes, like secret information hiding, transmission security etc. We limit our research to those tools which are capable marking images for copyright. The software which embed digital images into digital images is also beyond our interest, because image watermarks are difficult to trace and identify using automated tools. We limit our research to the software which can embed ascii text into the digital images. We have tested the applications given in Table 3.

Table 3
The software which was used to embed watermarks. The robustness of watermarks tampered using this software is discussed in this article

Data Stash
Digimark
Eikonamark
Icemark
JPHS
SignMyImage
StegHide
Tajnopis

Applications from Table 3 are capable of hiding text into the digital image, but not all of them are designed for copyright protection and therefore they show very diversified performance. Data Stash is a stegaongraphic security tool, which allows to hide sensitive data files within other files, which means that it allows hiding test in digital images (see http://www.skyjuicesoftware.com/software/ds_info.html). The software is designed to hide data in picture files rather than for copyright marking. Is easy to use, but very fragile to any watermarked picture modifications. Digimarc is a very powerful steganographic software developed by Digimarc Corporation which develops means to identify all forms of content (see https://www.digimarc.com/). Digimarc Corporation is very active in scientific field and owns several digital watermarking patents [8–13]. Another very powerful software has been developed by Alpha Tec LDT. Alpha Tec LDT is an research and development company specialized in digital image and video processing and multimedia. It has several research and development products including Eikonamark – the software for casting invisible watermarks on digital images and detecting these watermarks, which can be found here: http://www.alphatecltd.com/index.html. Yet another software has been developed by Prohibit Software. It is called Icemark (http://www.phibit.com/). It can be used for protecting photos, artwork or other types of images from illegal use. Offering an advanced watermarking technology, Icemark allows to embed an invisible watermark which can be detected when image is scanned by Icemark. It does not perform as well as Eikonamark and Digimarc, but still passes the majority of tests performed. JPHS is a software developed by Allan Latham which allows to hide a file in a jpeg visual image, see: http://linux01.gwdg.de/∼alatham/stego.html. It is quite old and is instable on contemporary systems. It has failed even simple tests. Filip Krolupper has developed software called SignMyImage (http://www.adptools.com/) which passed only very basic filetype conversion tests. StegHide, available at http://steghide.sourceforge.net/development.php, distributed on GNU license and developed on open bases, performed even worse, but this is probably due to the fact that it has been developed to hide information rather than to copyright protection. Tajnopis is the only polish steganography tool which we were able to find in the Internet. It has been developed by Marcin Dutkiewicz. It performs well as an information hiding software, but fails almost all test performed here: http://m_dutkiewicz.republika.pl/. The reason is probably the same as in StegHide.

5. Software comparison

Here we describe the test procedure which was performed in order to verify robustness of digital watermarking applications from Table 3 against attacks from Table 2. Firstly, each test image was watermarked using each program from Table 3. This resulted in 1696 watermarked images. Secondly, each of the watermarked image was attacked (modified) using each of operations from Table 2. This gave about 80000 attacked, watermarked images after 49 different attacks (notice that not all combinations are possible due to watermarking software shortcomings). Thirdly, we used suitable applications to read watermarks from modified, watermarked images. If the application was able to read the watermark from all 212 images we assumed it passed the test.

The results of the test are shown in Table 4. Attacks are given in the first column, digital watermarking software is shown in the first row. Letter “P” means that it was possible to read the watermark, F – denotes that it was impossible. P/F – mans that watermark existence was detected, but it was impossible to read the complete watermark.

Table 4
Digital watermarking software test results. P – means "pass" – it was possible to read watermark; F – means "fail" – it was impossible to read the watermark. P/F – mans that watermark existence was detected, but it was impossible to read a complete watermark

Test	Icemark	Digimarc	Tajnopis	Eikonamark	StegMark	steghide	SignMyImge	JPHS	Data Stash
Original image	P	P	P	P	P	P	P	P	P
Resize 0.1	F	F	F	F	F	F	F	F	F
Resize 0.25	F	F	F	F	F	F	F	F	F
Resize 0.5	F	F	F	F	F	F	F	F	F
Resize 0.75	F	P	F	F	F	F	F	F	F
Resize 1.5	F	P	F	F	F	F	F	F	F
Resize 2	F	P	F	F	F	F	F	F	F
Resize 3	F	P	F	F	F	F	F	F	F
Rotate 1°	F	P	F	F	F	F	F	F	F
Rotate 2°	F	P	F	F	F	F	F	F	F
Rotate 5°	F	P	F	F	F	F	F	F	F
Rotate 10°	F	P	F	F	F	F	F	F	F
Rotate 45°	F	P	F	F	F	F	F	F	F
Crop center 0.75	F	P	F	F	P/F	F	F	F	F
Crop center 0.5	F	P	F	F	P/F	F	F	F	F
Crop center 0.25	F	F	F	F	F	F	F	F	F
Crop center 0.1	F	F	F	F	F	F	F	F	F
Crop non center 0.75	F	P	F	F	P	F	F	F	F
Crop non center 0.50	F	P	F	F	F	F	F	F	F
Crop non center 0.25	F	F	F	F	F	F	F	F	F
Crop non center 0.10	F	F	F	F	F	F	F	F	F
Jpeg compression 0.75	P	P	F	P	P	F	F	F	F
Jpeg compression 0.5	F	P	F	P	P	F	F	F	F
Jpeg compression 0.25	F	P	F	F	F	F	F	F	F
Jpeg compression 0.1	F	F	F	F	F	F	F	F	F
Jp2 compression 0.5	P	P	F	P	P	F	P	F	F
Jp2 compression 0.25	P	P	F	P	P	F	P	F	F
Jp2 compression 0.1	F	P	F	P	P	F	F	F	F
Color depth 256->128	P	P	F	F/P	P	F	F	F	F
Color depth 256->64	P	P	F	F/P	P	F	F	F	F
Color depth 256->32	P	P	F	F/P	F	F	F	F	F
Color depth 256->16	P	P	F	F/P	F	F	F	F	F
Color depth 256->8	P	P	F	F/P	F	F	F	F	F
Color depth 256->4	F	P	F	F/P	F	F	F	F	F
Color depth 256->2	F	P	F	F	F	F	F	F	F
Color depth 256->Gray	P	P	F		P	F	F	F	F
Conversion JPG-> tiff	–	P	–	P	–	F	P	–	–
Conversion JPG -> bmp	–	–	–	–	P	F	P	–	–
Conversion tif->JPG	P	–	–		–	F	–	–	–
Conversion tif-> bmp	P	–	–		–	F	–	–	–
Conversion tif-> gif	P	–	–		–	F	–	–	–
Conversion jpg-> jp2-> jpg	–	P	–	P	P	F	P	F	F
Conversion jpg-> bmp-> jpg	–	P	–	P	P	F	-	F	F
Conversion jpg-> gif-> jpg	–	P	–	P	P	F	P	F	F
Conversion tif-> jp2	P		–	–	–	F	–	F	F
Conversion bmp-> jpg-> bmp			F	–	–	F	–	F	F
Conversion bmp-> jp2->bmp			F	–	–	F	–	F	F
Conversion bmp-> gif->bmp			F	–	–	F	–	F	F
Printscreen	F	F	F	P	F	F	F	F	F
Print->scan	F	F	F	F	F	F	F	F	F

6. Discussion of results

It is clearly noticeable that none of the software tested here passed all of tests. According to our tests Digimarc performed definitely best. It was the only software which passed 4 resize tests and 4 crop tests, all file format conversion tests and color depth tests. In our opinion this is the best software on the market. According to our tests the second best software is Eikonamark, which passed 2 jpeg compression tests and partially passed crop tests. In crop tests it was able to detect the existence of the watermark, but was unable to read it. Eikonamark, as the only software passed print screen test. The third best software in our test was Icemark. It passed 2 jpeg2000 compression tests, 6 color depth tests, 4 file compression tests and one jpeg compression test. The remaining software failed almost all tests and should not be used for copyright watermarking of images because the watermarks embedded using this software are very immune to attacks.

7. Conclusions

We have demonstrated in this paper that the all copyright marking applications available in the Internet are vulnerable to attacks involving the most frequently used operations performed by professional photographers and web developers. Even Digimarc, the most powerful watermarking software according to our tests, fails almost half of tests performed. This means, there is still a need to develop new, more robust algorithms for copyright protection. According to our survey, the industry is highly interested in digital watermarking software, but robustness of watermarks to cropping and rotation should be improved to make the digital watermarking technology more popular.

REFERENCES

[1] S. Decker, "Engineering considerations in commercial watermarking", *IEEE Communications Magazine* 4, 128–133 (2001).

[2] I.J. Cox, M.L. Miller, and J.A. Bloom, *Digital Watermarking*, Morgan Kaufmann, Berlin, 2001.

[3] G. Bhatnagar and B. Raman, "A new robust reference watermarking scheme based on DWT-SVD", *Computer Standards & Interfaces* 31, 2002–1013 (2009).

[4] W.H. Lin, Y.R. Wang, and S.J. Horng, "A wavelet-tree-based watermarking method using distance vector of binary cluster", *Expert Systems with Applications* 36, 9869–9878 (2009)

[5] I.J. Cox, M.L. Miller, J.A. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, Morgan Kaufman Publishers, Berlin, 2008.

[6] T.F. Rodriguez and D.A. Cushman, "Optimized selection of benchmark test parameters for image watermark algorithms based on Taguchi methods and corresponding influence on design decisions for real world applications", *Security and Watermarking of Multimedia Contents* 5, 215–228 (2003).

[7] D. Aucsmith, "Attacks on copyright marking systems", *Lecture Notes in Computer Science* 1, 218–222 (1998).

[8] J. Tian, "Reversible watermarking by difference expansion", *Proc. Workshop on Multimedia and Security: Authentication, Secrecy, and Steganalysis* 1, 19–22 (2002).

[9] A.M. Alattar, "Reversible watermark using difference expansion of quads", *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing ICASSP* 29, 377–380 (2004).

[10] A.M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform", *IEEE Trans. on Image Processing* 1 (8), 1147–1156 (2004).

[11] J. Stach and A.M. Alattar, "Security, stenography, and watermarking of multimedia contents", *SPIE Conf.* 1 (6), 386–396 (2004).

[12] H. Brunk, "Host-aware spread spectrum watermark embedding techniques", *Security and Watermarking of Multimedia Contents Conf.* 5020 (5), 699–707 (2003).

[13] A.M. Alattar, "Smart images using Digimarc's watermarking technology", *12^{th} Int. Symposium on Electronic Imaging.* 3971 (25), CD-ROM (2000).