A Practical, Almost Zero-knowledge Watermark Verification Algorithm

Priscilla Lee*, Mohammed Awad†, Ernst L. Leiss‡
*Dept. of Computer Science, Wellesley College, Wellesley, MA 02481, USA
plee3@wellesley.edu
†Dept. of Computer Science and Engineering, American University of Ras Al Khaimah,
Ras Al Khaimah, 10021, UAE
mohammed.awad@aurak.ac.ae
‡Dept. of Computer Science, University of Houston, Houston, TX 77204, USA
coscel@cs.uh.edu

***Abstract*—Digital watermarks embed hidden information directly in media content, such as audio, video and images, typically in a way that is imperceptible to a human viewer. This embedded information can be used to handle copyright control, verify or authenticate, and establish ownership. It is therefore important that these watermarks be invisible, robust, and private to guard against attacks. Unfortunately, once a watermark is employed as evidence of ownership, enough information about the watermark may have been disclosed, rendering it vulnerable to removal attacks. Thus, the idea of zero-knowledge watermarking, of verifying the presence of a mark without revealing information about it, helps to address this issue. In this paper, we discuss an alternative, more practical approach to the complicated zero-knowledge watermarking protocols that have been proposed. We also analyze the probabilities of successfully cheating the system, and discuss the use of a black box detector tool.**

Keywords: watermarks, JPEG compression, zero-knowledge

I. INTRODUCTION

A. Motivation

The existence of digital forms of copyrighted material such as videos, images, and audio, and the potential to illegally produce an unlimited number of perfect copies pose a threat to the rights of content owners. Digital watermarking attempts to address this problem by imperceptibly embedding information in the content itself, which can serve later as proof of ownership or authenticity.

Digital watermarks come in many forms: visible or invisible, fragile or robust, and public or private [1]. For the purposes of this paper, we will focus specifically on invisible, robust, and private watermarks. Invisible, sometimes referred to as imperceptible, watermarks are ideal because they manipulate the actual content without marring or distorting the original media by making changes that are undetectable to the naked human eye. Robust watermarks are able to withstand standard processing operations, such as rescaling, cropping, resizing, and filtering, as well as malicious attacks, such attempts to remove or overwrite the mark. Private watermarks, marks embedded and detected only with a certain secret key, ensure that only the legitimate creator of media can demonstrate ownership conclusively by protecting the watermark's integrity and discouraging attempts of fraud or removal [2,3].

B. JPEG Compression

In order to understand how watermarks are embedded, it is important to first understand how the original digital media is stored. We will briefly sketch the basic technique of JPEG compression [4]. JPEG compression is a type of lossy compression, which means the final compressed file loses some of its original information and the file size can be dramatically reduced. The secret to lossy compression is that it throws away information that is imperceptible to the human eye and will not be missed by the viewer. It achieves this effect through a series of transforms and manipulations.

JPEG compression relies on the fact that the human eye perceives color much less than light intensity, and lower frequencies much better than higher frequencies. At the start of the compression process, digital images are transformed from the traditional RGB color space into the YC_bC_r model, which separates luminance from chrominance. Since the human eye cannot detect minor changes in color, the chrominance component can be much more significantly reduced than the luminance component without perceptibly affecting the overall quality of the image. The image is then decomposed into blocks of 8 x 8 pixels, and the Discrete Cosine Transform (DCT) is applied to each of these matrices. The DCT generates a new 8 x 8 block of pixels that contains coefficients of increasing spatial frequency. These 64 DCT coefficients are quantized into a table of integers and rounded off so that many of the high-frequency coefficients go to zero. What results is a table that is ordered from low-frequency coefficients, which are more important in perception, to high frequency coefficients, which can be discarded. These integers are encoded using run-length and Huffman coding and then stored [4].

Knowing how JPEG files are stored, it is clear that we should not embed our watermark in material that will be thrown away by data compression. Instead, it is important to insert our watermark in the lower-frequency components of each of the image's 8 x 8 blocks. This should guarantee the security of our watermark. Usually, the same watermark is inserted into every single one of an image's 8 x 8 blocks, and this spatial redundancy allows it to resist some attacks such as cropping [5].

Existing watermarking schemes are able to successfully embed information in digital media and withstand common operations and attacks. Digital watermarks are continuing to develop to protect ownership and copyright control, but potential attackers still pose a threat to seemingly private watermarks.

II. ZERO-KNOWLEDGE PROTOCOLS

Watermarks are often private, embedded and detected with a secret key, in order to protect against malicious attempts at fraud or removal. Once a watermark is utilized to resolve ownership in a court of law, however, the mark must be revealed and it is then susceptible to attacks. Such watermarks are actually not private at all, for they must ultimately be made public when employed [6]. In order for digital watermarks to be truly useful, one must be able to demonstrate its presence without revealing its information. The idea of zero-knowledge watermark verification, of detecting a mark without revealing the mark itself or any other compromising information, can help to address this issue.

A zero-knowledge proof is a method by which one party (the prover) can demonstrate knowledge of some fact to another party (the verifier) without revealing any additional information or knowledge. General zero-knowledge protocols exist that are based on graph isomorphism or discrete logarithms [7], and others specifically meant for application to digital watermarking have also been proposed, such as protocols based on the Pitas scheme [8], hard problems, and invertibility attacks [6].

However, many zero-knowledge protocols are impractically complicated. In this paper, we introduce a simpler, alternative approach to currently existing zero-knowledge watermarking protocols.

III. AN ALTERNATIVE, MORE PRACTICAL APPROACH

A. The General Approach: Randomly Checking n Number of Blocks

The purpose of applying zero-knowledge protocols to watermarking is to outline a method by which one may verify ownership in a court setting by demonstrating the presence of a watermark without revealing it. We will attempt to achieve this goal without the unnecessarily complicated zero-knowledge protocols that exist.

For this paper, we assume that the content being watermarked is a JPEG image, though in principle such a watermarking technique is applicable to other multimedia files.

Let's imagine a watermarking scheme in which different watermarks with different keys are embedded in each of the 8 x 8 pixel blocks of a given image, let's say a medium image, of size 480 x 640 pixels, and its legitimate owner has the keys to each of these individual watermarks. In order to prove his/her ownership of this image, an arbitrary n number of blocks out of the 4800 available may be randomly selected for the watermark to be demonstrated and revealed. The randomness guarantees a strong degree of assurance that the entire image must then belong to the claimed owner, and only a small percentage of the watermark has been revealed, which doesn't grant attackers enough information to easily remove the whole watermark.

This approach addresses many of the issues that current zero-knowledge protocols attempt to resolve, but in a much simpler and cleaner way. It is important to note, however, that if this particular image or watermark is to be used as evidence of ownership in a court of law more than once, much more of the watermark may be revealed to potential attackers. But since so little of the watermark is exposed with each random selection of n number of blocks, the watermark can be used a great number of times before a significant portion of the watermark is compromised, especially given the fact that these random selections may overlap, hence the term "almost zero-knowledge." To exemplify, consider the following: if a court requests a number of blocks to be revealed, which blocks were chosen remains unknown to others unless the court provides the details of which blocks were selected. Thus, true zero-knowledge would be achieved under these conditions, which require cooperation of the party that requests adjudication. In the case of the absence of this assurance, the analysis in this paper illustrates how the proof of ownership could be determined by revealing a small percentage of the watermarks over time.

B. A Specific Example: Randomly Selecting 10 Blocks

To give a more concrete example, let's say we select 10 as our value for n. In order for a judge in a court setting to verify that some 480 x 640 image belongs to us, he/she may randomly select 10 distinct 8 x 8 pixel blocks for us to reveal. If all 10 of these randomly selected blocks contain our watermark, then the judge has strong evidence that this image is indeed ours. Using this system, we are able to prove ownership without exposing the entire watermark. In fact, verifying 10 blocks at a time reveals less than half a percent of our watermark. Even though this method isn't completely zero-knowledge, it is an almost zero-knowledge

verification algorithm that accomplishes the same goals in a more practical and, most importantly, more convincing way.

One of the biggest goals of zero-knowledge protocols has been to protect the security of watermarks each time they are used in court as proof of ownership. Our proposed almost zero-knowledge algorithm compromises a very small fraction of that security with each individual use, but still allows for many uses before a significant portion of the watermark has been revealed. But how much counts as a "significant portion"? We can determine the number of uses it may take for different percentages of a watermark to be revealed, with random selections of 10 blocks at a time.

C. The Number of Random Selections Before a Watermark is Compromised

We can determine that given a 480 x 640 pixel image, it would take, on average, 138 random selections of 10 blocks in order to compromise 25% of the watermark and 332 random selections in order to compromise 50%. The graphs on the next page (see Fig. 1, Fig. 2) result from running a simulation 100,000 times and determining the average number of times it takes to randomly select 10 from 4800 elements until a given percent is compromised. It makes sense that the curve from 0% to 80% looks fairly linear, since there is less of a chance of randomly selecting a previously selected element, given the large total number of blocks. The more interesting portion of the graph begins around 90%, especially as the curve approaches 100% and drastically spikes upward. This also is not too surprising, since it would take multiple random selections to reach those last few holdouts.

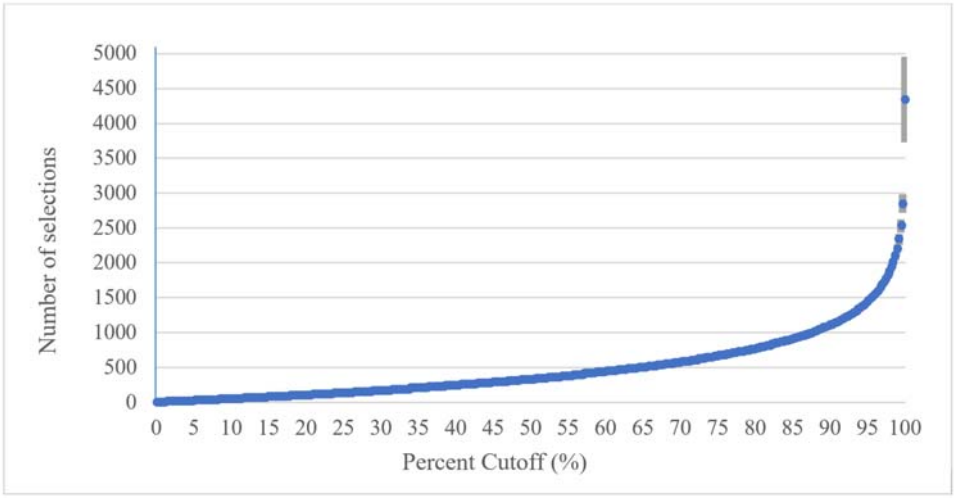

Fig. 1. The average number of selections of 10 from 4800 elements until the given percent cutoff is reached. The vertical bars represent 1 standard deviation above and below each average value

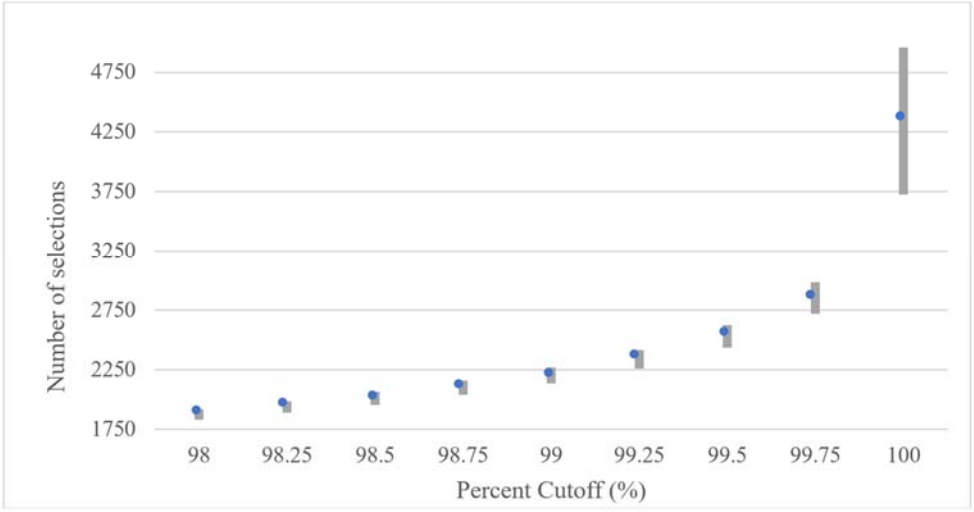

Fig. 2. [Zoomed in] The average number of selections of 10 from 4800 elements until the given percent cutoff is reached. Note again the vertical bars that represent standard deviation

The graph below (Fig. 3) explores the different curves that would result with different numbers of n blocks to be randomly selected with each trial. With fewer blocks randomly revealed each time, it takes on average more iterations to reach each percentage cutoff. The spikes for the lower n block values are also much more dramatic.

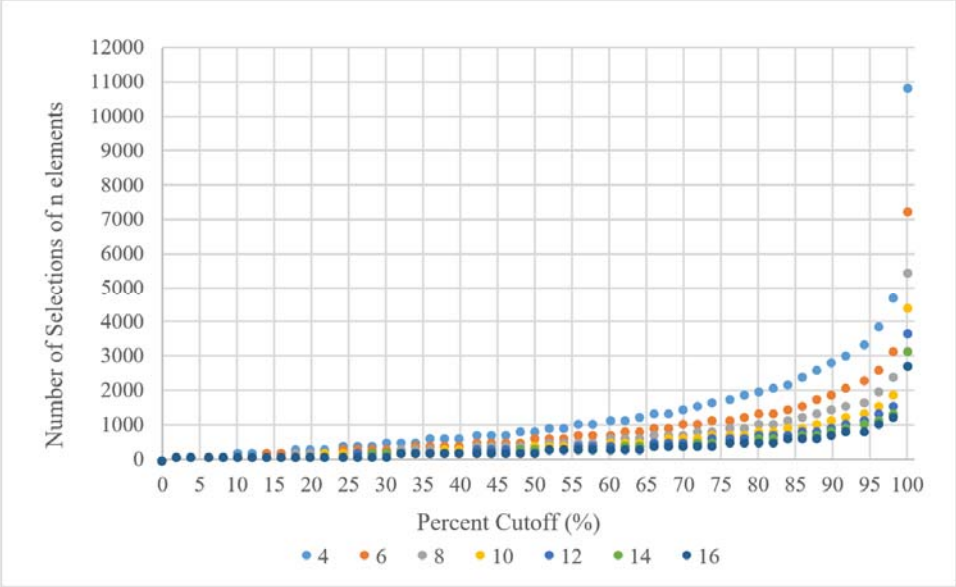

Fig. 3: The average number of selections of n from 4800 elements until the given percent cutoff is reached; n ranges from 4 to 16 (from top to bottom)

This method of randomly selecting blocks seems to guarantee many of the desired properties that zero-knowledge protocols strive to achieve, with very minor sacrifices. Even though this algorithm is not entirely zero-knowledge, it reveals so little information with each use that we can consider it almost zero-knowledge, since it may be employed many times without compromising much of its security to removal attacks.

IV. The Probability of Cheating the System

A. Removal Attacks by Random Guessing

To quantify the probability of an attacker successfully cheating the system, we can calculate the probability that an attacker may randomly guess the correct watermark for a single 8 x 8 pixel block.

If we are handling a JPEG image, we can reasonably assume that each 8 x 8 block contains at least 15 coefficients in which we can embed a watermark. Out of the 64 coefficients available, we might leave the 6 lowest frequency components untouched, modify the next 15, and ignore the remaining 43, many of which may be discarded by compression (Fig. 4). This allows more than two-thirds of the coefficients in an 8 x 8 block to be thrown away.

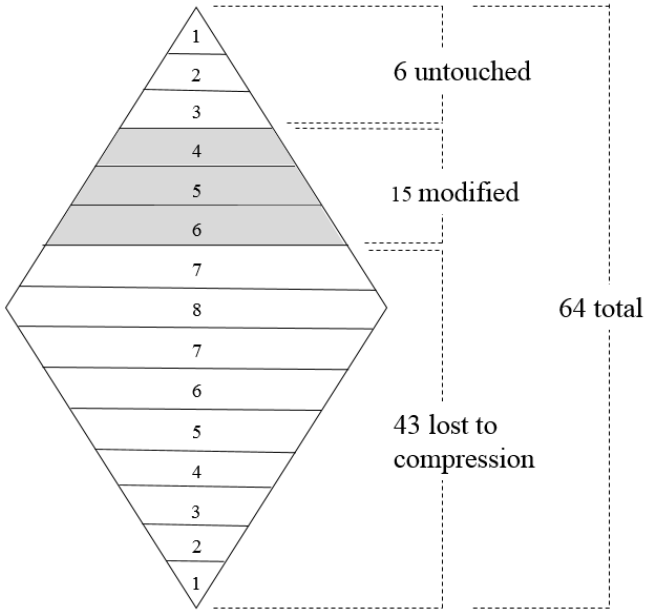

Fig. 4. A reasonable breakdown of coefficients in an 8 x 8 block that are left untouched, modified, or lost due to compression. From top to bottom, the coefficients are ordered from lowest to highest frequency

If, for a single block, we have 15 coefficients that can be modified in three ways (adding 1, leaving as is, or subtracting 1), then there exist a total of 3^{15}, or 14348907, watermarks possible. The chance of an attacker correctly guessing a watermark is then $1/3^{15}$. However, this is merely the probability of an attacker guessing the watermark of a single block. To guarantee success in a court setting, the attacker must successfully remove watermarks from at least 10 blocks, resulting in a probability of $(1/3^{15})^{10}$. Even this by itself is not enough to successfully attack a watermark, because there is an even smaller chance that the judge will randomly select these specific 10 blocks that have been attacked.

Of course, 15 coefficients with 3 possible modifications is a conservative assumption. If we instead worked with 22 coefficients, modifying each in five ways (+2, +1, 0, -2, and -2), we would have 2.3841858×10^{15} total watermarks possible.

But these probabilities assume that the attacker is randomly guessing watermarks without relying on any other information.

B. Failure to Prove Ownership with a Partially Compromised Watermark

The probability of an attack's success would change as more and more of the watermarks of an image are revealed with each court use. If an attacker removes individual watermarks as they are revealed in court, how many watermarks, or what percent, can be revealed before our algorithm fails to verify ownership?

The graphs below (Fig. 5, Fig. 6) represent the probability of randomly choosing 10 blocks that have been previously revealed in court and successfully removed by an attacker.

We can see that an image that has been 95% compromised, and therefore only retaining 5% of its watermarks, will fail a test of ownership with a probability of almost 60%. However, even an image that has been used in court enough times to reveal 75% of its watermarks will fail with a probability of only less than 10%. This shows that an image can withstand a high number of removal attacks before the image as a whole will fail verification in a court setting.

Even with the advantage of previously revealed watermarks, an attacker still has a very low probability of successfully cheating this system.

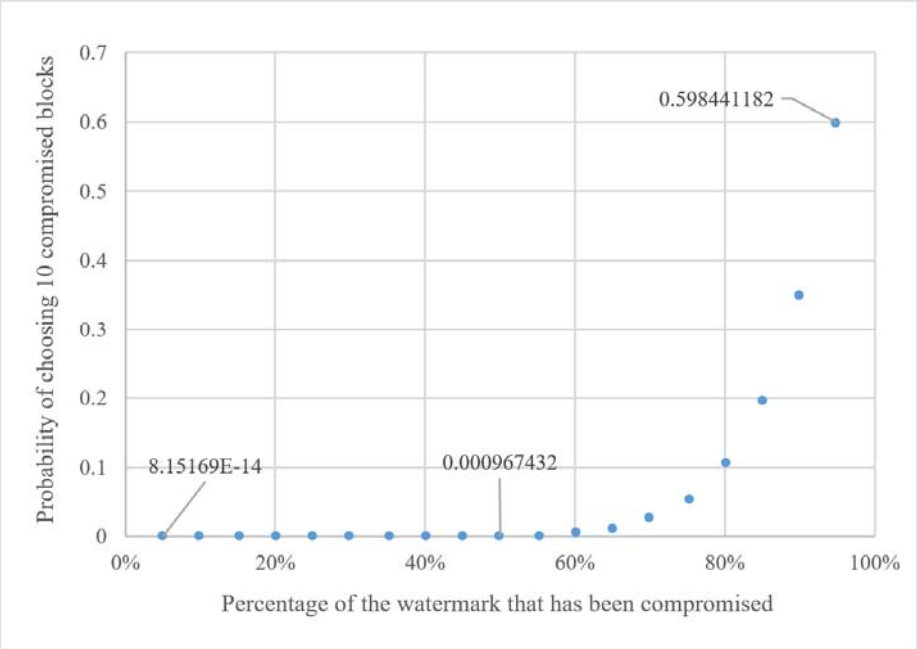

Fig. 5. The probability of failing to prove ownership: randomly choosing 10 from 4800 blocks that have all been compromised

Zooming into the first half of the graph, we can see that even after removing watermarks from a significant number of blocks, the probability of an attacker successfully preventing confirmation of ownership is incredibly low. This would strongly discourage potential attacks, since it would require the removal of almost 80% of the watermarks before reaching even a 10% chance of success.

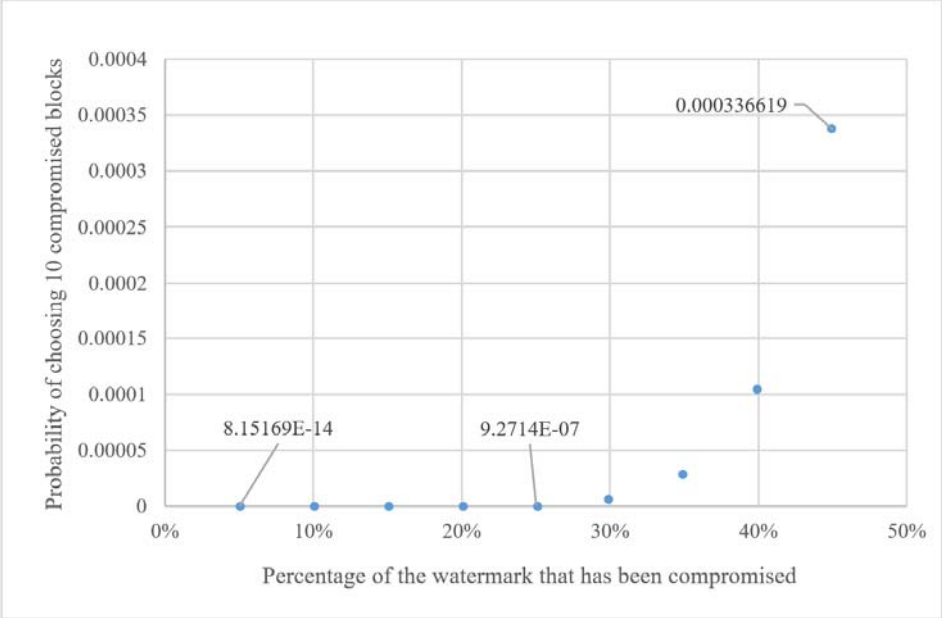

Fig. 6. [Zoomed in] The probability of failing to prove ownership: randomly choosing 10 from 4800 blocks that have all been compromised.

These graphs (Fig. 5, Fig. 6) of course assume that failure is defined by randomly selecting 10 blocks that have all been successfully attacked. But if a judge were to find that only 3 or 4 of these blocks contain the claimed owner's watermark, he/she may not have enough evidence to confidently believe their claim. Therefore, the judge may establish a certain cutoff to guarantee enough assurance of ownership proof. For example, if a judge requires at least 8 out of 10 randomly chosen blocks to demonstrate a watermark, a potential attacker only has to aim for a 7/10 result to prevent proof of ownership, and thus has a much higher chance of succeeding.

The judge's defined cutoff can drastically affect an attacker's probability of success. If a judge establishes 9 out of 10 as the minimum cutoff of convincing evidence, it would only take 2 attacked blocks to defeat the system. But, if a judge establishes 5 out of 10 as the minimum, it would take at least 6 attacked blocks to prevent ownership confirmation. The chances of success for each established cutoff turn out to be very different. This significant difference is reflected clearly in the graphs on the next page (Fig. 7, Fig. 8) that depict the probability of a successful attack depending on how much of the watermarked image has been removed as well as a minimum cutoff requirement that a judge may establish.

It is quite clear that, with 50% of watermarked blocks having been removed, a judge's decision has an extreme effect on the system's security against attacks. It can either dramatically raise an attacker's chance of success or dramatically lower it.

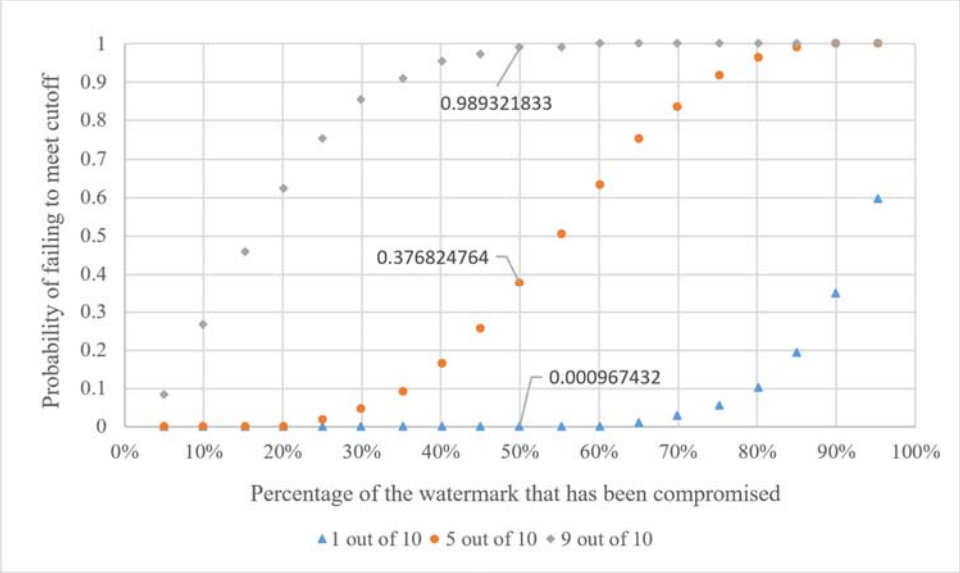

Fig. 7. The probability of success in preventing proof of ownership, given different minimum cutoff requirements

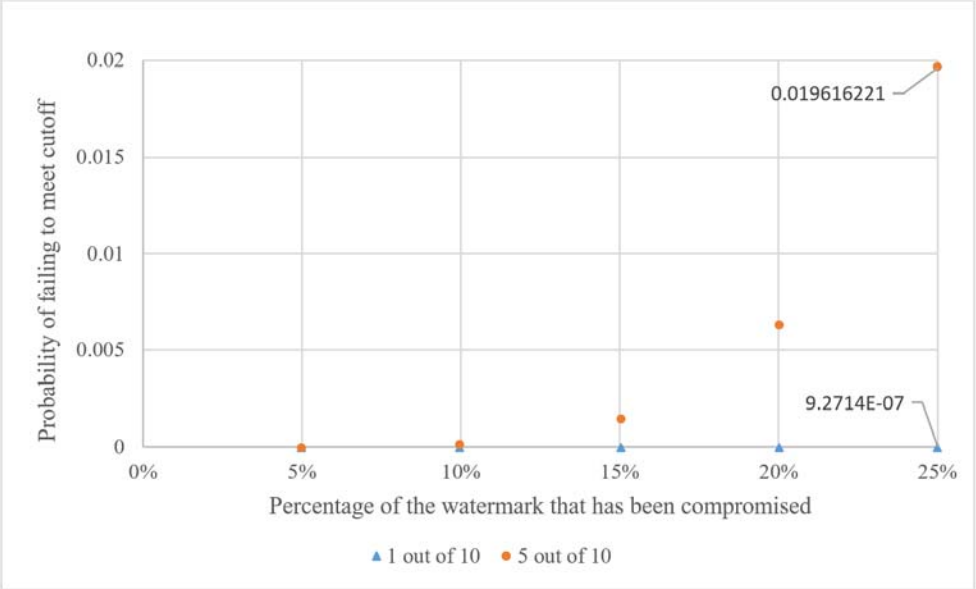

Fig. 8. [Zoomed in] The probability of success in preventing proof of ownership, given different minimum cutoff requirements

These graphs (Fig. 7, Fig. 8) make it clear that a judge's defined cutoff has a huge effect on an attacker's chance of success and therefore a direct impact on the security of a watermarked image. A judge's single decision holds a lot of power and may determine whether or not a claim of ownership will be verified.

After a quick glance at the graphs, one might conclude that the 1/10 requirement is obviously the best cutoff, since it results in the lowest probability of an attacker's success and thus guarantees the highest security. But it is not that simple. In order to maximize security, one has to be extremely lenient and accept unconvincing evidence as proof of ownership. But in order to maximize confidence in an owner's claim, one must sacrifice security against attacks. This significant trade-off between security and confidence presents a difficult dilemma. In the end, however this issue is merely a matter of definition, and it is ultimately up to the judge to define the border between a successful demonstration of ownership and a failing one.

V. USING A BLACK BOX DETECTOR

With the above probabilities as motivation, we can describe a black box detector – a standardized tool that takes as input the entire watermarked image, allows the selection of a single block, and outputs a simple "yes" or "no" to demonstrate whether a given block does or does not contain the claimed watermark.

The use of a black box detector discourages removal attacks even more heavily. With each use in court, the watermarks themselves now do not have to be entirely disclosed, but are instead protected by this tool's binary "yes" or "no" output.

Previously, we analyzed probabilities assuming that each use of a watermarked image directly reveals the selected watermarks and grants potential attackers enough knowledge to remove them. However, the introduction of a black box detector adds an extra layer of security. Earlier in this paper, we discussed the number of times our algorithm may allow one to verify ownership of a watermarked image before a significant percentage of its watermarks may be compromised and subsequently removed. But, with the added protection of a black box detector, it is very possible that a watermarked image may be tested in court multiple times without disclosing enough information to assist any removal attacks.

A. Potential Black Box Detector Attacks

With access to a black box detector, even with one that only indicates a binary decision, an attacker may, by brute force, determine the watermarks for each of these 8 x 8 blocks in a method similar to the one outlined in [9].

Cox describes an attack that takes advantage of the existence of an accessible black box detector. The objective of such an attack is to methodically identify the behaviour of the detector, and to use this knowledge to avoid triggering the detector with a targeted image. One may attempt to accomplish this by studying an image that is close to the boundary at which the detector changes its decision from "absent" to "present" [1].

Cox lays out five steps that an attacker may take to exploit the availability of a black box detector:

1) After obtaining a test image that is near the boundary of a watermark being detectable, meaning minor modifications cause the detector to change its response between "present" and "absent," the attacker adjusts the image step-by-step until the detector responds "no watermark found."

2) The attacker then manipulates the luminance of a single pixel until the detector detects the watermark again.

3) The attacker repeats this step for each pixel in the image.

4) With the knowledge of the detector's sensitivity to modifications of each pixel, the attacker estimates a combination of pixel values that best escapes detection while minimally affecting the image.

5) The attacker subtracts this estimate from the original marked image, resulting in an image that fails watermark detection.

In this way, an attacker may systematically estimate the watermarks for all of the blocks in an image by tackling a block pixel by pixel, or coefficient by coefficient.

Therefore, given access to a watermark detector, a removal attack is no longer a matter of random chance or probability, but rather a matter of time. But this same problem applies to existing watermark schemes that utilize black box detectors. Even with the potential of detector-based attacks, our algorithm is still more secure than current watermarking techniques. Using our method, successful attacks become much more difficult and take much more time simply because attackers must remove multiple watermarks instead of just a single one.

VI. CONCLUSION

When watermarks are employed as proof of ownership, the marks themselves are revealed and are thus compromised to removal attacks. To address this problem, we have outlined a practical, almost zero-knowledge watermark verification algorithm – a method by which one may prove the presence of a watermark while only revealing a very small fraction of the mark. Important for our approach is the requirement that each 8 x 8 block be marked with a different watermark. Analyzing the graphs in this paper demonstrates that our algorithm can withstand multiple uses before significant portions of the watermark are revealed, and still remains robust against attacks even after these significant portions are compromised. Our approach achieves many of the goals of existing zero-knowledge protocols, in a more practical way, with very minor sacrifices.

ACKNOWLEDGMENT

Support under NSF grants IIS-1359199 (REU Site Program) and DGE-1433817 (Scholarship for Service) is acknowledged.

REFERENCES

[1] Miller, M., et al. 1999. A Review of watermarking principles and practices. Digital Signal Processing in Multimedia Systems, Ed. K. K. Parhi and T. Nishitani, Marcell Dekker Inc. DOI=http://www0.cs.ucl.ac.uk/staff/I.Cox/Content/papers/1999/bookchapter99.pdf.

[2] Leiss, E. L. 2005. Time-Variant Watermarks for Digital Video: An MPEG-Based Approach. Digital Watermarking for Digital Media. J. Seitz (ed.). Idea Group Publishing, Hershey, PA.

[3] Leiss E. L. 2005. Time-Variant Watermarking of MPEG-Compressed Digital Videos. CLEI Electronic Journal. Vol. 8, No. 1, (Aug 2005), 1-12. http://www.clei.org/cleiej/papers/v8i1p1.pdf.

[4] Wallace, G. K. 1992. The JPEG Still Picture Compressing Standard. IEEE Transactions on Consumer Electronics. Vol. 38. No. 1 (Feb 1992), 18-35.

[5] Herrigel, A., et al. 1998. Secure copyright protection techniques for digital images. In: David Aucsmith. Second International Workshop IH'98. Springer, 169-190.

[6] Craver, S. 2000. Zero knowledge watermark detection. Information Hiding. DOI=http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.41.3556&rep=rep1&type=pdf.

[7] Schneier, B. 1996. Applied Cryptography: Protocols, Algorithms, and Source Code in C. 2nd ed. New York, NY. John Wiley and Sons.

[8] Pitas, I. 1996. A Method for Signature Casting on Digital Images. ICIP Proceedings. Vol. 3. IEEE press. 215-218.

[9] Cox, I. and Linnartz, J. 1997. Public watermarks and resistance to tampering. In Proceedings of International Conference on Image Processing.

AUTHOR PROFILE

Priscilla Lee is a computer science major at Wellesley College and is expected to graduate in 2018. In 2015, Ms. Lee was a Research Experience for Undergraduates (REU) grant recipient through the National Science Foundation (NSF) and spent that summer at the University of Houston working on Digital Watermarking. Her research interests include data security and artificial intelligence.

Mohammed Awad earned his PhD in Computer Science from the University of Houston in the United States in 2011. He earned a MSc in Computer Science at the same university in 2006. Dr. Awad's research interests include security, more specifically in E-voting and I-voting security. An additional area of interest concerns safeguarding the transmission of biometric data and integrating captured biometric data (iris scans or any other biometric data) into the electoral process in order to achieve more reliable systems.

Ernst L. Leiss earned graduate degrees in computer science and mathematics from the University of Waterloo and TU Vienna. He joined the Department of Computer Science of the University of Houston in 1979 where he has been a Full Professor since 1992. He has written over 170 peer-reviewed papers and six books and has supervised 17 PhD students. His research interests are in security and high-performance computing.