Photonics East, 1998

Authentication Techniques for Multimedia Content

N. Memon
Department of Computer Science,
Polytechnic University,
Brooklyn, NY 11201, USA

Poorvi Vora
Imaging Technology Dept.
Hewlett Packard Research Labs
Palo Alto, CA 640123

ABSTRACT

The recent proliferation of digital multimedia content has raised the issue of authentication techniques for multimedia content that is composed of still images, video and audio. Subsequently, there have been many authentication techniques for multimedia objects that have been recently proposed. One such class of techniques is based on digital watermarks and in this paper, we focus on such techniques. There are basically two types of watermarks that have been proposed for purposes of authentication, *Fragile Watermarks* and *Content-based Authentication Watermarks*. In this paper we survey different types of fragile and content-based authentication watermarking techniques that have been proposed in the literature. We point to new issues raised by the problem of authentication of multimedia content. We also discuss some shortcomings of proposed techniques and list open problems that still do not admit a satisfactory solution.

1 Introduction

Authentication techniques provide a means of ensuring the integrity of a message. It should be noted that, authentication, in general, is quite independent of encryption, where the intent is ensure the secrecy of a given message. Authentication codes are essentially designed to provide assurance that a received message has not been tampered with and has indeed originated from a specific source. This could be achieved with or without secrecy. In fact, for certain applications, secrecy could actually turn out to be an undesirable feature of an authentication technique. The general model under which authentication techniques are studied is shown in Figure 1.

In this model we have a transmitter, Alice, and a message X that she wishes to transmit to Bob over an open channel. In order for Bob to be assured that the message did originate from Alice and has not been modified, Alice computes an authenticated message Y which she sends over the open channel. Y is a function of X and a secret authentication key . In general, authentication is achieved by adding redundant information to a message. This redundant message could be in the form of an *authentication tag (or authenticator)* attached to the end of the message being authenticated. In this case Y would be of the form $Y = (X \ || \ a)$, where a is the appended authenticator and $||$ denotes concatenation. Authentication could also be achieved by redundancy present in the structure of the message, which could be recognized by the receiver.[12] For ease of exposition, lets assume the

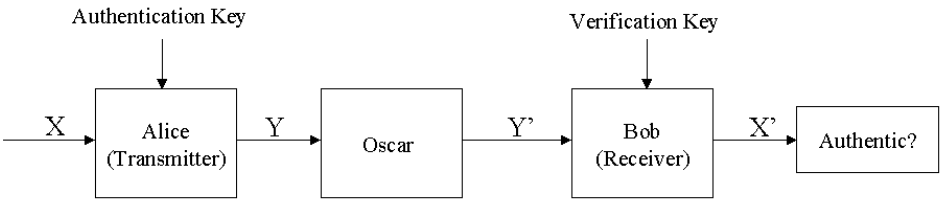

Figure 1: Authentication Model

former case.

If Bob receives $Y = (X \ || \ a)$ he could verify, using a verification key, that a is indeed a valid authenticator for X and accept the message. In a symmetric key system, the authentication and verification key are identical and both need to be kept a secret shared only between Alice and Bob. Since the authenticated message is being transmitted over an open channel, a malicious Oscar, can intercept the message and replace with another message $Y' \neq Y$ with $Y' = (X' \ || \ a')$ which he hopes Bob would accept as an authentic message. Note that Oscar performs this operation without knowledge of any secret key. Such an attack is called a *substitution attack*. Oscar may also insert a message Y' straight into the channel without knowledge of any authentic message that Alice has sent to Bob. Such an attack is called an *impersonation attack*. Oscar may also choose freely between a substitution attack and an impersonation attack. Authentication techniques that are unconditionally secure against these attacks, from an information theoretic point of view, are known.[12] One problem with the model described above is that Alice can always disclaim originating a message. Authentication techniques that are non-repudiable are also known. For an excellent recent survey on authentication techniques, the reader is referred to.[12]

Closely related to authentication techniques are digital signature schemes and message authentication code (MAC) generation algorithms. The former employs public key techniques to generate a signature for a message which can be verified by anyone having knowledge of the public key. Digital signature schemes are usually non-repudiable. MAC techniques are symmetric key (private key) based and in this sense similar to authentication codes. However, they only provide computational guarantees about security. That is, generating false messages is known to be (in most cases without any formal proof) computationally intractable. For an excellent introduction to digital signatures and related topics the reader is referred to[13] .

The recent proliferation of digital multimedia content has raised concerns about authentication mechanisms for multimedia data. In fact, there have been numerous authentication techniques for multimedia objects based on digital watermarks that have been proposed in the literature. Most of these techniques appear to have originated in the signal processing literature and are based on digital watermarks. Hence the focus of these efforts has been mainly towards embedding (and extracting) authentication codes in digital signals by means of an appropriate watermark. However, there has been little attention paid to cryptanalysis of proposed authentication techniques. In fact, we show later in this paper that some of the proposed techniques have some potential weaknesses and under certain reasonable assumptions, are subject to substitution attacks. Nevertheless, it appears that problem of authentication of multimedia content is not straight forward and potentially raises many new issues, some of which we list below.

- It may be desirable in many applications to authenticate the content, rather then the representation of the content. For example, converting an image from JPEG to GIF is a change in representation. One would like the authenticator to remain valid across different representation as long as the perceptual content has not been changed.
- When authenticating multimedia content, it is often desirable that the authenticator be embedded in the

data itself, thereby changing the very data that is being authenticated! A common reason cited for doing this is the fact that an authentication tag attached to a header can always be removed. Another advantage is the fact that embedding the authenticator in the content does not require any modifications to the large number of existing representation formats for multimedia content that do not provide any explicit mechanism for including an authentication tag (like the GIF format for still images, for example). However, what in our opinion is the most important advantage, is in the case of content-based authentication as mentioned above, where an authentication tag embedded in the content would be very convenient as it would survive transcoding of the data across different formats, including analog to digital conversions, in a completely transparent manner.

- When authenticating multimedia content, it is desired that in addition to detection of the event that modification has been made to content, one should also detect the exact location the modification has taken place. At first, it may seem straight forward to do this by blocking the bit-stream and appending authentication tags for each block. However, as we later show, due to the highly redundant representation of typical multimedia content, such an approach can lead to some simple substitution attacks.
- Given the highly data intensive nature of multimedia content, any authentication technique has to be computationally efficient to the extent that a simple real-time implementation, both in hardware and software should be possible.
- Multimedia content, by definition, consists of multiple bit streams representing different media which are ultimately presented in some synchronized manner. It would be important to authenticate not only each individual bit stream, but also their temporal or structural relationships. For example, in video, the image sequence and the audio need to be synchronized and this synchronization itself needs to be authenticated to some extent. Else, an attacker could rearrange the audio stream to change the nature of the scene being conveyed by the original video.

In the rest of this paper we survey different multimedia authentication techniques that have been proposed in the literature and point to some shortcomings and open problems that still do not admit a satisfactory solution. We first begin in section 2, by describing a few representative techniques from the literature. Then in section three we show how some of the proposed techniques are vulnerable to different types of attacks, especially, if they are not designed in a careful manner. Finally, in section four we conclude with a discussion on problems that still remain.

2 Authentication by Invisible Watermarks

Digital watermarking is the process of embedding a digital signature into digital multimedia content such that the signature (or watermark) can later be extracted or detected for a variety of purposes including authentication or identification. For ease of exposition we assume that the content being watermarked is a still image, though most digital watermarking techniques are, in principle, equally applicable to audio and video data. A digital watermark can be *visible* or *invisible*. A visible watermark typically consists of a conspicuously visible message or a company logo indicating the ownership of the image. On the other hand, an invisibly watermarked image appears visually very similar to the original. The existence of an invisible watermark can only be determined using an appropriate watermark extraction or detection algorithm. In this paper we restrict out attention to invisible watermarks.

In general, the watermark insertion step can be represented as follows:

$$X' = \mathcal{E}_K(X, W) \tag{1}$$

where X is the original image, W is the watermark information being embedded, K is the user's insertion key, and $\mathcal{E}$ represents the watermark insertion function. We adopt the notation throughout this paper that for

an original image X, the watermarked variant is represented as X'. Depending on the way the watermark is inserted, and depending on the nature of the watermarking algorithm, the detection or extraction method can take on very distinct approaches. One major difference between watermarking techniques is whether or not the watermark detection or extraction step requires the original image. Watermarking techniques that do not require the original image during the extraction process are called *oblivious* (or public) watermarking techniques. For oblivious watermarking techniques, watermark extraction works as follows:

$$\hat{W} = \mathcal{D}_{K'}(\hat{X}') \tag{2}$$

where $\hat{X}'$ is a possibly corrupted watermarked image, K' is the extraction key, $\mathcal{D}$ represents the watermark extraction/detection function, and $\hat{W}$ is the extracted watermark information. Oblivious schemes are attractive for many applications where it is not feasible to require the original image to decode a watermark.[9] Clearly, a watermarking scheme employed for the purpose of authentication needs to be oblivious.

Invisible watermarking schemes can also be classified as either *robust* or *fragile*. Robust watermarks are often used to prove ownership claims, and so are generally designed to withstand malicious attacks such as image scaling, cropping, lossy compression, and so forth. An example watermarking technique that is remarkably robust to such attacks is given in.[2] In comparison, fragile watermarks have been proposed for purposes of authentication, and can potentially be used to verify the integrity of a given image's content. For an excellent survey on robust and fragile watermarking techniques, see[3] or.[14]

As mentioned earlier, a watermark signal embedded into multimedia content can serve a variety of purposes including ownership assertion, fingerprinting, usage control, copy prevention, authentication and content labeling. In this paper, however, we are concerned with application of watermarks for the purposes of authentication. There are basically two types of watermarks proposed for authentication, *Fragile Watermarks* and *Content-based Authentication Watermarks*. In this section, we briefly describe a few examples of each that have been proposed in the literature. Our goal is not to give an exhaustive survey, but to provide the reader with an idea about the types of techniques that have been proposed.

2.1 Authentication by Fragile Watermarks

One of the earliest fragile watermark proposed in the literature was by Yeung and Mintzer.[18] In this technique, a binary watermark image W is embedded into a source image X, so that subsequent alterations to the watermarked image X' should be detected. Generally W is a binary image of the same dimensions as the image X. Watermark insertion proceeds by examining each pixel $X_{i,j}$ in turn, and applying the watermark extraction function $\mathcal{D}$. If the extracted watermark value is equal to the desired watermark value, $W_{i,j}$, processing continues with the next pixel; otherwise, the current pixel value is adjusted until the extracted watermark value equals the desired value. This process is repeated for each pixel in the image.

The watermark extraction function is computed from the owner's key, and is defined as:

$$W_{i,j} = LUT_{Red}(X_{Red}(i,j)) \oplus LUT_{Green}(X_{Green}(i,j)) \oplus LUT_{Blue}(X_{Blue}(i,j)) \tag{3}$$

for RGB color images, and $W_{i,j} = LUT(X(i,j))$ for greyscale images, where the LUT's are binary lookup tables, one per color component, and $\oplus$ indicates an XOR operation. The lookup table contents are known only to a user possessing the key; the key could be used to seed a pseudo-random number sequence used to generate the tables, for example. In addition to this process, a modified error diffusion method is used to maintain proper average color over the image. Subsequent image verification is accomplished by applying the watermark extraction function to X' to generate $\hat{W}$, which is compared to the original watermark W. Changes to any portion of the image X' should result in changes to the corresponding block of the extracted watermark.

Another, more secure fragile watermarking technique recently proposed by Wong[17] inserts an invisible watermark W into an $m \times n$ image, X. The original image X is partitioned into $k \times l$ blocks, such that X_r is taken to

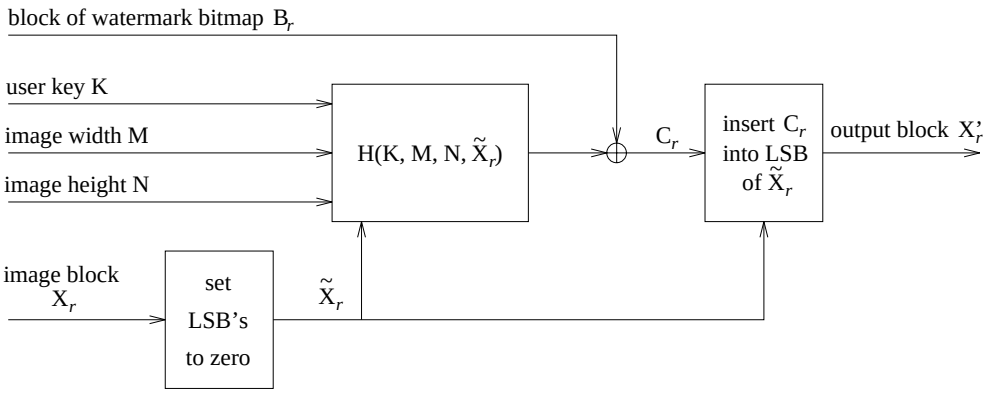

Figure 2: Wong's watermark insertion procedure, applied independently to each image block.

mean the r^{th} block of the image; the bi-level watermark W is partitioned likewise, such that W_r denotes the r^{th} block of the watermark. For each image block X_r, a corresponding block $\tilde{X}_r$ is formed, identical to X_r with the exception that the least significant bit of every element in $\tilde{X}_r$ is set to zero.

For each block X_r, a cryptographic hash $H(K, m, n, \tilde{X}_r)$ (such as MD5,[11]) is computed, where K is the user's key. The first kl bits of the hash output, treated as an $k \times l$ rectangular array, are XORed with the current watermark block W_r to form a new binary block C_r. Each element of C_r is inserted into the least significant bit of the corresponding element in $\tilde{X}_r$, generating the output block X'_r.

Image authentication is performed by extracting C_r from each block X'_r of the watermarked image, and by XORing that array with the cryptographic hash $H(K, m, n, \tilde{X}'_r)$ in a manner similar to above, to produce the extracted watermark block. As with the Yeung-Mintzer scheme, changes to the watermarked image result in changes to the corresponding binary watermark region, enabling the technique to be used to localize unauthorized alterations to an image. Figure 2, taken from Wong's paper,[17] illustrates the process of watermark insertion.

The watermarking algorithm can also be extended to a public key version where a private key K'_A of a user A is required to insert the watermark. However, the extraction only requires the public key of user A. More specifically, in the public key version of the algorithm, the MSB's of an image data block X_r and the image size parameters are hashed, and then the result is encrypted using a public key algorithm. The resulting encrypted block is then XOR'ed with the corresponding binary watermark block W_r before the combined results are embedded into the LSB of the block. In the extraction step, the same MSB data and the image size parameters are hashed. The LSB of the data block (cipher text) is decrypted using the public key, and then XOR'ed with the hash output to produce the watermark block.

2.2 Content-based authentication watermarks

The methods described in the previous subsection authenticate the data that forms the multimedia content, and the authentication process does not treat the data as being distinct from any other data stream. Only the process of inserting the signature into the multimedia content treats the data stream as an object that is to be viewed by a human observer. For example, while watermarking images, Yeung and Mintzer[18] maintain overall average image color; and Wong[17] inserts the watermark in the least significant bit thus discarding the least significant bits of the original data stream and treating them as perceptually irrelevant, or irrelevant to image content.

All multimedia content in current representations have a fair amount of in-built redundancy, that is to say that the data representing the content can be changed without effecting a change that is actually perceptible; further, changes to the data data can also perceptible, but may not affect the content. For example, when dealing with images, one can brighten an image, lossy compress it, or change contrast settings. The changes caused by these operations could well be perceptible, even desirable, but the image content is not considered changed - people, if any in the image, are in the same positions; the clothes they are wearing as well as the geographical setting are recognizable.

It is highly desirable that authentication of multimedia documents take this into account - that is, there be a set of 'allowed' operations, and 'image content'; it is with respect to allowing the first and retaining the second that any authentication should be performed for it to be genuinely useful.

There have been a number of recent attempts at authentication which address authentication of 'image content', and not of only image data. The problem with all these methods is that 'image content' is itself an extremely ill-defined quantity despite the attempts of the vision and compression communities to nail it down. In this subsection, we describe some of the attempts at authenticating image content and point out the problems with these methods.

Bhattarcharjee[1] suggests the use of feature points in defining image content that is robust to image compression. An image authentication scheme that allowed image compression would then be one which used cryptographic authentication schemes to authenticate the feature points. Typical feature points include, for example, edge maps. The problems with this method include the following: edge maps do not sufficiently define image content - for example it is possible to have two images with fairly different content (the face of one person replaced by that of another) but with identical edge maps.

Fridrich[5] suggests the use of an invisible robust watermark which depends on 'image content'. If the image content changes significantly, the robust watermark generated by the changed image will be significantly different from the robust watermark embedded in the image (this embedded watermark was generated by original image content and remains in the image in spite of changes because it is robust). In this case, 'image content' is defined by quantized coefficients with respect to a set of smoothed pseudo-random sequences generated using the camera key. The quantization level decides the amount of information contained in the watermark and also decided the amount and kind of transformations allowed. This method has all the problems of other methods that use correlation and quantization for representation of image content (including JPEG), in that perceptual errors and errors in image content do not always correlate well with (quantization) errors in data. Nevertheless, in avoiding cryptographic digests, Fridrich manages to incorporate a degree of redundancy-tolerance in authentication. It should be possible to extend Fridrich's method to use other compression schemes, where the basis set is in some way dependent on the camera key. It should also be possible to quantify the relationship between image tampering and differences in signature.

3 Substitution Attacks

We mentioned before that watermarking techniques for authentication purposes are usually designed such that the receiver is able to locally pin-point changes made to the content. This is often achieved by partitioning the content into blocks and inserting an authentication watermark in each block.[6,8,16–19] A block-based approach can be convenient in terms of simplicity and lack of computational overhead. However, a number of block-based proposed methods proposed in the literature suffer from an inherent weakness, this weakness being the *block-wise independence* of the watermark insertion and detection process. More precisely, the insertion of a watermark $W = \{w_1, w_2, \ldots, w_n\}$, into an image X, consisting of blocks $\{X_1, X_2, \ldots, X_n\}$, satisfies the following property, where $\|$ denotes concatenation:

$$\mathcal{E}_K(X, W) = \mathcal{E}_K(X_1, w_1) \,\|\, \mathcal{E}_K(X_2, w_2) \,\|\, \ldots \|\, \mathcal{E}_K(X_n, w_n) \tag{4}$$

In other words, the watermark inserted in each image block is independent of the both the watermark inserted in the other blocks and the image in the other blocks. The same property applies to the watermark detection process.

In this section we show that schemes possessing the property stated in equation 4 are potentially vulnerable to impersonation attacks whereby counterfeit watermarks can be inserted into images without the consent of the original watermark owner. Specifically, given one or more images containing an owner's watermark W inserted using a fixed key K, and an unwatermarked image Y, it is possible for an attacker to construct a watermarked image Y' such that $\mathcal{D}_K(Y') = W$, without having any knowledge of the original watermark owner's key, K and, in the case of robust watermarks, without the knowledge of W.

3.1 The Yeung-Mintzer scheme

Forging a watermark in the Yeung-Mintzer scheme takes advantage of the fact that each pixel output is independent of any other, so we treat the scheme as a 1×1 block-based technique. The attack assumes knowledge of the binary watermark logo embedded in a user's images, and essentially groups pixels from a watermarked image into two disjoint sets; the first consists of pixel values associated with zero bits in the binary watermark logo, while the second consists of pixel values corresponding to one bits in the logo. Counterfeiting the watermark in a new image therefore reduces to the problem of quantizing the new image's pixel values using the two sets as codebooks; the choice of set to use depends upon the desired binary logo value at each location.

More formally, given an existing watermarked image X' containing a binary logo watermark image W, and given an unwatermarked image Y, counterfeiting the watermark W in Y to produce Y' proceeds as follows. As noted above, we assume here that the binary watermark W is known; but the user's key, and consequently the lookup tables used during watermark insertion, are unknown to the attacker. Two sets of pixel values (RGB triples if we are dealing with RGB color images) are created, S_0 and S_1, corresponding to the two possible logo intensities 0 and 1. For every pixel $X'_{i,j}$ in X' we add $X'_{i,j}$ to S_0 if $W'_{i,j} = 0$ and to S_1 if $W'_{i,j} = 1$. Following this step, for every pixel $Y_{i,j}$, we find an approximating pixel value in the set $S_{W_{i,j}}$, and output that as the watermarked pixel $Y'_{i,j}$. In essence, for RGB images, the counterfeiting process involves vector quantization of the RGB triples of the unwatermarked image Y using the two codebooks S_0 and S_1 constructed from the watermarked image X'.

In general, as a consequence of the pixel-based nature of the basic approach, forging a watermarked image in this scheme results in excellent results in terms of perceptual quality; this is particularly true when using images of similar colors, since the quantization error in each reconstructed pixel is a function of this parameter. An example of an attack on this scheme is shown below; Figure 3 shows a legitimately watermarked image, which is visually indistinguishable from its original counterpart. Figure 4 shows the original (unwatermarked) second image, and Figure 5 shows an approximation to the second image containing a forged version of Figure 3's watermark. PSNR's were 40.34 dB (red), 39.87 dB (green), and 42.21 dB (blue) for the reconstructed image in Figure 5.

3.2 The Wong scheme

As with the Yeung-Mintzer scheme, counterfeiting a watermark in Wong's scheme can be accomplished by vector quantization of an image, where the codebook to use for a given image block is determined by the watermark logo block to be embedded at that location; construction of an image containing a forged watermark is subject to the constraint that every block in the forged watermarked image must contain the logo block associated with that particular location.

Figure 3: Original image, watermarked using the Yeung-Mintzer scheme.

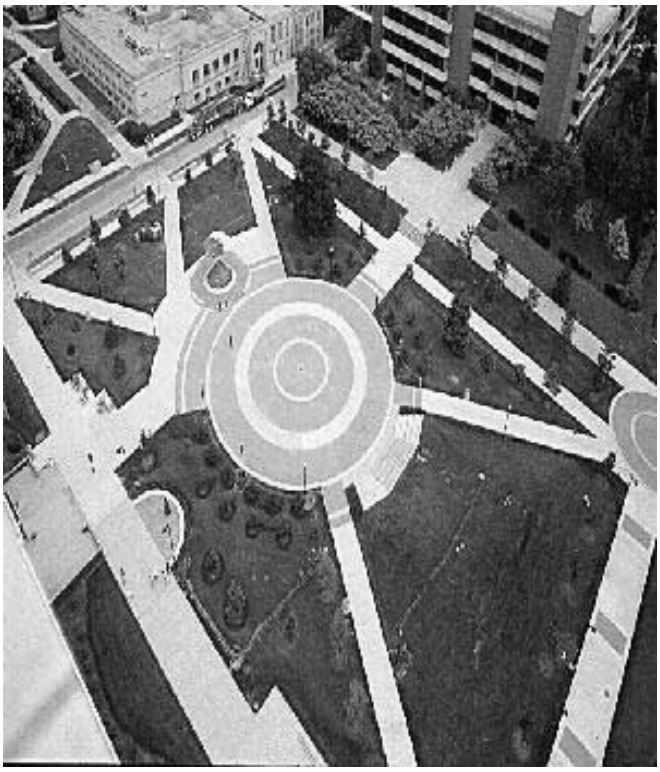

Figure 4: Original image, unwatermarked.

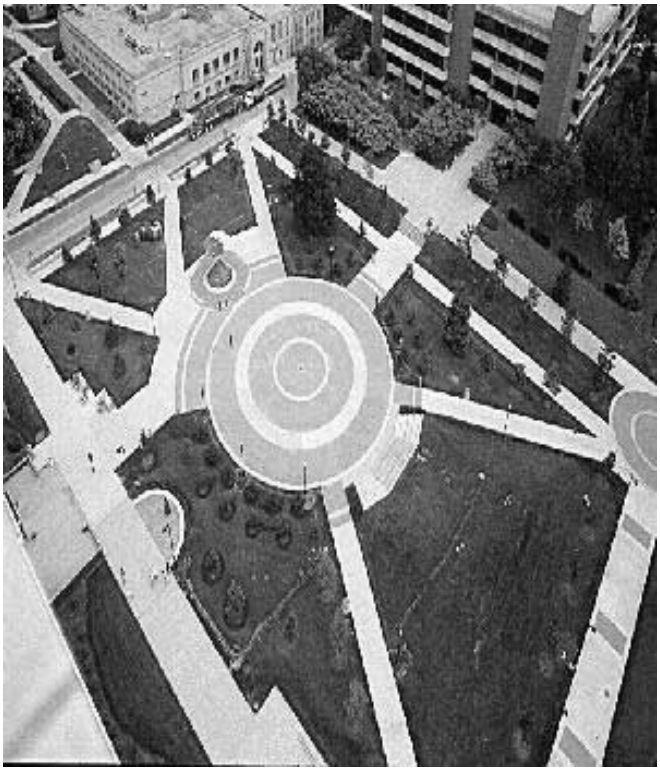

Figure 5: Constructed image, containing the counterfeit Yeung-Mintzer watermark.

Figure 6 illustrates an example watermark logo configuration possible when using this watermarking technique. In general, the binary watermark may likely consist of a tiled rendition of a smaller image, as depicted here, or it could consist of a single image, possibly padded out to the correct size. The visual quality of a constructed image containing a forged watermark is therefore dependent upon both the periodicity of the binary watermark logo, and upon the number of watermarked images available to the attacker.

Consider a set of watermarked images, $\{X'_1, X'_2, \ldots, X'_k\}$, each of size $m \times n$, and each containing the same binary watermark W, inserted using a fixed key K. We assume here that the logo W and the block size parameters, k and l, are known, and that the user's key, K, is unknown. Given an unwatermarked image Y, also of size $m \times n$, generating an image Y' which contains the fake watermark W proceeds as follows. For each unique block W_r in the binary watermark logo image W, determine the set of blocks, S_{W_r}, in the set of watermarked images X'_i available such that each image block in S_{W_r} has embedded in it the same binary logo W_r. For each block Y_r, which is to contain an associated block W_r from the binary image W, find an approximate block in the set S_{W_r}, and output that as the watermarked block Y'_r. Again, as in the previous subsection, this essentially resembles vector quantization of the image Y using the codebooks S_{W_r}, one codebook corresponding to each unique block in the binary logo image W.

The attack described above is for the private key version of the algorithm. However, the same attack essentially applies to the public key version described in the previous section. In fact, for the public key version the binary watermark logo is necessarily known to the attacker.

It should be noted that the above attack is applicable to many other watermarking techniques proposed in the literature. For more details the reader is referred to.[7] However, it should also be noted that the ease by which the above attack can be carried out depends on the block size used (that is k and l) and the nature

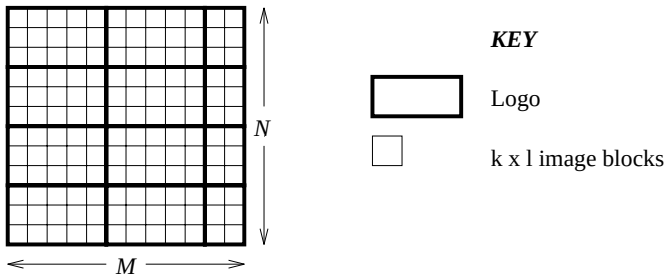

Figure 6: Example logo configuration using Wong's technique.

of the watermark logo image.[7] The key to the substitution attacks presented above the manner in which the watermarking technique embeds the watermark information block by block into discrete, independent segments of an image. Such attacks can be defeated by making watermark insertion in a given block dependent upon other blocks in the image. For example, in Wong's technique, the computed hash for each image block could be a function of preceding blocks, not just of the block in question.

4 Discussion

Authentication techniques have been well studied in the literature for the past few decades. However, given the large amount of redundancy present in multimedia content, and consequently the large number of different representations of perceptually identical content, authentication techniques for multimedia presents some unique problems. The most difficult problem, perhaps is the development of techniques that authenticate content rather than representation of content. The foremost difficulty is in defining image content, which remains an unsolved problem in spite of the enormous amount of recent research in image understanding. Further, the essential cryptographic authentication techniques used are highly discontinuous and signatures change considerably with a small change in the data. This is what provides the security of the authentication scheme, but it is also what provides the limitation in the authentication of multimedia content with high redundancy. The real need is for cryptographic methods that deal with redundancy in a well-defined way.

Although the authentication problem addressed by fragile watermarks appears to be well suited to traditional authentication techniques, the fact that the authenticator is embedded in the content raises some interesting questions that to our knowledge have not been addressed in the literature. Since, embedding the authenticator is introducing distortions into the content, it is natural to ask about the rate-distortion like trade-off's between the authentication bits embedded and the fidelity of the underlying content. What is the optimal number of authentication bits that we can embed? How is optimality defined? Clearly, there will be an optimal manner in which the data can be embedded, as also an optimal part of the image (in some domain) for embedding. Addressing these problems requires the use of models of perceptual error, for example recent work by Fleet and Heeger[4]

5 REFERENCES

[1] S. Bhattacharjee, "Compression Tolerant Image Authentication", *Proceedings, Int. Conf. Image Proc.*, Chicago, Oct. 1998.

[2] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol 6, no 12, pp 1673-1687, 1997.

[3] I. J. Cox and M. L. Miller. A review of watermarking and the importance of perceptual modeling. In *Proceedings, SPIE Human Vision and Electronic Imaging II*, volume SPIE Vol. 3016, February 1997.

[4] D. Fleet and D. Heeger, "Embedding Invisible Information in Color Images", *Proceedings, Int. Conf. Image Proc.*, Santa Barbara, Oct. 1997.

[5] J. Fridrich, "Image Watermarking for Tamper Detection", *Proceedings, Int. Conf. Image Proc.*, Chicago, Oct. 1998.

[6] F. Hartung and B. Girod. "Digital watermarking of uncompressed and compressed video," *Signal Processing*, to appear 1998.

[7] M. Holliman and N. Memon. "Counterfeit Attacks on Block-wise Independednt Watermarking Techniques," *Pre-print*, 1998.

[8] G. Langolaar, V. Lubbe and J. Biemond, "Copy protection for multimedia data based on labeling techniques," *www-it.et.tudelft.nl/pda/smash/public/benelux_cr.html*, 1996.

[9] N. Memon and P. Wong. "Digital Watermarks: Protecting Multimedia Content," *Communications of the ACM*, July 1998.

[10] F. Petitcolas, R. Anderson, and M. Kuhn. "Attacks on copyright watermarking systems,", Proceedings of the Information Hiding Workshop, Portland, Oregon, April 1998.

[11] R. L. Rivest, "The MD5 message digest algorithm." Internet RFC 1321, April 1992.

[12] Gus Simmons. "A Survey of Information Authentication," *In Contemporary Cryptography, The Science of Information Integrity*, IEEE Press, 1992.

[13] D. Stinson, *Cryptography, Theory and Practice*, CRC Press, 1995.

[14] M. Swanson, M. Kobayashi, and A. Tewfik, "Multimedia Data Embedding and Watermarking Technologies," *IEEE Proceedings*, vol. 86, No. 6, pp 1064-1087, June 1998.

[15] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Data Hiding for Video-in-Video," *Proceedings of the IEEE Int. Conf. on Image Processing (ICIP 97)*, Santa Barbara, CA, October, 1997.

[16] M. Swanson, B. Zhu, and A. Tewfik, "Transparent robust image watermarking," in *Proceedings of the International Conference on Image Processing*, volume 3, pp. 211–214, September 1996.

[17] P. W. Wong, "A watermark for image integrity and ownership verification." To appear in Proceedings of IS&T PIC Conference (Portland, OR), May, 1998. Also available as Hewlett Packard Laboratories Technical Report HPL-97-72, May 1997.

[18] M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *Proceedings of the International Conference on Image Processing*, volume 1, pp. 680–683, October 1997.

[19] J. Zhao and E. Koch, "Embedding Robust Labels into images for Copyright Protection," *Intellectual Property Rights and New Technologies, Proceedings of the KnowRight'95 Conference 1995*, pp. 242–51.