Elektrotehniški vestnik 77(5): 251-258, 2010
Electrotechnical Review: Ljubljana, Slovenija

An Authentication System for Digital Images

Tomaž Nahtigal , Andrej Žemva

University of Ljubljana, Faculty of Electrical Engineering, Tržaška 25, 1000 Ljubljana, Slovenia
E-pošta: tomaz.nahtigal@fe.uni-lj.si

Abstract. The ability to modify digital images can cause a serious problem in some applications. In recent years, a considerable amount of research has been directed towards developing methods that would protect the authenticity of digital images. The major part of this research has been devoted to obtaining an understanding of the information theory involved and only a small part of it has been focused on the actual implementation of the methods. A workable authentication system that would be secure and ready for use in a wide range of applications is still to be made. In this paper we propose a novel method derived from the Wong's authentication scheme that is capable of authenticating JPEG images as well as uncompressed images, but is not a watermarking method. Our method offers great flexibility in terms of the size of the authenticator and the time needed to generate the authenticator, thus making it suitable for real-time image processing. We demonstrate this by implementing a simplified version of the authentication unit in a programmable FPGA circuit.

Keywords: Digital image authentication system, Holliman-Memon attack, Global logo, Localization

Sistem za avtentikacijo digitalnih slik

Razširjeni povzetek. Možnost spreminjanja digitalnih slik je v nekaterih aplikacijah lahko resen problem. V zadnjih letih je bilo veliko raziskav usmerjenih v razvoj metod, ki bi zaščitile verodostojnost digitalnih slik. Večji del teh raziskav je bil namenjen pridobivanju razumevanja matematične teorije, ki je osnova za razvoj in varnost metod za avtentikacijo, le majhen del pa je bil osredinjen na dejansko implementacijo metod v praksi. Sistema za avtentikacijo digitalnih fotografij, ki bi bil varen in pripravljen za širšo uporabo, v praksi še ni. V tem prispevku predlagamo novo metodo, ki izhaja iz Wongove metode za avtentikacijo digitalnih slik in je namenjena preverjanju avtentičnosti JPEG, kot tudi nekomprimiranih slik. Ponuja možnost izbire končne velikosti prstnega odtisa slike in časa, potrebnega za izračun prstnega odtisa, s čimer je primerna za obdelavo slik v realnem času. Ustreznost metode pokažemo z implementacijo poenostavljene različice v programirljivem vezju FPGA.

Ključne besede: sistem za avtentikacijo digitalnih slik, napad Holliman-Memon, globalni logo, lokalizacija

1 Introduction

We live in a digital age. The rapid development of digital techniques, cost-effective digital storage devices and the widely spread personal computers and the Internet allow us to exchange and manipulate digital data with great ease. Analog data is in the process of being replaced by digital counterparts. The same applies to images. Digital images are easy to store, copy, edit and manipulate. They can be shared via computer networks, processed and stored in databases where they can be relatively simply managed.

Received 15 March 2010
Accepted 22 March 2010

Because of the image editing software, ensuring the authenticity of digital images poses a big problem. The image editing software allows for malicious modifications of digital images which can be very difficult to detect. In some applications, authenticity of images is of vital importance. For instance medical images, images in news, images for evidence in court, etc. need to be protected in order to avoid false judgments. Here, the use of digital images is not appropriate. Several methods, ranging from conventional cryptography to fragile watermarking, have been proposed to protect the authenticity of digital images. The methods differ in terms of the services they provide, that is tamper detection, localization and robustness against different image processing operations. For an overview of the methods refer to [1].

Despite the considerable number of proposed methods, only a few digital cameras equipped with authentication capabilities have emerged on the market. In [2-4], the authors propose a VLSI architecture for watermarking of digital images allowing for embedding of biometric data in the image. The majority of papers proposing authentication methods are not concerned with the actual implementation of the authentication system. We believe that block authentication methods are the most suitable for implementation because they process the image in a sequential manner. Wong [7] proposed a fragile block-wise watermarking scheme with tamper localization, which elegantly merges cryptography with watermarking. The scheme processes 8x8 blocks in a sequential manner. As it requires no storing of the whole image, it is fast and efficient. Other schemes [8-12] derived from Wong improve its

security, especially the resistance against the Holliman-Memmon attack.

Our goal is to design a method suitable for hardware implementation and enabling the system based on it to be secure, flexible and useful in real-life applications.

2 Authentication methods for images

The task of the authentication method is to generate an authenticator (tag) from the data to be authenticated. The authenticator is a series of bits derived in a prearranged manner for the purpose of attesting the authenticity of an image. The authenticator can be stored in a separate file or file header. Another technique used in image authentication is fragile watermarking. Watermarking employs data hiding or steganography. The basic idea behind fragile watermarking is to generate a watermark (tag) and to insert it so that any modification made to the image is also reflected in the inserted watermark. By checking the presence of the inserted watermark, the image authenticity is verified and tampered regions are eventually localized.

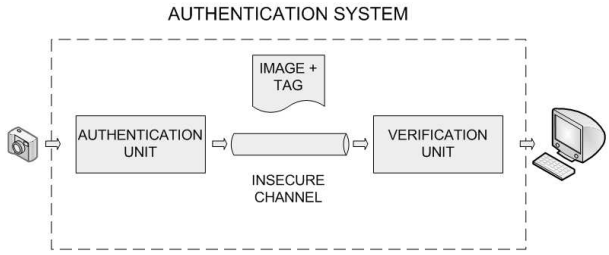


Figure 1. Authentication system

The authentication method is implemented in the authentication system (Figure 1). The system consists of an authentication unit and a verification unit. The authentication unit generates the authentication tag (watermark) for the image and the verification unit verifies the authenticity of the image. On one hand, we have the source of the image (digital camera) and on the other hand the end "user" of the image wanting to test the authenticity of the image. The transition of the image from the digital camera to the end user can be seen as a sort of communication through an insecure communication channel. At this point let us define the role of the attacker. The objective of the attacker is to produce a perfect forgery. To achieve this objective, the attacker needs to effectively mimic the workings of the authentication unit. The attacker is able to produce an authentication tag for any image. In practice this is very difficult to achieve, so other types of attacks trying to fool the authentication system have been devised. Instead of creating a perfect forgery, their aim is to modify the original image so that the authentication system recognizes it as being authentic. Because of the severity of consequences of a false authentication, any successfully attacked authentication system should be abandoned.

2.1 Authentication unit

The first important requirement for an authentication system to be effective is that the authentication tag is generated as close as possible to the source of the image. This means that the authentication tag must be generated in the digital camera. If the authentication unit is not implemented in the digital camera, but is instead a program running on a computer, the attacker can modify the image before the authentication tag is generated and the changes in the image will remain undetected. If such be the case, the authentication process serves only for authenticating the sender of the image and not the image itself. To implement the authentication unit in the digital camera, several design challenges need to be solved. The digital camera offers limited memory and limited processing capacity. The unit will also have to be capable of real-time image processing. An even greater challenge is imposed by the design of the authentication system that needs to be built into a digital camera of the mobile phone. Here factors like power consumption and cost play a much more important role. For an authentication system to be implemented in a digital camera, implementation constraints should be duly observed when designing the authentication algorithm. This means that special care should be taken in selecting the features of the authentication algorithm so as to allow for a balance between ease of implementation and the necessary level of security.

2.2 Verification unit

Another issue that strongly affects the complexity of the authentication system is accessibility of the verification unit. There are two possible implementations, i.e. an authentication system with a public access to the verification unit (public authentication system) and an authentication system with a private (restricted) access to the verification unit (private authentication system). A public access to the verification unit means that anyone can acquire a verification unit from the device manufacturer. The attacker thus has access to the verification unit. It can verify the authenticity of any image and can generate any number of image-authenticator pairs. This is called the oracle attack. The strength of this attack depends on the output available to the attacker. The output can be either a binary yes/no for the whole image or it can be a bitmap with pixels or blocks indicated as authentic or tampered. Again, the attacker is interested in making undetectable changes. Another potential problem is information leakage. The attacker may be interested in obtaining some information about the secret authentication key as well

as about the placement of the watermark in the image. Many of the proposed watermarking schemes [5, 6] are not resistant to oracle attacks, as their security is based on data hiding and not on a cryptographic primitive. The attacker having access to the verification unit can launch an oracle attack and obtain secret information vital to the scheme. For a method to be resistant to oracle attacks, it should incorporate a cryptographic primitive (encryption) so that the security of the method is assured with a secret cryptographic key and not just with the secrecy of the procedure itself (placement of the watermark). For the system to be practical, a verification unit should be capable of verifying an image from any camera of the same manufacturer. This implies that the cameras and the verification units should share a limited number of pre-chosen secret keys or that a more elaborate key management scheme should be adopted. The problem of this approach is in the responsibility for the management of the secret keys which is with the manufacturer.

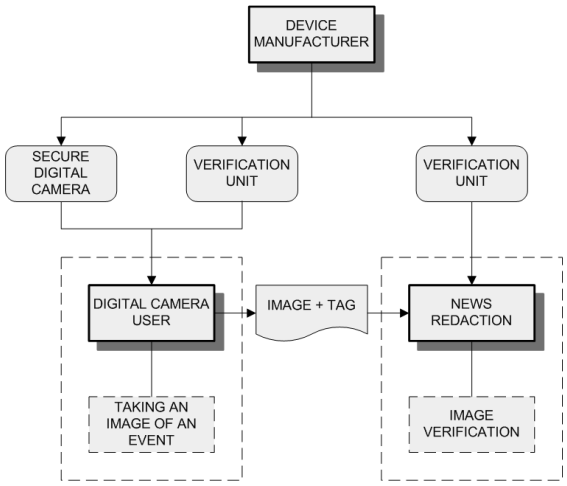


Figure 2. Public authentication system

Figure 2 illustrates the public authentication system. The digital camera and the verification unit are acquired from the device manufacturer who is responsible for providing and putting the secret key into the camera and the verification unit in a secure place in which the key is unable to be read out. For example, when taking images of an important event, the captured images can serve as digital visual information for the public. After the news redaction receives the images, it will authenticate them with the verification unit of the corresponding device manufacturer. The verification unit will corroborate or refute the authenticity of the digital image.

The private authentication system is more secure; it prevents the oracle attack and information leakage. Though the authentication method itself can therefore be simpler and easier to implement, the verification procedure is more complicated. In order to supervise verification, a trustworthy authentication (verification) center must be established. This gives rise to many questions. Who should be given the authority to manage such a center? What legal provisions should be adopted? Who will provide the required resources and financing? This altogether is the downside of the private authentication system.

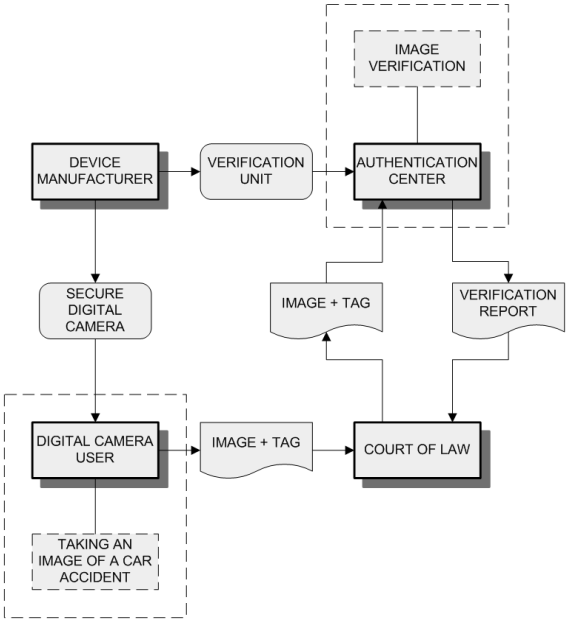


Figure 3. Private authentication system

Figure 3 illustrates the private authentication system. The device manufacturer needs to register the serial number of each camera with the authentication center in the manufacture procedure, and then the authentication center will provide the corresponding key for each camera. The device manufacturer puts the key into the camera in a secure place in which it is unable to be read out. The key is hereafter used in the authentication process. The camera being equipped with the authentication unit and with the secret key issued by the authentication center initiates the authentication process autonomously. For example, when taking images of a criminal act, the captured images can serve as digital evidence to the court. After the court receives the images, it will request the authentication center to authenticate them. The authentication center will carry out the verification procedure based on the corresponding key, indicated by the serial number of the camera. Finally, the authentication center will submit the authentication report to the court to confirm the authenticity of the digital evidence. In some applications (for instance forensics, law enforcement, court of law, judicial system), for which the state can manage the authentication center, restricted access to the verification unit is a conceivable solution. In applications (for example journalism, insurance

companies) for which the establishment of a verification center is not feasible, the verification unit must be publicly accessible. For our method to cover the widest possible range of applications, we chose to design it for the worst case scenario. In this case, a method suitable for a public authentication system is appropriate.

Even if a private authentication system does not require a method resistant to oracle attacks, it is nevertheless prudent to use it. Moreover, using it makes management of access to the verification unit less demanding.

2.3 Block-wise authentication methods

One of the first fragile block-wise watermarking schemes with tamper localization was proposed by Wong [7]. In this scheme, an image is divided into non-overlapping blocks and watermarking is performed for each block independently. The seven most significant bits (MSBs) of all pixels in a block are hashed using a secure key-dependent hash. The hash is then XORed with a chosen binary logo and inserted into the LSBs of the same block. The verification process starts in the reverse order by calculating the key-dependent hash of the seven MSBs in each block and XOR operation is performed with the LSBs. The tampered blocks can be found by comparing the output with the used logo. Block-wise authentication schemes with non-overlapping independent blocks are vulnerable to a certain kind of attack. If a set of images is authenticated with the same key, it is possible to modify an arbitrary image to be authentic. The attacker divides the image into non-overlapping blocks and for each of them performs a search in the set of authentic blocks. The original block is replaced with the most similar one to maintain the perceptual quality of the forged image. This attack is known as the Holliman-Memon attack or collage attack. There have been various countermeasures proposed in the literature [8-12] to resist the Holliman-Memon attack. In general, the countermeasures try to eliminate block independence so that block swapping is no longer possible.

2.4 Localization and watermarking

Localization is one of the features a lot of researches have been focused on. The argument for localization is that the knowledge of when and where the data has been altered can be used to infer the motive for tampering and identifying the culprits responsible. In the case of block-wise authentication schemes, we have to be very careful what kind of conclusions we draw from this information. The only thing we can be certain of is that the non-tampered parts of the image are authentic. For instance, the attacker changes a part of the image so that the verification unit fails to recognize it as authentic but the visual information of the image remains the same. The conclusions we draw from the verification results are therefore misleading. We see what the attacker wants us to see. An example is shown in Figure 4, where Figure 4a is the original image, Figure 4b shows the tampered images and Figure 4c the verification results. The plate number in Figure 4a is the same as in the lower image of Figure 4b, but the verification result shows that the image has been tampered with. Both images in Figure 4b have been tampered with, but the visual information of the lower image remained the same. Based on the verification result, we cannot infer which plate number in Figure 4b has been changed. In order to make useful conclusions based on localization of the tampered regions, the attacker must be unaware that the changes it makes will be detected. In practice, if the authentication system is available on the market, this is impossible to achieve. This means that there is no real use for localization in block-wise authentication schemes.

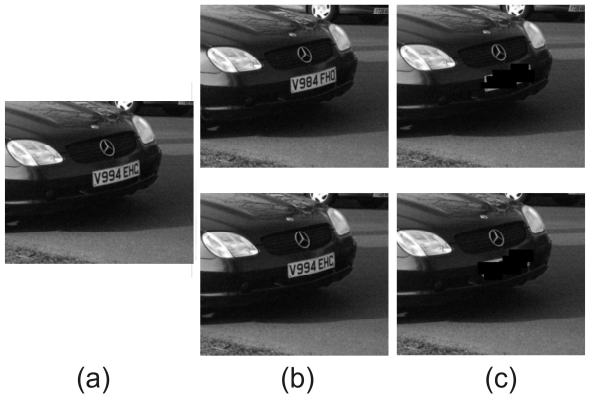


Figure 4. a.) Original image b.) tampered image c.) results of verification

The Wong's method puts the watermark in the LSBs of the image. The attacker can easily extract (change, delete) the watermark. The advantage of data hiding is therefore lost. The only advantage is that the authenticator is present in the image and not in a separate file or file header. On the other hand, the watermark changes the original image and the size of the watermark is limited. The authentication system without watermarking is easier to implement, as it is not necessary to insert the authenticator in the image (LSB plane).

3 Proposed method

In this section we propose a novel authentication method based on the Wong's localization scheme [7]. We extracted the general features from the Wong's scheme and used them to construct a general authentication scheme presented in Figure 5. The authentication scheme is in general a MAC (message authentication code) algorithm based on universal hash functions. Such an algorithm consists of two building blocks, an efficient keyed compression function that

reduces long inputs to a fixed length and an encryption function that encrypts the hash.

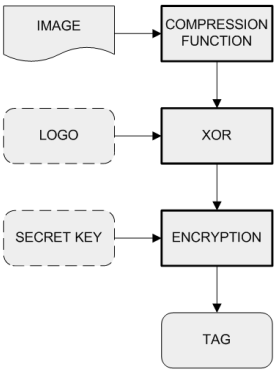


Figure 5. General authentication scheme

To construct the authentication scheme we use a family of (hash) functions $H = \{h : \{0,1\}^{\alpha} \to \{0,1\}^{\gamma}\}$, a family of secure pseudorandom functions $F = \{f : \{0,1\}^{\tau} \to \{0,1\}^{\tau}\}$ and a logo $L = \{0,1\}^{\mu}$. The scheme $\Sigma = (K, MAC)$ is defined as follows:

$$\begin{gathered} function \quad K() \\ f \leftarrow F \\ h \leftarrow H \\ return \ (f,h) \end{gathered}$$

$$\begin{gathered} function \ MAC_{(f,h)}(I,L) \\ \text{Break } I \text{ into m-bit blocks } I = I[1]\ldots I[n] \\ \text{For } i = 1\ldots n \text{ do } y_i \leftarrow f(l[i] \oplus h(I[i])) \\ Tag \leftarrow y_1 \| y_2 \| \ldots y_n \\ return \ (Tag). \end{gathered}$$

The key-space for this authentication scheme is $Key = H \times F$. A random key for the MAC is a random hash function $h \in H$ with a random function $f \in F$.

In our scheme, the input data is first compressed and then XORed with the logo. The result is further encrypted with a secret key and stored in the file header or in a separate file. To verify the authenticity of the image, the verification unit compresses the received image and XORs it with the decrypted tag. The verification unit decides if the image is authentic based on the resulting logo.

The logo proposed by Wong was either a binary image with a graphical meaning or a randomly generated black and white pattern. The idea of the logo was extended by Friedrich et. al. [12]. They proposed a symmetry structure allowing the logo to carry additional information, such as block index, image index, author ID, etc. We propose a similar approach with an improved security presented in the following section. Our logo also carries information about the block index, image index, camera id, time and date the image has been taken, camera settings (aperture, shutter speed, focal length, etc.) and possibly GPS coordinates. The additional information can help the end user determine the circumstances under which the image has been taken and make it easier to interpret the visual information of the image.

In this general scheme, the format of the input and realization of the compression function are left undetermined. The output of the scheme is a part of the authenticator or the complete authenticator. The input can vary from a single pixel to the whole image. There are several ways in which we can apply the general scheme regarding the input and output, each allowing for a different size of the authenticator and the speed of the method.

3.1 Block-wise authentication method

One of the possible implementations of the general scheme is presented in Figure 6. The encryption unit uses the 128 bit AES algorithm to encrypt the data. AES has a fixed block size of 128 bits and uses a 128 bit secret key. The reason we use a block-wise authentication method is not localization but speed. Because the pixels are read from the image sensor sequentially, it is convenient to process the image in a sequential manner. To process the image as a whole, we would have to first read the image from the image sensor and then store it in memory. This would require extra time and memory.

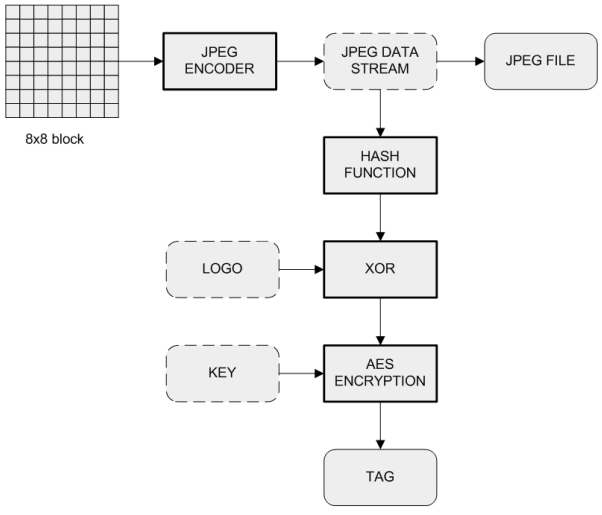


Figure 6. Authentication scheme

3.2 Compression function

The compression function consists of two steps. In the first step the block is compressed using the JPEG algorithm. The output is then passed on to a universal hash function for further processing. A universal hash function is a family of functions indexed by a parameter called the key with the following property: for all distinct inputs, the probability over all keys that they collide is small. The role of the hash function is to take the variable-sized input from the JPEG compression and return a fixed-sized output suitable for the next step in the authentication algorithm. By using the JPEG algorithm in the compression function we actually authenticate the JPEG representation of the image. The end result is an image-authenticator pair with the image in the jpg format. Our method is therefore tolerant to JPEG compression with a single quality setting. Because JPEG compression is lossy, a subsequent recompression of the image is not tolerated as it would cause the verification unit to recognize the image as unauthentic. If we wanted to authenticate an uncompressed image, we would only have to omit the first step in the compression function. The end result would be an image-authenticator pair with the image in a raw format.

3.3 Global logo

Our countermeasure against the Holliman-Memon attack is a variation of the content origin authentication [12]. The logo carries the local block information as well as parts of information which, when put together, form a global pattern throughout the image. The logo consists of three bit fields: block index, image index and content data. The block index is a unique predefined number assigned to the block. The image index is a random number generated at creation of the image and is also unique to the image. The verification unit does not know the image index, but knows that the image index is the same in all the blocks. The content data field is a fragment of the data string consisting of the camera id, time and date the image has been taken, camera settings (aperture, shutter speed, focal length, etc.) and possibly GPS coordinates. The data string is larger than the content data field, so in order to reconstruct the data string, we must join together several consecutive content data fields. The string is repeated throughout the image. The block and the image are seen as being authentic if the block index corresponds to the position of the block, the image index is the same in all the blocks and the same data string is repeated throughout the image.

4 Results and discussion

In this section we discuss security of our method and present simulation results for the Holliman-Memon attack. We also provide synthesis results for a simplified authentication unit.

4.1 Security

Security of our method is based on the Advanced Encryption Standard (AES). In cryptography, AES, also known as , is a block cipher adopted as an encryption standard by the U.S. government. It has been analyzed extensively and is now used worldwide.

The system can be vulnerable also to other types of attack. In cryptography, a channel attack-side is any attack based on information gained from the physical implementation of the cryptosystem, rather than theoretical weaknesses in the algorithm (cryptanalysis). For example, timing information, power consumption and electromagnetic leaks can provide an extra source of information which can be exploited to break the system. Many side-channel attacks require considerable technical knowledge of the internal operation of the system on which the cryptography is implemented. With careful design it is possible to limit the information emitted by the system. If the authentication unit is implemented as a stand-alone integrated circuit in the digital camera, the following side-channel attack is possible. The attacker can unsolder and move the integrated circuit to a custom-built board, where it can manipulate the inputs to the integrated circuit. It can effectively authenticate any image, even a fake one. A possible solution to the attack is to integrate the authentication unit (JPEG compression unit) into the image sensor, so that the attacker has no longer access to the inputs of the authentication unit.

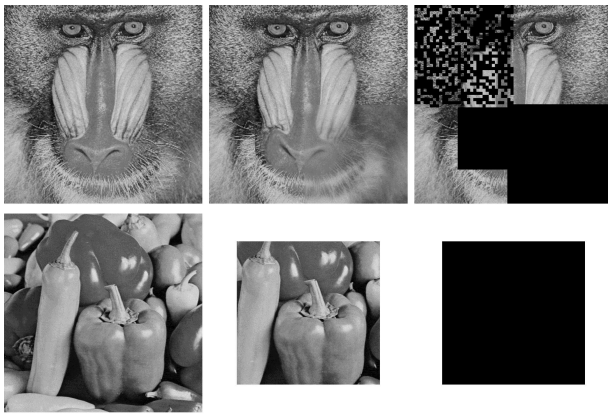
4.2 Simulation results

Our method was implemented and verified in Matlab. The verification of the method was performed on a standard test image set [13, 14]. The images from the set underwent different image manipulations (block interchanging, pixel manipulation, noise addition, low-pass filtering, cropping…). Two of the test images are shown in Figure 7a and the tampered ones in Figure 7b. In the latter, we corrupted some of the pixels and interchanged some of the blocks in the images. The results of the verification are shown in Figure 7c with the unauthentic blocks marked in black. As seen, the verification unit detects the tampered areas and the collage attack is not possible as the blocks are no longer independent.

(a) (b) (c)

Figure 7. a.) Original image b.) tampered image c.) results of verification

Figure 8 shows test images that underwent other image manipulations along with results of the verification. The pepper image in Figure 8 was cropped. The result of the verification is a black image. The mandrill image was divided into quadrants. A low-pass filter was used in the lower right quadrant. The lower left quadrant was distorted. In the upper left quadrant noise was added and the upper right quadrant was left unchanged (Figure 8b). The tampered and non tampered areas were detected correctly in all the test images.



(a) (b) (c)

Figure 8. a.) Original image b.) tampered image c.) results of verification

4.3 Implementation

The proposed method can be integrated in the JPEG compression unit already present in the digital camera, therefore consuming fewer resources. The main part of computation is required for AES encryption. The method generates an authenticator for every block. By adequately adjusting the hash function, it can compress more than one block. This enables adaptation of the size of the authenticator and the speed of the method. The only thing that is lost is the resolution of the localization.

To validate our approach in terms of the speed and size of the authentication unit, we implemented a simplified authentication unit in a programmable FPGA circuit. Our goal was to get a rough estimate on the size and speed of the authentication unit. The simplified unit is comprised of the hash function and AES encryption. The AES encryption core constitutes the main part of the authentication unit regarding the area and computation intensity. The unit was modeled using VHDL, the functional simulation was carried out using Modelsim XE III 6.3c and the design was synthesized using Xilinx ISE 9.2.i. The target device was xc3s200 from the Spartan3 family. The device utilization summary is presented in Table 1.

Device	Size/Area	Speed/Performance
XS3C200	1203 Slices (63%)	140MHz(~1Gbit/sec)

Table 1. Device utilization summary

To have the verification unit integrated into the digital camera, our primary design goal was a small area. The resulting design provides an estimate on the requirements for a workable authentication unit. Especially the design of the AES core allows for a variety of possible implementations. There have been several hardware implementations of AES [15, 16] proposed, yielding good performances. Ultimately, the best architectural decision is to select the design of the smallest possible area meeting the throughput requirement for the whole system, including the JPEG compression unit. Following the above, we can assume that the authentication unit is suitable for hardware implementation.

5 Conclusion and future work

In this paper we show that localization and watermarking properties of the block-wise authentication methods bring no significant benefit to protection of authenticity of digital images. Considering this fact, we made tag generation and verification much more efficient. The tag consists of a number of block codes carefully chosen to protect regions of interest in the image and to enable various levels of accuracy. Our logo also carries information about the camera id, time and date the image has been taken, camera settings (aperture, shutter speed, focal length, etc.) and possibly GPS coordinates. The additional information can help the end user determine the circumstances under which the image has been taken and make it easier to interpret its visual information. We propose an efficient JPEG tolerant authentication method with adjustable security that can be combined with the JPEG compression

algorithm. The system being specifically designed to tolerate JPEG compression does not tolerate other types of acceptable modifications. By skipping the first step in the compression function, the system is also capable of authenticating uncompressed images. Compared to the Wong's scheme, our scheme outperforms it with improved security and flexibility.

In our future work, we shall take an effort towards implementing the authentication unit together with the JPEG compression unit in a programmable FPGA circuit to allow for validation of our method in terms of the speed and size of the authentication unit.

6 References

[1] A. Haouzia, R. Noumeir, Methods for image authentication: a survey, Multimedia Tools and Applications, 2007.

[2] O. B. Adamo, S. P. Mohanty, E. Kougianos, M. Varanasi, VLSI architecture for encryption and watermarking units towards the making of a secure digital camera, in Proc. of the IEEE International SOC Conference, 2006, pp. 141-144.

[3] P.Blyth, J. Fridrich, Secure digital camera, in Proc. Digital Forensic Research Workshop, 2004.

[4] G. R. Nelson, G. A. Jullien, O. Y. Pecht, CMOS image sensor with watermarking capabilities, in Proc. of the IEEE International Symposium on Circuits and Systems, 2005, pp. 5326-5329.

[5] M. Wu, B. Liu, Watermarking for image authentication, in Proc. of the IEEE International Conference on Image Processing, vol. 2, 1998, pp. 437-441.

[6] R. Radhakrishnan, N. Memon, On the security of the SARI image authentication system, in Proc. of the IEEE International Conference on Image Processing, vol. 3, 2001, pp. 971-974.

[7] W. Wong, A public key watermark for image verification and authentication, in Proc. of the IEEE International Conference on Image Processing, vol. 1, 1998, pp. 455-459.

[8] N. B. Puhan, A.T.S. Ho, Secure authentication watermarking for localization against Holliman-Memon attack, Multimedia Systems, vol. 12, no. 6, 2007, pp. 521-532.

[9] P. W. Wong, N. Memon, Secret and public key image watermarking schemes for image authentication and ownership verification, IEEE Trans. Image Process., vol. 10, no. 10, 2001, pp. 1593-1601.

[10] M.U. Celik, G. Sharma, E. Saber, A.M. Tekalp, Hierarchical watermarking for secure image authentication with localization, IEEE Trans. Image Process., vol. 11, no. 6, 2002, pp. 585-595.

[11] M. Holliman, N. Memon, Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes, IEEE Trans. Image Process., vol. 9, no. 3, 2000, pp. 432-441.

[12] J. Fridrich, Security of fragile authentication watermarks with localization, in Proc. SPIE, vol. 4675, 2002, pp. 349-356.

[13] G. Schaefer, M. Stich, UCID - An Uncompressed Colour Image Database, Technical Report, School of Computing and Mathematics, Nottingham Trent University, U.K, 2003.

[14] http://www.imageprocessingplace.com/root_files_V3/image_databases.htm

[15] N.M. Kosaraju, M. Varanasi, S. P. Mohanty, A high performance VLSI architecture for Advanced Encryption Standard(AES) algorithm, in Proc. of the 19th IEEE International Conference on VLSI Design, 2006, pp. 481-484.

[16] T.Good, M. Benaissa, AES on FPGA from the Fastest to the Smallest, in Proc. of CHES, 2005, pp. 427-440.

Tomaž Nahtigal is a Ph.D. student at the Faculty of Electrical Engineering, University of Ljubljana, Slovenia. He received his B.Sc. in Electrical Engineering at the same university in 2007.His research interests include design and verification of digital systems, development of video and imaging applications and HW/SW co-design.

Andrej Žemva received his B.Sc., M.Sc. and Ph.D. degrees in electrical engineering from the University of Ljubljana in 1989, 1993 and 1996, respectively. He is Professor at the Faculty of Electrical Engineering. His current research interests include digital signal processing, HW/SW co-design, ECG signal analysis, logic synthesis and optimization, test pattern generation and fault modeling.