Image Authentication Techniques for Surveillance Applications

FRANCO BARTOLINI, MEMBER, IEEE, ANASTASIOS TEFAS, STUDENT MEMBER, IEEE,

MAURO BARNI, MEMBER, IEEE, IOANNIS PITAS, SENIOR MEMBER, IEEE

M. Barni is with the Department of Information Engineering, University of Siena, Via Roma 56, 53100 - Siena, Italy, (phone: +39 0577 234621; Fax: +39 0577 233630: e-mail: barni@dii.unisi.it).

F. Bartolini, is with the Department of Electronics and Telecommunications, University of Florence, Via S.Marta 3, 50139 - Firenze, Italy, (phone: +39 055 4796380; Fax: +39 055 494569; e-mail: barto@lci.die.unifi.it).

I. Pitas and A. Tefas are with the Department of Informatics, University of Thessaloniki, 54006 Thessaloniki, Greece (Phone,Fax: +3031 996304, e-mail: pitas@zeus.csd.auth.gr).

Abstract

In automatic Video Surveillance (VS) systems, the issue of authenticating the video content is of primary importance. Given the ease with which digital images and videos can be manipulated, practically they do not have any value as legal proofs, if the possibility of authenticating their content is not provided. In this paper, the problem of authenticating video surveillance image sequences is considered. After an introduction motivating the need for a watermarking-based authentication of VS sequences, a brief survey of the main watermarking-based authentication techniques is presented and the requirements that an authentication algorithm should satisfy for VS applications are discussed. A novel algorithm which is suitable for VS visual data authentication is also presented and the results obtained by applying it to test data are discussed.

Keywords -*Automatic Video Surveillance. Watermarking. Video Authentication.*

I. INTRODUCTION

Recent advances in electronic and telecommunication technology, together with the development of increasingly powerful signal and image processing techniques, considerably broadened the scope and quality of automatic Video Surveillance (VS) systems. Research is currently being carried out by several industrial and academic institutes to improve automatic surveillance in terms of continuous and effective monitoring, cost reduction and reliable control of remote or dangerous sites [1,2,3,4]. At the same time, it is possible to significantly enlarge the application scenarios of VS systems, by taking advantage of the diffused availability of wide area networks such as the Internet, e.g. towards residential users or their use in small commercial buildings [3]. However, some practical issues must be taken into account, in order to take full advantage of the potentiality of VS systems. The validity of visual data acquired, processed and possibly stored by the VS system, as a proof in front of a court of law is one of such issues. As a matter of fact, due to the availability of ever more sophisticated processing tools, and their consequent use in modifying the visual data content without leaving any visible traces of the modification, digital video/image data have virtually no value as legal proofs, since doubts would always exist that they had been intentionally tampered with to incriminate or exculpate the defendant. In addition, the possibility of proving the true origin of data must be considered, e.g. by associating to each meaningful piece of information a label indicating where and by whom it has been produced. By data authentication we mean here a procedure capable of ensuring that data have not been tampered with (data integrity) and of indicating their true origin.

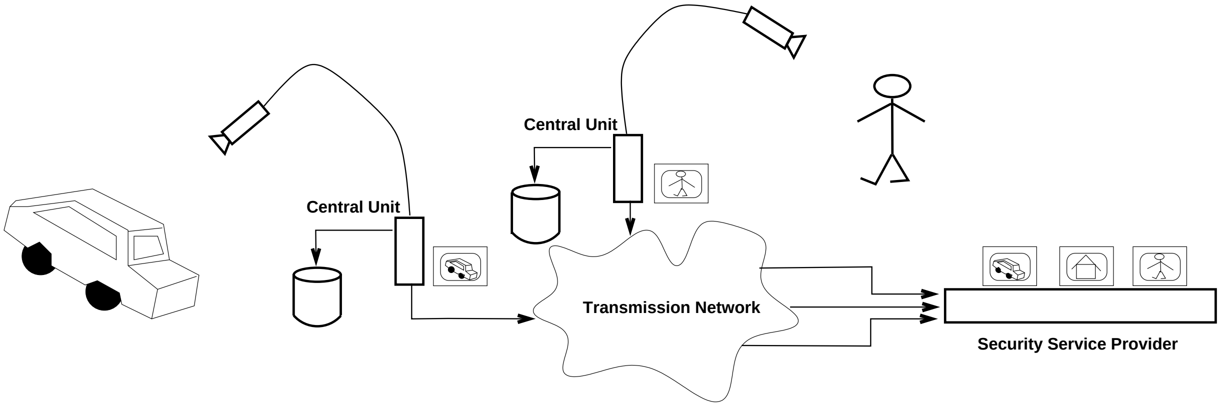


Fig. 1. Simplified model of a VS system. Images acquired by the digital cameras are first processed by a central unit to look for alarm situations. The video sequence could then be transmitted to an intervention center. The possibility to store the video sequences is also considered.

To analyze the most important issues in VS data authentication, we will refer to the simplified model reported in Figure 1. A pool of digital cameras frame the scene of interest and send the corresponding video sequences to a central unit. The central unit is in charge of analyzing the sequences and generating an alarm whenever a suspicious situation is detected. The alarm, as well as part of the video sequences generating the alarm, may be transmitted to an intervention center such as a security service provider, the police department or a security guard unit. Video transmission may avail of existing transmission facilities, e.g. Internet, or use a transmission system expressly installed to serve the VS system. Somewhere in the system, the video sequence or some parts of it may be stored for subsequent use. When needed, stored sequence can be used as a proof in front of a court of law. As said before, if the stored digital video sequences have to be legally credible, some means must be envisaged to deter or detect content tampering, and to reliably trace back to the data origin.

A. Cryptographic data authentication

A straightforward way to provide video authentication is by cryptography means, namely through the joint use of asymmetric-key encryption and a digital hash function [5,6]. Let us assume that each video camera is assigned a different public/private key pair, with private keys hardwired within the cameras (which of course should be as tamper-proof as possible). Before sending the video sequences to the central unit, cameras calculate a digital summary (digest) of the video by means of a proper hash function. Then, they encrypt the digest with their private key, thus obtaining a signed digest which is transmitted to the central unit together with the acquired sequences. Later on, the digest can be used to prove data integrity or to trace back to their origin: one only needs to read the signed digest by using the public key of the camera which produced the video and check if it corresponds to the actual video content.

Added value can be provided by tying each frame to the particular label of the instant the frame has been produced at. This can be achieved by "printing" the date and time of creation in each frame. Note that any modification of either the date or the time would be easily revealed, since it would change the image digest.

To ensure integrity at the frame level, the above procedure must be applied to each frame of the sequence. When the integrity of the entire video has to be checked, the digest should be computed on the entire sequence or on suitable sub-parts. This may be a problem, since it could require computation and storage requirements that are currently unavailable at the video camera side. In addition, a considerable delay would be introduced, that is not allowed in systems where the timely generation of alarms is of major concern. The introduction of a serial number within the frame along with the acquisition time and date is an alternative solution, since it would permit to easily detect operations such as the removal or flip of two or more frames. Note that the substitution of a true frame with a fake one is not possible without using the private key of the video camera.

B. Watermarking-based authentication

Though cryptography may provide an adequate solution to the authentication of VS data in many practical cases, the development of alternative approaches is desirable in order to deal with some potential weaknesses of the cryptographic approach [7]. Suppose, for example, that the digest-based approach outlined in the previous section is adopted to provide data authentication. This approach requires that the image digest is tied to the image itself, e.g. by defining a proper image format envisaging the use of image authentication tools (see for example the MPEG21 effort of ISO/IEC JTC1/SC29/WG11). In this way, however, the possibility of authenticating the image is constrained to the use of a particular image format, thus making impossible to use a different format, or to authenticate the image after printing. This is not the case with watermarking-based authentication, since authenticating data is embedded within the image itself.

Another drawback with digest-based systems is that the digest changes dramatically as soon as any modification, be it a small or a large one, is applied to the image [5]. Thus, it is impossible to distinguish between malicious and innocuous modifications. Suppose, for instance, that the video sequence has to be compressed before transmission or storage. In the previous scheme, compression must be performed by the video camera before computing the digest, otherwise the correspondence between digest and data would be lost. Very often, however, the camera does not have enough computing power to compress the video sequence, and compression must be performed by the central unit, thus making the digest computed before transmission useless. Another case where the distinction between malicious and innocuous manipulations is of primary importance is in privacy-compliant VS systems. According to the legislation of many countries (e.g. those in the European Union [8]), a VS system can acquire, process and store video sequences only for a limited time and only if the identity of the involved individuals can not be identified. Usually this is not a problem, since a VS system does not need to know the identity of a person and is only intended to identify behaviors, to localize people or objects, and to classify events. However, the necessity to respect privacy laws may require that some parts of the scenes (e.g. the faces of individuals) are obscured before storage, thus nullifying the utility of the pre-computed digest. Similar considerations hold, whenever the video sequence has to be processed before storage.

Watermarking-based authentication represents a feasible solution to the above problems [9,10,11,12,13,14]. A digital watermark [15,16] is a signal that is imperceptibly embedded within digital data (audio, images, video, text). Though imperceptible, such a signal can be detected or extracted by means of computations in order to make some assertions about the host data. The characteristics of a watermarking system largely depend on its application scenario [17,18,19]. For instance, copyright protection applications require that the watermark is robust against most common data manipulations, i.e., its presence can still be detected after non-destructive transformations of the host document.

With regard to data authentication, two possible approaches are possible [7,20]: to use a fragile watermark or a robust one.

Fragile watermarking refers to the case where the watermark inserted within the data is lost or altered as soon as the host data undergoes any modification: watermark loss or alteration is taken as an evidence that data has been tampered with, whereas the recovery of the information contained within the data is used to demonstrate data origin. Interesting variations of the previous paradigm, include the capability to localize tampering, or to discriminate between malicious and innocuous manipulations (e.g. moderate image compression). In the latter case, the watermark is said to be semi-fragile, since it survives only a certain kind of manipulations.

The use of robust watermarking for data authentication relies on a different mechanism: a summary of the candidate frame or video sequence is computed and is subsequently inserted within the video sequence itself by means of a robust watermark. Information about the data origin is embedded together with the summary. To prove data integrity, the information conveyed by the watermark is recovered and compared with the actual content of the sequence: their mismatch is taken as an evidence of data tampering. The capability to localize the manipulations will depend on the accuracy of the summary which is embedded into the image. As will be better discussed in Section II, a precise summary requires hiding many bits, thus causing the watermark to be less resistant to manipulations. The authentication of the data origin label is immediate since such an information can be reliably hidden within data.

The main differences between the cryptographic and watermarking approaches to authentication can be summarized as follows: (a) in watermarking no header or separate file is used to ensure data integrity, (b) distinction is possible between different kinds of modifications and (c) watermarking-based authentication allows to localize modifications occurred to an image or a video sequence under test.

Conversely, the main drawbacks of watermarking-based authentication derive from the relative immaturity of watermarking technology with respect to cryptography. Some open questions must still be answered with reference to the ultimate level of security and robustness achievable through watermarking and to the number of bits that can be reliably hidden within an image of given size. Depending on the desired robustness, in fact, the watermark payload for a 512×512 image varies from some hundreds of bits to several Kbits. Some concerns regard computational complexity as well. Existing algorithms range from very fast algorithms, operating in the spatial or the compressed image domain, to very complex ones requiring the application of time consuming full frame transforms (e.g. full frame DFT or DCT), thus such an issue must be carefully considered when choosing the watermarking technology to be used for a given application.

In this paper the main issues in watermarking-based authentication of VS sequences are reviewed. First, a brief review of the state of the art of watermarking technology is given as far as image/video authentication is concerned (section II). Particular attention is given to the distinction between (semi-)fragile and robust watermarking: the advantages and drawbacks of each class of techniques will be highlighted and the open issues yet to be solved will be pointed out. In section III, the authentication problem is casted in the VS framework. The peculiarities of the requirements set by this particular application are discussed, and the possible solutions are analyzed in the light of the current technology. A novel watermarking algorithm for data authentication belonging to the semi-fragile category is presented in section IV and its effectiveness in a VS scenario is evaluated in section V. Some conclusions are finally drawn in section VI.

II. WATERMARKING IN BRIEF

The techniques for secret information hiding in an host document have received increasing attention since, at the beginning of nineties, they were proposed either for data authentication [9,10] or for data protection from unauthorized copying and distribution [21,22,23,24,25]. Later on, attention was mainly paid to the use of digital watermarking for copyright protection. To this aim, research was mainly focused on the development of robust watermarking algorithms, since robustness turns out to be one of the main requirements for the effective use of watermarking in copyright protection applications [17]. Recently, researchers paid a renewed attention to (semi-)fragile watermarking.

Regardless of the application, image/video watermarking can be seen as a communication task consisting of two main steps: watermark casting, in which the signal, represented by the watermark, is transmitted over the channel (original image) and watermark detection, in which the signal is received and extracted from the possibly corrupted image. Intentional/unintentional attacks and distortions applied to the image further complicate the transmission channel used. In watermark casting, an encoder function $\mathcal{E}$ takes a host image $\boldsymbol{f}$ and a watermark $\boldsymbol{w}$, and generates a new image $\boldsymbol{f}_w = \mathcal{E}(\boldsymbol{f}, \boldsymbol{w})$, called the watermarked image.

The function $\mathcal{E}$ embeds the watermark $\boldsymbol{w}$ into a set of features $\boldsymbol{F}(\boldsymbol{f})$ by means of a proper insertion operator.

According to the set of features the watermark is injected into, watermarking techniques can be divided into three main categories: (i) *spatial domain techniques* directly add the watermark to pixel values. They allow easy exploitation of the Human Visual System (HVS) characteristics for better data hiding; (ii) *transform domain techniques* add the watermark to the coefficients of a full-frame image transform (e.g., DFT, DCT, Mellin, Radon, Fresnell). These techniques do not allow precise localization of the watermarking perturbations on the image, and thus, make it difficult to tune it to the HVS characteristics; (iii) finally, *hybrid techniques* (mainly using block-wise DCT, and wavelets) are those working in a transform domain, but without completely loosing spatial localization. Despite their difficulty in exploiting the HVS system for watermark concealment, transform domain techniques usually exhibit a higher robustness against attacks. Hybrid techniques (in particular wavelet-based ones) try to trade off between the advantages of spatial domain techniques in the localization of the watermarking perturbation and the good resistance to attacks of methods operating in the transform domain.

Second generation watermarking algorithms are currently being developed that embeds the watermark by modifying a set of *semantic* features such as regions, boundaries and object characteristics [26,27]. In addition to their compatibility with new coding standards (e.g., MPEG-4, JPEG2000), second generation methods may be designed in such a way that selective robustness to different kinds of attacks is obtained, thus improving considerably watermark flexibility.

The differences between watermarking algorithms to be used in various applications come into play when one analyzes the problem at the protocol level, and when the requirements on the watermarking algorithm are taken into account [17,18]. As a matter of fact, a general treatment of watermarking is virtually impossible without considering application and protocol issues. For example, copyright protection demands that the watermark survives at least the most common image manipulations. Conversely, when a watermark is used for data authentication, requirements on robustness depend on the protocol used for data authentication. In fact, as presented already in section I, either a robust watermark or a fragile one may be used. Furthermore, a semi-fragile approach may be considered. In this case the class of manipulations that have to be classified either as innocuous or malicious depends, once again, on the application.

Among the characteristics of image watermarking algorithms, a crucial role is played by the way the watermark is extracted from data [17,18,28,29]. In *blind watermarking*, the decoder does not need the knowledge of the original image, or of any information derived from it, to recover the watermark. Conversely, *non-blind* decoding refers to the case where extraction is accomplished with the aid of the original, non-marked data. Blind techniques are sometimes referred to as *oblivious* [30]. In spite of the benefits it gives in terms of robustness, non-blind decoding is undesirable in many applications, where the availability of the original data to the receiver cannot be granted. For example, it is clear that as long as data authentication is concerned, the adoption of a blind approach is mandatory.

A general framework for describing watermarking techniques for images authentication is here given. During the embedding phase, a watermarking signal is generated through a suitable watermark generation function $\mathcal{G}$, and based on a secret key K and, possibly, on the to be authenticated image $\boldsymbol{f}$; this watermarking signal is then hidden into the image data:

$$\begin{aligned} &\boldsymbol{w} = \mathcal{G}(\boldsymbol{f}, K) \\ &\boldsymbol{f}_w = \mathcal{E}(\boldsymbol{f}, \boldsymbol{w}). \end{aligned} \tag{1}$$

In the verification phase, the watermark ($\boldsymbol{w}$') is computed again for the image $\boldsymbol{f}'_w$ by means of the same function $\mathcal{G}$. The watermark $\boldsymbol{w}$" is then extracted from the to-be-tested image $\boldsymbol{f}'_w$ by means of the watermark decoder $\mathcal{D}$; the two watermark are finally compared to verify authenticity:

$$\begin{aligned} &\boldsymbol{w}' = \mathcal{G}(\boldsymbol{f}'_w, K) \\ &\boldsymbol{w}'' = \mathcal{D}(\boldsymbol{f}'_w) \\ &\text{If} \quad \boldsymbol{w}' = \boldsymbol{w}'' \quad \text{Then} \\ &\qquad \text{Image is authentic} \\ &\text{Else} \\ &\qquad \text{Image is fake.} \end{aligned} \tag{2}$$

This general framework is valid both for (semi-)fragile and for robust watermarking-based authentication. The differences among the two approaches is on the mechanism they rely on for detecting manipulations: while fragile techniques assumes that any manipulation (or some strong enough manipulations for the case of semi-fragile methods) modify the embedded watermark, robust techniques assumes that the watermark is not affected by any manipulations, it is the watermark generation function that, in this case, produces for the manipulated image a different watermark, not corresponding to the embedded one. In a more formal way we can say that for (semi-)fragile techniques we expect that:

$$\begin{aligned} &\boldsymbol{w}' = \boldsymbol{w} \\ &\boldsymbol{w}'' \neq \boldsymbol{w} \end{aligned} \tag{3}$$

while for robust techniques:

$$\begin{aligned} &\boldsymbol{w}' \neq \boldsymbol{w} \\ &\boldsymbol{w}'' = \boldsymbol{w} \end{aligned} \tag{4}$$

In the sequel, a brief overview of the main robust and fragile watermarking algorithms proposed so far for images authentication is given. For a more comprehensive description of watermarking state-of-the-art, readers may refer to a number of excellent papers surveying this field [15,31,32,33].

A. Fragile watermarking

Historically, the fragile approach was the first to be considered for visual data authentication [9]. For example, Van Schyndel et al. [25] and Wolfgang et al. [11], add small amplitude, maximum length sequences to the images: any modifications of the pixel values will alter the embedded sequences, thus allowing for tamper detection and localization. The added watermark does not depend, in this case, on the image (i.e., by referring to the general framework previously presented function $\}$ solely depends on the key K). Given that the added watermark only influences LSBs, the system is not resistant to compression; furthermore the embedded LSBs can be easily extracted thus obtaining a fake watermark to be added to another manipulated image (collage attack [20]).

To avoid this problem, Wong [34] substitutes the LSBs of a block of pixels by the hash value computed on the corresponding MSBs (i.e., by again referring to our general framework, they make function $\mathcal{G}$ also depend on the image): localization is still achieved, but resistance to compression is not. Yeung et al. [12] modifies image pixels in such a way that, when mapped through a key-dependent function, they produce a bitmap image which is equal to an authentication logo. The use of a logo allows a human observer to visually evaluate the amount of manipulation the image has undergone. Anyway, it has been demonstrated that this method has a serious security gap: if the same logo is embedded in two different images, the mapping function can be recovered almost exactly, thus allowing the creation of forged images [35].

An attempt to realize a fragile authentication system allowing to get qualitative information about the characteristics of the occurred manipulations is presented in [13]. The wavelet decomposition coefficients are suitably quantized for embedding a key-dependent binary watermark: during the watermark detection phase, the resolution level of the wavelet decomposition where a mismatch between the embedded and the retrieved watermark is found, gives some information on the frequency characteristics of the manipulations that affected the image. In particular it is demonstrated that resistance to low compression factors can be achieved. Lin et al. [36] embed a pseudo-random sequence into the DCT coefficients of the image blocks; for authentication, the pseudo-random sequence is checked for presence through correlation. In order to increase the reliability of detection while using a very short sequence, pixel and sequence values differences (instead of original values) are correlated : this allows the image energy to be rejected, and detection to be more robust.

Most of the (semi-)fragile techniques that can be found in the literature are of the spatial or hybrid type: in this way it easier, in fact, localise manipulations, by referring to those areas of the image where the mismatching samples of the watermarking signal come from.

B. Robust watermarking

In this approach, is the watermarking generation function $\mathcal{G}$ itself that has to be sensitive to image manipulations. This characteristic make the $\mathcal{G}$ function to be similar to the classical cryptographic hashing functions: in practice, the watermark is a sort of summary (digest) of the image. Two issues are worth mentioning. First of all the capacity to localize manipulations depends on the accuracy of the embedded summary: more accurate summaries require more bits to be hidden. Thus they are less resistant to innocuous manipulations. As a matter of fact, the issue of how many bits can be reliably hidden into an image is still open, although some results begin to appear [37,38,39]. Second, effective summary computation techniques have to be defined which should be invariant to innocuous manipulations (in particular to lossy compression), but still accurate enough to detect malevolent forgeries. Some examples of summary extraction procedures can be found in [40], where a digest is computed by considering the differences between pair of corresponding coefficients in neighboring blocks; in [41], where statistical moments and edge maps are used for building an image summary; and in [42] where salient point co-ordinates are fed to an hash function for producing a compression invariant digest.

To resist lossy compression, Wu et al. [43] and Xie at al. [44] embed the data summary into the images by explicitly referring to the structure of the compression algorithms based on block DCT and wavelet transform respectively. In particular in [43] the JPEG-quantized DCT coefficients of each block of the image are modified in such a way to map to the bit to be embedded while put as input of a suitable look-up table. Tampering localization can also be performed, because, further to the content summary, a visually meaningful binary pattern (i.e. a logo) is also embedded. Similarly in [44] each wavelet coefficient is modified according to the value of the to be embedded bit and of the quantization threshold. It is just the exploitation of knowledge about the quantization thresholds used by the compression algorithms that makes these two methods resistant to image lossy compression.

Fridrich [45] proposes a robust block-based visual hashing function and uses a robust watermarking technique for embedding the hash values on each block. The hashing function computation works in this way: the projections of each block of data into a set of smoothed randomly generated blocks are first computed; the projection values are then binarized by thresholding to produce a summary codeword for each block. The resulting hashing function has good characteristics, being able to produce uncorrelated summary codewords for different blocks, and almost identical codewords for similar looking blocks.

A promising approach to image authentication consists of those techniques allowing to embed a low resolution image into a higher resolution one. In particular, in [46] a low resolution copy of the image is embedded into the LSBs of the image itself. Authenticity checking can be performed by comparing the image under test with its low resolution version extracted by watermark decoding. Similarly, in [47] a low resolution copy of the image (in particular the low-pass band of a wavelet decomposition) is hidden into the image by means of a spread spectrum watermarking technique. Both these methods were able to resist to JPEG at low compression factors. The success of robust watermarking for data authentication highly depends on the possibility of reliably embedding large amount of data into an image (see [48] for an example of high capacity watermarking).

C. Robust vs (semi-)fragile watermarking

Having outlined the main features of (semi-)fragile and robust watermarking, we can now discuss the advantages and drawbacks of these two technologies.

Fragile and semi-fragile watermarking technology is more mature than robust-watermarking-based authentication , since it has been studied for a longer time. Many algorithms have been proposed so far that can be used in different application scenarios. Another advantage of (semi-)fragile watermarking is the ease with which tamper localization is achieved. On the negative side, with fragile and semi-fragile techniques it is rather difficult to distinguish between malicious and innocuous manipulations. Once the classes of allowed and non-allowed manipulations have been defined (a complex, application-dependent task by itself), it is very difficult to design a semi-fragile watermarking algorithm that survives all the manipulations of the first class, but it is destroyed by the manipulations belonging to the second class.

Image authentication by means of robust watermarking is very promising, especially with regard to the distinction between innocuous and malicious manipulations. Nevertheless, the research in this field is still in its infancy, thus it is difficult to judge, right now, the feasibility of such an approach. With reference to the current state of the art, the two most challenging issues of robust-watermarking-based authentication are the design of an effective visual hashing function and the possibility of hiding a large number of bits into the image without loosing robustness.

III. VS DATA AUTHENTICATION

In order to highlight the peculiarities of VS data authentication, let us consider the most common requirements for watermarking-based authentication techniques [17]:

1. The authentication technique must not deteriorate the visual quality of data quality.
2. The authentication technique should be able to identify any non-authorized (assumed as malicious) processing occurred to the visual data.
3. The authentication technique should not consider innocuous manipulations, e.g. image compression or zooming, as valid authentication attacks.
4. It should be difficult for non-authorized persons to forge an authenticated image.
5. The authentication checking procedure should be easily performed by authorized persons.
6. The authentication checking procedure should localize data tampering.

Let us now re-consider these requirements for the particular case of the authentication of VS data.

A. The invisibility constraint

The requirement on the deterioration of the visual quality of authenticated data is usually referred to as the invisibility constraint. In the VS case, however, this is not a crucial issue, since VS data do not exhibit a quality comparable to that of visual data used in arts/media. Furthermore, VS data are usually acquired by inexpensive, low quality devices and in highly un-controlled environment, thus being of low visual quality. In addition, the visual analysis that would possibly be carried out in a law court will focus on the semantic content of the images, rather than on their VISUAL quality.

Nevertheless, in VS systems, there is a critical observer, namely, an Automated Visual Inspection (AVI) system. Thus, it is necessary to investigate what invisibility means from the AVI point of view.

An AVI system usually works at two levels: a low level, where the images acquired by the camera are processed to detect moving objects, and a higher level where the behavior of moving objects is analyzed, in order to understand if an alarm situation is currently present. Let us consider first how low level processing could be affected by the authentication process. As a first consideration, it could be said the allowable modification level is related to the noise level tolerable by the AVI system. Given that AVI systems have to operate in uncontrolled environments, they usually exhibit a high degree of robustness with respect to image perturbations. These considerations permit to conclude that the invisibility requirement is more relaxed in the VS framework.

To move the analysis one step further, the design of a watermarking algorithm for VS data authentication could follow two different guidelines.

A first possibility derives from the observation that if the embedded signal (i.e. the embedded noise pattern) is the same for all video frames, the presence of the watermark will not disturb the correct functioning of the system in any way, given that AVI techniques aim at detecting differences between frames. Of course, this would imply that the watermark is neither frame-dependent, nor time-varying (which means that it will not be able to help in maintaining the correct sequential frame order). The main drawback of this approach is that a frame-independent watermark can be easily found by a comparative analysis of all image sequence frames, and then could be easily added again to fake frames.

As a second possibility, it has to be considered that low level image processing techniques usually employed in AVI systems, largely exploit the time coherence of natural object motion (see for example [49]); this means that a certain degree of correlation is assumed to be present between subsequent frames, and uncorrelated frame differences are filtered out as noise. Thus, it is important that watermarks embedded in successive frames are highly uncorrelated, in order to avoid producing misinterpreted moving objects, thus originating false alarms. Let us, for example, consider a spatial-domain watermarking technique such as that proposed in [50], which embeds a binary code in an image by using a spread-spectrum amplitude modulation, and let us assume that the sequential number of each frame has to be embedded within the frame. The technique is by itself image-independent, but the watermark becomes frame-dependent because it contains the frame number. Given that frame numbers of subsequent frames are highly correlated, it is important that the spreading sequence is changed from frame to frame, to avoid confusing the AVI system.

The considerations regarding the opportunity of using watermarks which are highly uncorrelated among frames, are still valid when the watermark effect on high level processing stages is considered. High level reasoning procedures usually examine the moving objects detected by low level analysis and try to estimate their trajectories in subsequent frames. A strong assumption underlying such procedures is related to motion coherence: A moving object should follow a naturally smooth trajectory. This assumption allows a drastic reduction of noise effects. With high probability, possible fake moving objects due to the watermark will not exhibit a natural motion, and thus, will not be interpreted by high level analysis as true moving objects.

However, the most important consequence of the particular meaning of the transparency in VS, is that stronger (and thus more robust) watermarking procedures can be applied in this case than in classical authentication applications [9]. This will surely boost their acceptance as legal proofs. In [7], for example, the invisibility constraint is relaxed, thus allowing to obtain a more robust authentication scheme which resists moderate JPEG compression. Of course the strength of the watermark should not seriously degrade the image, thus reducing its value as a legal proof.

B. Malicious vs innocuous manipulations

Requirements 2 and 3 are linked together, since they refer to the possibility of distinguishing between innocuous (allowed) and malicious (not-allowed) manipulations. Two VS scenarios in which the necessity to distinguish between innocuous and malicious processing comes into play are the following. The first situation refers to a system in which data compression is performed by the central unit. Under such an assumption, the authentication is performed before compression and therefore must survive it. Another important case is related to the necessity of satisfying privacy compliance. To respect privacy it is, in fact, probable that some processing is performed on the authenticated video, before it is stored, e.g. for obscuring the faces of persons which are un-important for the AVI system. In this case, this kind of processing has to be considered as innocuous. On the other side the same processing procedure has to be considered as malicious when information that is important for a court law is removed. A solution to this chicken and egg problem is offered by authentication techniques able of localizing manipulations (requirement 6): once the modification is precisely localized, it will be up to the court law to decide if it is malevolent or innocuous.

As it is shown in the above examples, an clear distinction between allowed and non-allowed manipulations is very difficult. In the second part of this work. we will propose a new semi-

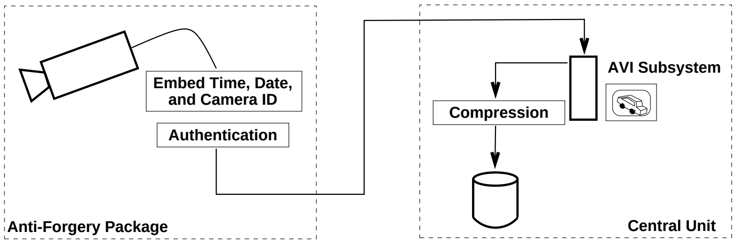


Fig. 2. Watermark-based authentication for automatic VS. The case of raw data authentication is considered. The authentication tool should be transparent to the AVI system processing methods and resistant to compression.

fragile authentication scheme which is robust against moderate JPEG compression. As to the other kinds of manipulations, the proposed scheme is able of distinguishing between global and localized manipulations, and to extract areas that were severely modified. It will be up to a court of law to judge whether the extent and the importance of manipulated areas is such that the image proof value is nullified or not.

C. Other requirements

The requirement that it should be very difficult to produce a forged authenticated image (requirement 4) must be strictly satisfied if the system has to have any value. The easiest and safest way to fulfill this requirement is through the use of key-based encryption; for example, if the watermark is signed with the private key of the video camera it is impossible to reproduce it, while anyone is allowed to read it.

Ease of authenticity checking (requirement 5) is not of major concern here, since it is likely that VS data authentication is performed only when the video sequence has to be used in front of a court law. However, applications where the central unit authenticates the video sequence before analyzing it, can be thought of: in these cases, indeed, authentication checking should be as straightforward as possible.

We already discussed the necessity of localizing tampering (requirement 6): this, in general, is not mandatory by itself in VS applications, but, as we have seen while discussing requirements 2 and 3, it gives a useful indication of where alterations have taken place.

D. VS-specific requirements

Along with the general purpose constraints discussed so far, some VS-specific issues must be considered. In cases where compression of the video sequence is performed by the camera, it may be desirable to watermark the compressed video stream, so that robustness to compression is not a concern anymore. However, under such an assumption, cryptography-based authentication may be used, unless a stringent demand for the localization of tampering is raised. In some cases the video camera may not be powerful enough to compress the sequence. To overcome the problems due to the limited processing capabilities at the camera site, a PC station may be added right after the camera output; however, as PC stations are usually not tamper-proof, watermarking must still be performed inside the camera. As a consequence watermark embedding should not have high computational demands and should be compression resistant.

As a second issue, the need to ensure video data integrity as a time sequence, and to tie the video sequence to the time and date it has been produced must be considered. Though many solutions are possible, the easiest way to detect the removal of one or more frames or the alteration of the original frame order, is embed on each image a serial number before authentication tools are applied. In fact, such a number can neither be removed nor modified without affecting the authentication check, thus making it impossible to remove or change the position of any frame of the sequence. In the same way, by embedding in each frame the time and date of its creation, an indissoluble link is created between the sequence content and a time instant, so that the legal value of the sequence is completely preserved. Another possible solution is to make the embedded watermark depend on the frame number and to bear time information: frame exchange or substitution would thus be easily detected, and acquisition time can be reliably extracted from watermarked data.

It is also worth mentioning that for an authentication system to be valuable in front of a court law, a Trusted Third Party must be established, which is in charge of storing the camera keys and the detection algorithm, in order to perform the authenticity check.

In Figure 2, a sketch of a simple VS system in which watermarking is used to authenticate VS data in its raw form is given. Time, date and frame serial number are overwritten to every single frame before authentication. The authenticated sequence is processed by a central unit for detecting pre-alarm situations, and then is compressed for storage purposes. In this case, authentication tools should be transparent to the image processing algorithms applied by the central unit and resistant to compression.

IV. A WATERMARKING ALGORITHM FOR VS DATA AUTHENTICATION

A watermarking algorithm for VS data authentication based on semi-fragile watermarking of each frame of the video sequence is thoroughly described in this section. In the following, the three steps (i.e. watermark generation, embedding and detection) of a general watermarking-based authentication technique are resumed, with particular reference to the proposed approach. Let $\mathcal{R}$ and $\mathcal{Z}$ denote the set of real and integer numbers, respectively and $\mathcal{U} = \{0, 1\}$ or $\{-1, 0, 1\}$. Given an image $f(\boldsymbol{x}) : \mathcal{B} \subseteq \mathcal{Z}^2 \rightarrow \mathcal{Z}$ and a watermark key $K \in \mathcal{Z}$, a watermark signal $w(\boldsymbol{x}) : \mathcal{M} \subseteq \mathcal{Z}^2 \rightarrow \mathcal{U}$ can be produced using the watermark generation procedure $\mathcal{G}$:

$$\boldsymbol{w} = \mathcal{G}(\boldsymbol{f}, K) \tag{5}$$

In watermark generation, either chaotic techniques or pseudorandom number generators can be used for producing the watermark $w(\boldsymbol{x})$. The embedding $\mathcal{E}$ of the watermark $w(\boldsymbol{x})$ in the image $f(\boldsymbol{x})$ is given by:

$$\boldsymbol{f}_w = \mathcal{E}(\boldsymbol{f}, \boldsymbol{w}) = \boldsymbol{f} \otimes \boldsymbol{w}, \tag{6}$$

where $\boldsymbol{f}_w$ is the watermarked image and $\otimes$ is a generalized superposition operator, which includes appropriate data truncation and quantization, if needed. Watermark detection $\mathcal{D}$ has twofold aim: (i) to produce a binary decision on image authenticity and (ii) to detect the tampered image regions if any. In the proposed method, watermark detection produces an image denoting the tampered (non-authentic) regions of the watermarked image.

A. Watermark Generation and Embedding

Watermark generation aims at producing a binary or ternary watermark $w(\boldsymbol{x})$ using a digital key K and a host image $f(\boldsymbol{x})$. The watermark key corresponds to the image owner or the camera that has captured the image. In the sequel the term "host image" $f(\boldsymbol{x})$ will be used when we are referred to the video frame to be marked. The watermark signal is a sequence of random data which can be produced by using a pseudorandom number generator, filtering for obtaining certain spectral properties and appropriate thresholding. Alternatively, the watermark sequence can be generated by successive iterations of a chaotic map applied to an initial state. In this case, the watermark sequence can be produced having pre-specified spectral characteristics that are controlled by varying the parameter of the chaotic map and uniform distribution [51]. It has

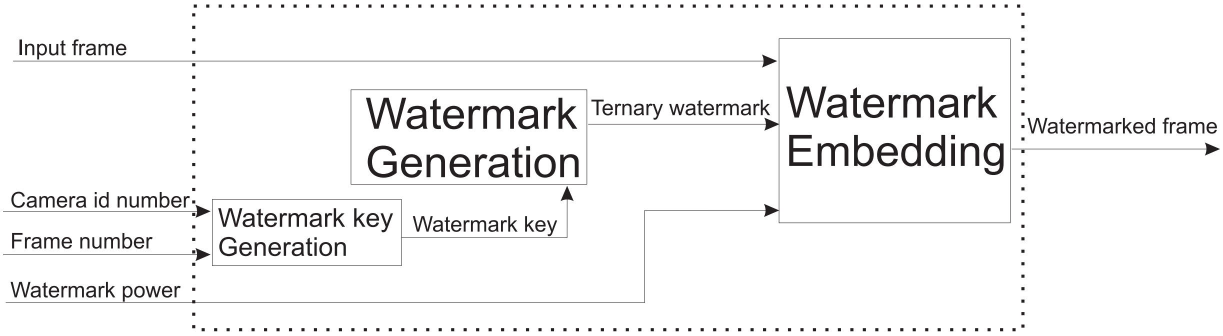


Fig. 3. Watermark generation and embedding.

been proven that chaotic sequences having certain first and second order correlation statistics can perform better than filtered pseudorandom sequences having the same spectral characteristics [52]. The spectral properties of the watermark signal play an important role on the watermarking system performance since they determine, to a great extend, the watermark fragility/robustness against attacks [53].

In the proposed method, the watermark sequence is generated using a pseudorandom number generator and appropriate thresholding. The watermark key used for watermarking a specific frame in the sequence is comprised of the camera id and the frame number. Generating the watermark key using the frame number provides the advantage of producing frame dependent watermarks. In this case, frame removal will cause authentication check failure. Moreover, an attacker cannot use regions of other frames for frame editing, since these regions contain a different watermark and will be detected as non-authentic. The disadvantage of using the frame number in the watermark key is that the detector should also know the frame number in order to perform the authenticity check. However, this problem can be easily solved by using consecutive numbers as frame numbers in the sequence. In this case, the detector needs only the number of the first frame in the sequence which can be always zero.

The block diagram of watermark generation and embedding is depicted in Figure 3. Watermark embedding is performed by altering all the pixels of the original (host) frame according to the following formula:

$$f_w(\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) & \text{if } w(\boldsymbol{x}) = 0 \\ g_1(f(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x})) & \text{if } w(\boldsymbol{x}) = -1 \\ g_2(f(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x})) & \text{if } w(\boldsymbol{x}) = 1 \end{cases} \tag{7}$$

where g_1, g_2 are suitably designed functions based on $\boldsymbol{x}$. $\mathcal{N}(\boldsymbol{x})$ denotes a function that depends on the neighborhood of $\boldsymbol{x}$. The functions g_1, g_2 are called *embedding functions* and are selected so as to define an inverse detection function $\mathcal{D}(f_w(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x}))$. The detection function, when applied to the watermarked image $f_w(\boldsymbol{x})$, produces the watermark $w(\boldsymbol{x})$:

$$\mathcal{D}(f_w(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x})) = w(\boldsymbol{x}). \tag{8}$$

Obviously several embedding functions and the appropriate detection function can be designed giving various watermarking schemes. The function that is used in our method is based on a superposition of real quantities on the pixels which are going to be signed:

$$g_1(f(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x})) = \mathcal{N}(\boldsymbol{x}) \otimes \alpha_1 f(\boldsymbol{x}), \tag{9}$$

$$g_2(f(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x})) = \mathcal{N}(\boldsymbol{x}) \otimes \alpha_2 f(\boldsymbol{x}) \tag{10}$$

where α_1, α_2 are suitably chosen constants and $\mathcal{N}(\boldsymbol{x})$ is a local neighborhood operation of the pixels around $\boldsymbol{x}$. The sign of α_1, α_2 is used in the detection function and their magnitude determines the watermark embedding power. That is, the magnitude of α_1, α_2 is increased for stronger embedding and better watermark resistance against lossy image compression. The value of these constants is defined by the specifications of the watermarking application scenario. In the case of VS data, their value is bounded by the level of noise that is tolerable by the AVI system.

B. Watermark Detection

In the watermark detection procedure the detector generates first the watermark for each frame to be checked. To do so, the id number of the camera that produced the sequence, as well as the frame number are needed. We suppose that the frames are numbered consecutively (e.g. 0...1000). Thus the detector knows the number of the frame in question. The detection function resulting from (9),(10) is defined by:

$$\mathcal{D}(f_w(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x})) = \begin{cases} -1 & \text{if } f_w(\boldsymbol{x}) - \mathcal{N}(\boldsymbol{x}) > 0 \\ 1 & \text{if } f_w(\boldsymbol{x}) - \mathcal{N}(\boldsymbol{x}) < 0 \end{cases} \tag{11}$$

The detection function (11) is valid if $\alpha_1 > 0$ and $\alpha_2 < 0$. This fact should be accounted for in the design of the embedding functions. By applying the detection function to the watermarked image, a bi-valued detection image $d(\boldsymbol{x})$ is produced:

$$d(\boldsymbol{x}) = \mathcal{D}(f_w(\boldsymbol{x}), \mathcal{N}(\boldsymbol{x})). \tag{12}$$

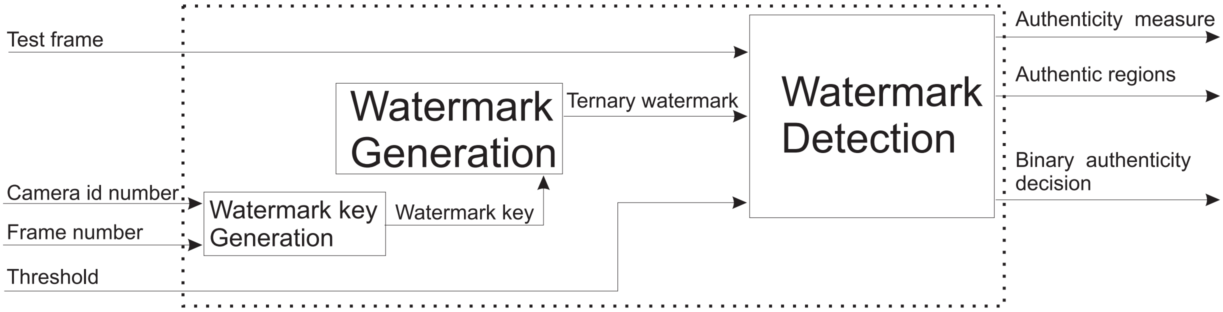


Fig. 4. Watermark detection.

(a) (b)

Fig. 5. (a) Original video frame. (b) Watermarked video frame.

Based on the watermark $w(\boldsymbol{x})$ and the detection image $d(\boldsymbol{x})$, we can decide on image authenticity. We can also detect changes made at certain image regions. The detection is based on the pixel-to-pixel comparison for the nonzero pixels in $w(\boldsymbol{x})$. By comparing the watermark $w(\boldsymbol{x})$ and the detection image $d(\boldsymbol{x})$ we form the *false detection image* given by:

$$e_w(\boldsymbol{x}) = \begin{cases} 1 & \text{if} \;\; w(\boldsymbol{x}) \neq 0 \;\; \text{and} \;\; w(\boldsymbol{x}) \neq d(\boldsymbol{x}) \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

The false detection image has value 1 (white pixels) if a watermarked pixel is falsely detected and 0 otherwise. The watermark detection ratio is given by the ratio of the correctly detected pixels to the sum of the watermarked pixels in the image:

$$D_w = 1 - \frac{\mathrm{card}\{e_w(\boldsymbol{x})\}}{\mathrm{card}\{w(\boldsymbol{x})\}}, \tag{14}$$

where card$\{\cdot\}$ denotes the cardinality of the argument set.

The embedding functions are designed in such way that the probability of a pixel to be detected as signed with g_1 or g_2 is $p \approx 0.5$ for an unwatermarked image. Thus, the detection ratio for an unwatermarked image forms a binomial distribution. The cumulative distribution function (*cdf*) of the watermark detection ratio for an unwatermarked image is given by:

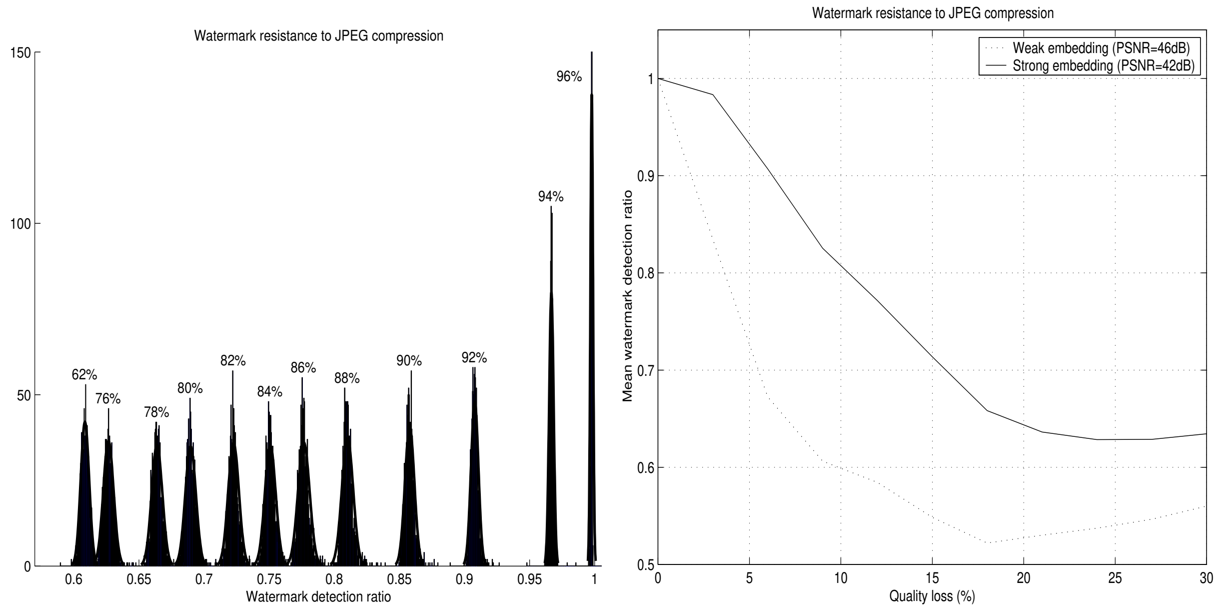
$$P_n = p^k \sum_{i=0}^{n} \frac{k!}{i!(k-i)!}, \tag{15}$$

where k is the total number of the watermarked pixels and n is the number of correctly detected watermarked pixels.

A first level decision on image authenticity is taken by comparing the watermark detection ratio of the test image with a prespecified threshold T. The threshold determines the fragility of the authentication method. If the threshold is set equal to 1, any alteration of the watermarked image that causes a drop in the watermark detection ratio is considered as destroying image authenticity. If the threshold on the watermark detection ratio is lower than 1, then the watermarking method is considered semi-fragile since alterations of the original data that do not reduce significantly the watermark detection ratio and are not concentrated to certain regions are not considered to destroy image authenticity. Such alterations are caused by lossy image compression. If the first-level decision test indicates that the image is somehow altered but authentic, i.e., the watermark detection ratio is lower than 1 but greater than the threshold, a second level decision test should be performed to indicate whether the alterations made on the image are concentrated in certain regions (malicious tampering) or are spread on the image (innocuous alterations). The issue of automatic discrimination between the attacks that cause alterations in the entire image (considered as innocuous when they are weak) and those that affect certain regions, considered as malicious, will be discussed in the following section. The block diagram of the watermark detection module is depicted in Figure 4.

C. Content Tampering Detection

A usual way for tampering VS data is to manipulate certain regions in some frames in order to hide something or to create a new video sequence with different content. The method presented in



(a) (b)

Fig. 6. (a) Probability density functions of the watermark detection ratio after JPEG compression at several compression qualities. (b) Mean value of the watermark detection ratio versus the quality loss caused by JPEG compression.

the previous section succeeds in giving a measure of the image authenticity of each frame and can also provide the user with a binary false detection image $e_w(\boldsymbol{x})$ for each frame. If an image region (e.g., an object or a person) is severely tampered with, then we expect the watermark detection ratio in this region to follow equation (15) and the false detection image to have spatially dense false detection errors (i.e., $e_w(\boldsymbol{x}) = 1$).

However, if the video sequence is compressed with high quality, then the watermark detection ratio for each frame is high and the detection errors (i.e., white pixels in the false detection image) are spread all over the image domain. Automatic discrimination between the regions that are considered maliciously tampered (e.g., replaced objects) and the regions that are considered not severely tampered (e.g., compressed with high quality) is addressed in this section. The objective is the development of an automatic method that creates compact visual descriptions (regions) that correspond to the tampered regions while it clears the isolated falsely detected watermarked pixels that are produced due to high quality lossy compression.

To do so, mathematical morphology operations can be used. Using properly these operations, with certain structuring elements, we can remove impulsive noise and create compact image regions in the false detection image $e(\boldsymbol{x})$. Given a binary image $f(\mathbf{x}) : \mathcal{B} \subseteq \mathcal{Z}^2 \rightarrow \mathcal{U} = \{0, 1\}$ and a structuring element $g(\mathbf{x}) : \mathcal{M} \subseteq \mathcal{Z}^2 \rightarrow \mathcal{U}$, the *binary dilation* of the image $f(\mathbf{x})$ by $g(\mathbf{x})$ is defined as $f \oplus g$. The complementary operation is the *binary erosion* $f \ominus g$ [54,55]. The size of the structuring element used in these operations determines the size of the objects that will be removed from the image or the size of the gaps that will be filled. A sequence of morphological operations that can be applied on the false detection image $e(\boldsymbol{x})$ in order to form the image $t(\boldsymbol{x})$ which contains the non-authentic image regions is the following:

$$t(\boldsymbol{x}) = (e \oplus v_1 \ominus v_2 \oplus v_3)(\boldsymbol{x}). \tag{16}$$

The first dilation using $v_1(\boldsymbol{x})$ aims at creating compact regions that correspond to the tampered regions. The erosion with $v_2(\boldsymbol{x})$ aims at removing the falsely detected watermarked pixels that are produced due to lossy compression and the final dilation using $v_3(\boldsymbol{x})$ restores the original size of the tampered regions. In order to find the size of the structuring elements used in these operations, we consider that the gaps among the falsely detected pixels ($e_w(\boldsymbol{x}) = 1$) that come from a tampered region must be filled completely after the operation $e \oplus v_1$. Thus, the size of the structuring element should be at least the size of the minimum distance between two neighboring white pixels in the false detection image. The mean value of the minimum distance can be computed in a straightforward manner using order statistics [14,56,57]. The calculation of the analytical expressions will not be presented here, since it is beyond the scope of this paper.

Having calculated the mean value s of the minimum distance between two watermarked pixels, we first apply a dilation with the structuring element v_1 of size $s \times s$ in order to fill the regions that are not considered authentic. Afterwards, we want to remove the regions that are produced from isolated falsely detected watermarked pixels. To do so, we apply an erosion by a structuring element of size $(s + 2) \times (s + 2)$, since the size of that objects is now $s \times s$. Finally, a dilation by a structuring element of size 3×3 is applied in order to bring the tampered regions back to their original size.

V. EXPERIMENTAL RESULTS

The proposed authentication algorithm is adequate for both fragile and semi-fragile watermarking. This can be achieved by varying the threshold on the watermark detection ratio. Experiments were conducted to illustrate the performance of the proposed method when lossy image compression, video editing and combined attacks are made to the host VS sequence. A representative color VS sequence comprised of 1000 frames of size 768 × 576 has been used in all our experiments. Each frame was watermarked with a watermark constructed using a pseudo-random number generator with a frame-dependent watermark key. Thus, the correlation of the watermarks between subsequent frames is very small and statistical detection of the watermark using all frames is confronted.

The PSNR of the watermarked sequence frames was approximately 41 dB for all experiments. It is worth noting at this point that, for VS applications, stronger watermark embedding can be used since the invisibility constraint of the watermarking method is not very crucial for this type of applications. The only constraint is the level of noise that is tolerable by the AVI system. Unless otherwise specified, all the results presented refer to the case of frames watermarked with PSNR of 41 dB. A frame of the original VS sequence and the corresponding watermarked frame are depicted in Figure 5. It is evident that the watermark is invisible to a human observer.

(a) (b) (c)

Fig. 7. (a) Initial tampered frame (b) False detection image. (c) Detected tampered regions

The proposed method has the additional advantage of not being computationally intensive. The computational complexity is low, mainly due to the fact that watermark embedding and detection are performed in the spatial domain, thus time-consuming image transforms are needed. Moreover, watermark embedding and detection are not performed in the entire image but only on certain image pixels. The time needed for watermark embedding and detection linearly increases as the number of watermarked pixels is increased. Furthermore, the morphological operations needed can be performed very efficiently by using binary morphology algorithms [55]. Experimental results indicated that an average of 30 fps was achieved both in watermark embedding and watermark detection. That is, watermark embedding and detection are performed in real time for video sequences of size 768×576. The speed of the authentication algorithm is a very critical issue, especially if watermark embedding is performed by the camera.

(a) (b) (c)

Fig. 8. (a) Initial tampered and JPEG compressed frame (b) False detection image. (c) Detected tampered regions

A. Video or image compression

Although lossy compression is not desirable for high quality multimedia products, a VS authentication system must be robust against high quality lossy image compression (e.g., JPEG compression) or video compression (e.g., MPEG compression).

In order to estimate the pdf of the watermark detection ratio after lossy image compression with several compression ratios, each frame of the host VS sequence was watermarked with a watermark generated by a frame dependent key and was compressed at several compression qualities. The VS sequence was watermarked using two different embedding levels. The first was weak watermark embedding resulting in a PSNR of approximately 46dB and the second was strong embedding resulting in a PSNR of approximately 41dB. The estimated pdfs of the watermark detection ratio using strong watermark embedding for several compression qualities are depicted in Figure 6a. Compression with 62% quality corresponds to compression ratio 10:1. The decrease of the mean value of the detection ratio versus the compressed image quality loss is plotted in Figure 6b for the two embedding levels. It can easily be observed that strong embedding generates watermarked sequences that are more robust against lossy compression. Thus, by using stronger watermark embedding, the robustness of the watermark can be increased.

It is also obvious that the watermark detection ratio is reduced smoothly according to the compression ratio and, thus, by setting the appropriate detection threshold, robustness to lossy compression is achieved. The mean watermark detection ratio for lower compression qualities reaches the theoretical value for unwatermarked images i.e., ≈ 0.5. That is, the watermark is totally destroyed and the image is considered non authentic. The rise in performance for lower qualities can be attributed to the different effects of compression on homogeneous regions. Moreover, the watermark is firstly removed from the homogeneous regions while it remains in the regions having rich texture even for lower compression qualities.

B. Video editing

The most frequent type of tampering attack to a video sequence is the replacement/removal of certain frames or the selective editing of certain regions and the tampering of fine details. Frame replacement/removal attack can be easily counteracted by using a frame dependent watermark key. Another solution to this problem is to print the frame number on each frame or to encrypt it in the watermark signal using scrambling techniques. Malicious tampering aims at changing the content of the sequence and not necessarily at reducing the quality of the original sequence. Thus, it is very difficult to detect such alterations, especially when they affect only small details. This kind of alterations succeed to change the image content without reducing significantly the watermark detection ratio. A figure of merit for the performance of the authentication algorithm is the size of the tampered regions that can be detected.

The algorithm described in Section IV-C is used for tampered region detection. A tampered frame of the VS sequence is illustrated in Figure 7a. The original frame is the one shown in Figure 5. Several objects of the original image have been removed and the alterations on the frame are imperceptible if the original is not available. The watermark detection ratio for the tampered frame is 97%. Thus, we conclude that the test frame is altered but we can not decide about image authenticity because the alterations may have been produced by high quality image compression. A second-level test is then needed for detecting authentic image regions. By applying the procedure described in Section IV-C, non-authentic regions are highlighted. The false detection image and the non-authentic regions are shown in Figures 7b and 7c, respectively. It is obvious that the proposed method succeeds in detecting both big object and small detail tampering. However, if the tampered region is very small (e.g. the road line in the original frame) the affected region can not be precisely detected.

C. Combined distortions

When a watermarked sequence is compressed and then is edited somehow, the detection ratio will be decreased because of the lossy compression as well as because of the malicious tampering. The false detection image will contain falsely detected watermarked pixels produced by lossy compression and by image editing. In this case, the objective is to discriminate the regions that contain falsely detected pixels caused by lossy compression and the tampered regions that correspond to image editing.

The method proposed in Section IV-C is used for discriminating between these two types of alterations (lossy compression, image editing). The presented method has the ability of creating visual presentations of the regions where the density of falsely detected watermarked pixels is high, while it clears the isolated falsely detected watermarked pixels. A watermarked image that was compressed with JPEG compression with compression ratio 3:1 and then was edited is depicted in Figure 8a. The false detection image is shown in Figure 8b where the different density in falsely detected pixels on the edited regions of the image can be observed. The detection procedure and the image that results after applying the method proposed in Section IV-C is illustrated in Figure 8c. The tampered regions that are not considered authentic are highlighted by the use of the proposed algorithm while high quality compressed regions are considered to be authentic.

D. Effects on the performance of an AVI system

Finally the effects of the proposed authentication algorithm on the performance of an AVI system have been tested. A motion detection technique similar to the one proposed in [49] has been used in the experiments. An original frame of the sequence, along with two copies watermarked with increasing power (respectively 35 and 30 dB), is shown in Figure 9. The detected moving object obtained by processing the original and the two differently watermarked sequences are correspondingly presented in Figure 10. It appears clearly that the watermarking procedure does not affect the performance of the motion detection procedure when the modification distortion does not exceed the PSNR of 35 dB. Although the watermark is clearly visible to a human at this level, the AVI system can be appropriately tuned in order to filter it out as noise: this is mainly achievable thanks to the temporarily un-correlated nature of the frame dependent watermark. This is no longer valid when the watermark is embedded with a power of 30 dB of PSNR: in this case, it is difficult for the AVI system to distinguish between the modifications caused by the watermark and those caused by moving objects. A fact that should be accounted for, is that the AVI system used for testing the proposed method is limited to low level video sequence processing (simple motion detection): better performance could be obtained by integrating it with higher level object tracking algorithms that are able to eliminate detected moving objects that do not exhibit an homogeneous trajectory (as is the case of the blobs due to the watermark perturbations).

VI. CONCLUDING REMARKS

In this paper the issues related to image authentication for video surveillance applications have been discussed. First of all the need to rely on watermark-based authentication instead of classical cryptography has been motivated. Watermark-based authentication will allow to achieve compression resistance, manipulation localization, and capability to distinguish between innocuous and malicious manipulations (a particularly important feature while dealing with privacy constraint). Subsequently, a survey of the main approaches that can be found in the literature for watermark based authentication has been given, by identifying two classes of algorithms, those based on fragile watermarking and those based on robust watermarking. Furthermore, the main requirements of a watermark-based authentication scheme have been reviewed in the VS framework. In particular, it has been highlighted that the invisibility constraint can be relaxed in this case. A novel (semi-)fragile watermarking algorithm for VS data authentication has also been proposed and its performance has been discussed. The main features of the algorithm can be summarized as: (i) controllable visual deterioration of the VS sequence by varying the watermark embedding power, (ii) robustness to high quality lossy image compression, (iii) automatic discrimination between malicious and innocuous manipulations (lossy image compression) using mathematical morphology operations and (iv) possibility of performing watermark embedding and detection in real time for digital video.

References

[1] C. Regazzoni, G. Fabri, and G. Vernazza, Eds., *Advanced Video-based Surveillance Systems*, Kluwer Academic Publishers, Norwell, MA, USA, 1999.

[2] G. L. Foresti, P. Mähönen, and C.S. Regazzoni, Eds., *Multimedia Video-Based Surveillance Systems. Requirements, Issues and Solutions*, Kluwer Academic Publishers, Norwell, MA, USA, 2000.

[3] "Special issue on video communications, processing and understanding for third generation surveillance systems," *Proc. of IEEE*, This issue.

[4] "Special session on signal processing for multimedia surveillance systems," in *Proc. EUSIPCO 2000, 10th European Signal Processing Conference*, Tampere, Finland, 5-8 September 2000.

[5] B. Schneier, *Applied Cryptography*, John Wiley and Sons, 1994.

[6] D. R. Stinson, *Cryptography, Theory and Practice*, CRC Press, New York, 1995.

[7] M. Barni, F. Bartolini, J. Fridrich, M. Goljan, and A. Piva, "Digital watermarking for the authentication of AVS data," in *Proc. EUSIPCO00, 10th Europ. Signal Processing Conf.*, Tampere, Finland, Sept. 2000.

[8] *95/46/CE Directive of the European Union*, 1995.

[9] G. L. Friedman, "The trustworthy digital camera: restoring credibility to the photographic image," *IEEE Trans. on Consumer Electronics*, vol. 39, pp. 905–910, Nov. 1993.

[10] S. Walton, "Image authentication for a slippery new age," *Dr. Dobb's Journal of Software Tools for Professional Programmers*, vol. 20, Apr. 1995.

[11] R. Wolfgang and E. J. Delp, "A watermark for digital images," in *Proc. ICIP96, IEEE Int. Conf. on on Image Processing*, Lausanne, Switzerland, Sept. 1996, vol. III, pp. 219–222.

[12] M. M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *Proc. ICIP97, IEEE Int. Conf. on Image Processing*, Santa Barbara, California, Oct. 1997, pp. 680–683.

[13] D. Kundur and D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," *Proc. IEEE*, vol. 87, no. 7, pp. 1167–1180, Jul. 1999.

[14] A. Tefas and I. Pitas, "Image authentication and tamper proofing using mathematical morphology," in *Proc. EUSIPCO 2000, Tenth European Signal Processing Conference*, Finland, Sept. 2000.

[15] M. D. Swanson, M. Kobayashi, and A. H. Tewfik, "Multimedia data-embedding and watermarking technologies," *Proc. IEEE*, vol. 86, no. 6, pp. 1064–1087, Jun. 1998.

[16] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding: a survey," *Proc. IEEE*, vol. 87, no. 7, pp. 1062–1078, Jul. 1999.

[17] F. Mintzer, G. W. Braudaway, and M. M. Yeung, "Effective and ineffective digital watermarks," in *Proc. ICIP97, IEEE Int. Conf. on Image Processing*, S.Barbara, CA, Oct. 1997, vol. III, pp. 223–226.

[18] A. Piva, M. Barni, F. Bartolini, and V. Cappellini, "Application-driven requirements for digital watermarking technology," in *Proc. EMMSEC98, Europ. Multimedia Microproc. System and Electronic Commerce Conf. and Exhibition*, Bordeaux, France, Sept. 1998, pp. 513–520.

[19] G. Voyatzis and I. Pitas, "The use of watermarks in the protection of digital multimedia products," *Proc. IEEE*, vol. 87, no. 7, pp. 1197–1207, Jul. 1999.

[20] J. Fridrich, "Methods for tamper detection in digital images," in *GMD Report 85, Proc. Multimedia and Security Work. at ACM Multimedia '99*, Orlando, FL, Oct.-Nov. 1999, pp. 29–33.

[21] W. R. Bender, D. Gruhl, and N. Morimoto, "Techniques for data hiding," in *Storage and Retrieval of Image and Video Databases, Proc. SPIE Vol. 2420*, Feb. 1995, pp. 164–173.

[22] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *NEC Research Institute Technical Report*, , no. 95-10, 1995.

[23] E. Koch, J. Rindfrey, and J. Zhao, "Copyright protection for multimedia data," in *Proc. Int. Conf. on Digital Media and Electronic Publishing*, Leeds, U.K., Dec. 1994.

[24] K. Matsui and K. Tanaka, "Video-steganography," in *Proc. IMA, Intellectual Property Project*, January 1994, vol. I, pp. 187–206.

[25] R. G. van Schyndel, A. Z. Tirkel, and C. F. Osborne, "A digital watermark," in *Proc. ICIP94, IEEE Int. Conf. on Image Processing*, Austin, Texas, Nov. 1994, vol. II, pp. 86–90.

[26] M. Kutter, S. K. Bhattacharjee, and T. Ebrahimi, "Towards second generation watermarking schemes," in *Proc. ICIP 1999, IEEE Int. Conf. on Image Processing*, Kobe, Japan, 24-28 October 1999, vol. I, pp. 320–323.

[27] "Special session on second generation digital watermarking methods," in *Proc. ICIP00, IEEE Int. Conf. on Image Processing*, Vancouver, Canada, Sept. 2000.

[28] S. Craver, N. Memon, B. L. Yeo, and M. M. Yeung, "Resolving rightful ownership with invisible watermarking techniques: limitations, attacks and implications," *IEEE Journ. Sel. Areas Commun.*, vol. 4, no. 16, pp. 573–586, May 1998.

[29] M. Barni, F. Bartolini, G. Bini, V. Cappellini, A. Fringuelli, G. Meucci, and A. Piva, "Enforcement of copyright laws for multimedia through blind, detectable, reversible watermarking," in *Proc. ICMCS99, Int. Conf. Multimedia Comput. and Syst.*, Firenze, Italy, Jun. 1999, vol. II, pp. 199–203.

[30] B. Pfitzmann, *Information hiding terminology*, vol. 1174 Lecture Notes in Computer Science, Springer Verlag, Cambridge, UK, May 1996.

[31] "Special issue on: Identification and protection of multimedia information," *Proc. IEEE*, vol. 87, no. 7, Jul. 1999.

[32] M. Barni, F. Bartolini, and A. Piva, "Digital watermarking of visual data: state of the art and new trends," in *Proc. EUSIPCO00, 10th Europ. Signal Processing Conf.*, Tampere, Finland, Sept. 2000.

[33] N. Nikolaidis and I. Pitas, "Digital image watermarking: an overview," in *Proc. ICMCS99, Int. Conf. on Multimedia Comput. and Syst.*, Florence, Italy, 7-11 June 1999, vol. I, pp. 1–6.

[34] P. W. Wong, "A public key watermark for image verification and authentication," in *Proc. ICIP98, IEEE Int. Conf. on Image Processing*, Chicago, Illinois, Oct. 1998, vol. I, pp. 455–459.

[35] J. Fridrich, M. Goljan, and N. Menom, "Further attacks on Yeung-Mintzer fragile watermarking scheme," in *Security and Watermarking of Multimedia Contents, Proc. SPIE Vol. 3971*, S.Jose, CA, USA, Jan. 2000, pp. 428–437.

[36] E. T. Lin, C. I Podilchuck, and E. J. Delp, "Detection of image alterations using semi-fragile watermarks," in *Security and Watermarking of Multimedia Contents, Proc. SPIE Vol. 3971*, S.Jose, CA, USA, Jan. 2000, pp. 152–163.

[37] S. D. Servetto, C. I. Podilchuk, and K. Ramchandran, "Capacity issues in digital image watermarking," in *Proc. ICIP98, IEEE Int. Conf. on Image Processing*, Chicago, IL, Oct. 1998, vol. I, pp. 445–449.

[38] M. Barni, F. Bartolini, A. De Rosa, and A. Piva, "Capacity of full frame DCT image watermarks," *IEEE Trans. Image Processing*, vol. 9, no. 8, pp. 1450–1455, August 2000.

[39] P. Moulin, "The role of information theory in watermarking and its application to image watermarking," *Signal Processing, Special Issue on Information theoretic Issues in Digital Watermarking*, vol. 81, Jun. 2001,

[40] C. Y. Lin and S. F. Chang, "Issues and solutions for authenticating MPEG video," in *Security and Watermarking of Multimedia Contents, Proc. SPIE Vol. 3657*, S.Jose, CA, Jan. 1999, pp. 54–65.

[41] M. P. Queluz, "Content based integrity protection of digital images," in *Security and Watermarking of Multimedia Contents, Proc. SPIE Vol. 3657*, S.Jose, CA, Jan. 1999, pp. 85–93.

[42] S. Bhattacharjee and M. Kutter, "Compression tolerant image authentication," in *Proc. ICIP98, IEEE Int. Conf. on Image Processing*, Chicago, Illinois, Oct. 1998, vol. I, pp. 435–439.

[43] M. Wu and B. Liu, "Watermarking for image authentication," in *Proc. ICIP98, IEEE Int. Conf. on Image Processing*, Chicago, Illinois, Oct. 1998, pp. 437–441.

[44] L. Xie and G. R. Arce, "A blind wavelet based digital signature for image authentication," in *Proc. EUSIPCO98, 9th Europ. Signal Processing Conf.*, Rodos, Greece, Sept. 1998, pp. 21–24.

[45] J. Fridrich, "Image watermarking for tamper detection," in *Proc. ICIP98, IEEE Int. Conf. on Image Processing*, Chicago, Illinois, Oct. 1998, vol. II, pp. 404–408.

[46] J. Fridrich and M. Goljan, "Images with self-correcting capabilities," in *Proc. ICIP99, IEEE Int. Conf. on Image Processing*, Kobe, Japan, Oct. 1999, vol. III, pp. 792–796.

[47] L. M. Marbel, G. W. Hartwig, and C. Boncelet, "Compression-compatible fragile and semi-fragile tamper detection," in *Security and Watermarking of Multimedia Contents, Proc. SPIE Vol. 3971*, S.Jose, CA, USA, Jan. 2000, pp. 131–139.

[48] B. S. Manjunath J. J. Chae, D. Mukherjee, "Color image embedding using multidimensional lattices structure," in *Proc. ICIP98, IEEE Int. Conf. on Image Processing*, Chicago, IL, Oct. 1998.

[49] P. Bouthemy and P. Lalande, "Recovery of moving object masks in an image sequence using local spatiotemporal contextual information," *Optical Engineering*, vol. 32, no. 6, pp. 1205–1212, June 1993.

[50] J. R. Hernandez, F. Perez-Gonzales, J. M. Rodriguez, and G. Nieto, "Performance analysis of a 2-D-multipulse amplitude modulation scheme for data hiding and watermarking of still images," *IEEE Journ. Sel. Areas Commun.*, vol. 4, no. 16, pp. 510–524, May 1998.

[51] N. Nikolaidis, S. Tsekeridou, A. Nikolaidis, A. Tefas, V. Solachidis, and I. Pitas, "Applications of chaotic signal processing techniques to multimedia watermarking," in *Proceedings of the IEEE workshop on Nonlinear Dynamics in Electronic Systems*, Catania Italy, May 18-20 2000, pp. 1–7.

[52] A. Tefas, A. Nikolaidis, N. Nikolaidis, V. Solachidis, S. Tsekeridou, and I. Pitas, "Statistical analysis of markov chaotic sequences for watermarking applications," in *Proc. ISCAS 2001, IEEE Int. Symp. on Circuits and Syst.*, Sydney, Australia, 2001.

[53] J. K. Su and B. Girod, "Power spectrum condition for energy-efficient watermarking," in *Proc. ICIP99, IEEE Int. Conf. on Image Processing*, Kobe, Japan, Oct. 1999, vol. 1, pp. 301–305.

[54] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, London, 1982.

[55] I. Pitas and A.N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*, Kluwer Academic Publ., Boston, MA, 1990.

[56] A. Tefas and I. Pitas, "Image authentication and tamper proofing using chaotic mixing and mathematical morphology," *IEEE Trans. on Image Processing*, submitted December 1999.

[57] H.A. David, *Order Statistics*, John Wiley & Sons, New York, 1981.

Fig. 9. An original frame of the sequence is shown, along with two copies of it, watermarked with increasing power (respectivelly 35 and 30 dB). The watermark is well visible in both the heavily watermarked images.

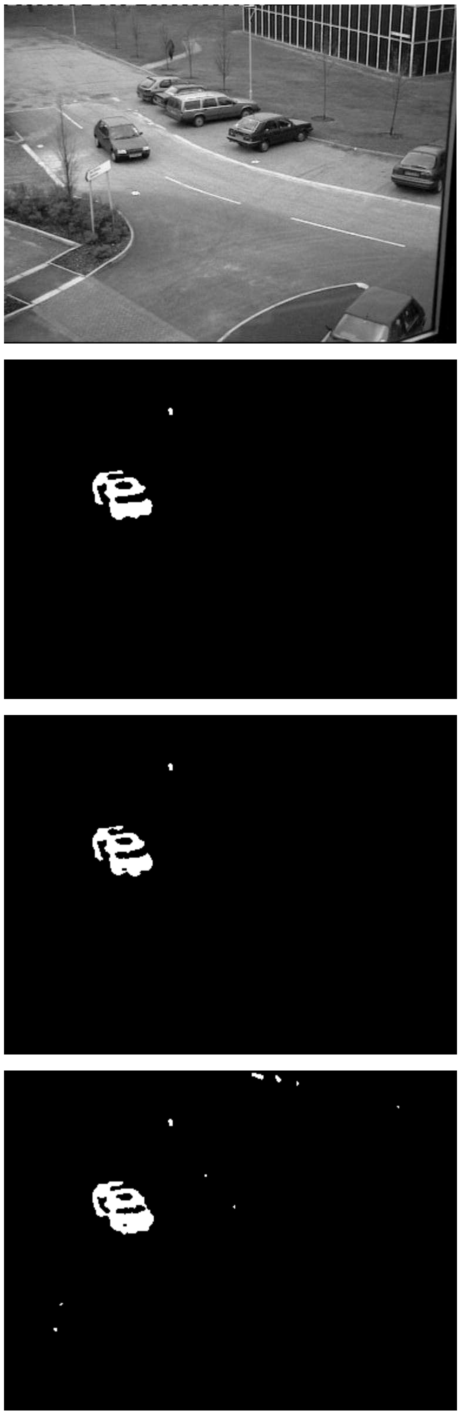


Fig. 10. The original frame (upper upper) and the masks of the detected moving object obtained by processing the original (upper middle) and the two differently watermarked sequences whose frames are shown in Figure 9 are presented. It is evident that, while yet visible to the humane eye the watermark embedded with 35 dB of PSNR does not affect the results of the AVI system (middle bottom), while it does when the embedding level reaches 30 dB (bottom bottom).