Applying Web Oriented Technologies to Implement an Adaptive Spread Spectrum Watermarking Procedure and a Flexible DRM Platform

Franco Frattolillo **Salvatore D'Onofrio**

Research Centre on Software Technology,
Department of Engineering,
University of Sannio,
Benevento, Italy,
Email: {frattolillo,donofrio}@unisannio.it

Abstract

The advances in multimedia technologies have created opportunities for Internet pirates, who can copy multimedia documents and illegally distribute them, thus violating the legal rights of document owners or web content providers. Such a situation is an actual threat particularly for web content providers, which often have neither the technical competence nor the economical advantage to directly implement effective security services to combat the unauthorized trading of their distributed multimedia documents. This paper shows how known and widespread web oriented technologies can be exploited to develop a simplified but flexible digital rights management web platform. In particular, XML based technologies are used to implement an "on buyer", adaptive watermarking procedure, while web services are used to implement a web platform by which web service providers can supply copyright protection services on behalf of web content providers in a secure network context. Thus, content providers that exploit the proposed platform can take advantage of a copyright protection system acting as a trusted third party without having to directly implement it. On the other hand, service providers can follow the proposed XML based approach to implement fully in the W3C XSLT language complex watermarking procedures without having to interface native codes with web services frameworks commonly used to develop web platforms.

Keywords: digital rights management, watermarking, web services, XML.

1 Introduction and motivations

The use of digital watermarking has gained popularity as a main technology to set up reliable digital rights management (DRM) systems for the copyright protection of digital contents distributed in the Internet. However, watermarking procedures can give a support to DRM systems only if they can provide a good degree of robustness against the most common, nonmalevolent manipulations, and prove to be secure against intentional attacks. Moreover, even if watermark robustness and security are crucial requirements, a closer look at the most common business models used by web content providers (CPs) to distribute multimedia contents in the Internet reveals that many other important requirements have to be matched in order to device a correct interaction protocol among content sellers and buyers (Barni & Bartolini 2004).

In the most classical scenario served by watermarking, the perceptual invisible digital signature inserted in a content commonly allows to potentially establish if a user is illegally in possession of it, but it does not enable to establish the "source" of the content, that is, who has initially bought and then illegally shared it via, for example, peer-to-peer network applications. Therefore, to discourage unauthorized content duplication and distribution, watermarking should make it possible to implement a mechanism to trace unauthorized copies to the original owners of the contents, so as to track the authors of the infringements. This is made possible by letting the seller insert a distinct watermark, which in this case is called a "fingerprint", identifying the buyer within any copy of content that is distributed (Cox, Bloom & Miller 2001). However, even if "readable" watermarking schemes based on "blind" and not publicly available decoders are adopted (Barni & Bartolini 2004), the dangerous "average" and "collusion" attacks are still to be considered actual threats (Barni, Bartolini & Furon 2003). Therefore, possible countermeasures against such attacks consist both in letting the watermark depend on the host signal and of adopting "anticollusion" codes, in which case the watermarked information and the embedding strategy are chosen in such a way that averaging different watermark signals, each identifying a different colluding user, leaves certain parts of the watermark unaffected, thus permitting the recovery of some information about the colluding user pool (Trappe, Wu *et al.* 2003).

A fingerprinting application induces an "on buyer" behavior, that is, watermark requires to be embedded whenever a user requests a content. In particular, when user requests are sent to CPs, the watermarks have to be embedded into the required contents on the fly, that is, during the web transactions. This requires a fast implementation of the watermarking procedures that does not compromise security and robustness. Furthermore, since a robust and secure watermarking procedure can be computationally intensive or increase the size of the protected contents, it is important to adapt it to the specific characteristics of both the terminal used to open the required content and the transaction carried out between the user and the CP. For example, a PDA or a mobile phone or a terminal with no storing capacity or limited visualization capacities could receive low-quality, "lightly watermarked" videos during transactions taking place on low performance networks. Therefore, an advanced, web oriented watermarking

procedure should:

- take into account the buyer identities;
- be fast without compromising security and robustness;
- implement a readable scheme based on a blind and not publicly available decoder;
- depend on the host signal and adopt "anticollusion" codes;
- exhibit an "adaptive" behavior, that is, it should take into account the characteristics of both the terminals used to open the contents and the network transactions carried out to deliver the protected contents.

However, a watermarking procedure matching the above requirements does not take into account the buyers rights, since the watermark is autonomously inserted by the CP, that is the seller, without any control. Thus, a buyer whose watermark is found in an unauthorized copy cannot be legally prosecuted, since he/she can claim that the unauthorized copy was created and distributed by the CP. To this end, a possible solution consists in resorting to a certificated trusted third party (TTP), which takes charge of both inserting watermarks within the contents to be protected and distributed in the Internet and retrieving them in case a dispute resolution protocol has to be run.

The idea of resorting to a certificated TTP is nowadays considered by CPs a clever way to avoid the above problems. In fact, CPs often have neither the technical competence nor the economical advantage to directly apply complex or not certificated watermarking procedures to the distributed contents. They appear to be more involved in improving their specific and consolidated web consumer- or business-focused applications, rather than implementing new services based on advanced technologies that are not part of their original core business. In practice, CPs generate revenues by essentially focusing on creating and running high end web applications by which to provide multimedia contents, not on implementing the backend software infrastructures for supporting them. On the other hand, service providers (SPs) are the new web entities which have knowledge and expertise in the use of web programming technologies, and their core business is just to supply complex and specialized software services to CPs. This model has already proven highly successful in the Internet, where SPs enable the building of web applications for CPs with good ideas but little time for technology. Therefore, SPs can be considered particularly suited to deploy certificated watermarking services on behalf of CPs in a secure network context, thus acting as TTPs, while CPs remain the only valid candidates to provide high end web applications to better satisfy new and complex needs coming from end users. However, this is possible only if the integration among the services implemented by CPs and those ones provided by SPs can be assured without requiring a tight coupling among the involved different web entities. To this end, XML based and web services technologies can be used to simplify this integration process, that is the process of dynamically integrating elemental system and business services into more complex customer services (Brunner, Cohen *et al.* 2001).

This paper shows how known and widespread web oriented technologies can be exploited to develop a simplified but flexible DRM web platform. In particular, XML based technologies are used to implement an "on buyer", adaptive watermarking procedure for MPEG videos matching the above requirements and based on a spread-spectrum additive embedding technique, while web services are used to implement a web platform by which SPs can supply copyright protection services on behalf of CPs in a secure network context. Thus, CPs that exploit the proposed platform can take advantage of a copyright protection system acting as a TTP without having to directly implement it. On the other hand, SPs can follow the proposed XML based approach to implement fully in Java complex watermarking procedures without having to interface native codes with web services frameworks commonly used to develop web platforms.

Although many techniques have been proposed in literature for embedding fingerprints in MPEG videos, spread-spectrum additive embedding techniques (SS) have proven robust and secure against a number of signal processing operations and attacks (Cox, Kilian *et al.* 1997, Lin, Wu, Lui *et al.* 2001, Lubin, Bloom & Cheng 2003). Moreover, with appropriately chosen parameters and adopting specific and documented improvements (Malvar & Florêncio 2003), the spread-spectrum watermark can survive moderate geometric distortions without suffering from the sensitivity to amplitude scaling evidenced by quantization index modulation (QIM) (Chen & Wornell 2001) and roughly achieving the same noise robustness gain as QIM. Furthermore, since an SS embedding technique depends on a few parameters, it can be well exploited to implement the adaptive behavior of the proposed watermarking procedure.

As reported above, the implemented web platform should provide a security context within which CPs can exploit the copyright protection services exported by SPs. In particular, the main goals in devising the platform have been: (1) to make the platform simple, modular and extensible; (2) to create a secure network context for all the transactions that take place among end users and the service entities composing the platform; (3) to show that CPs, adhering to the service model proposed for the platform, can take advantage of advanced services without having to change their original role and main web applications.

The paper is organized as the follows. Section 2 describes the implemented watermarking procedure. Section 3 describes the procedure implementation based on XML technologies. Section 4 presents the web platform implementing the security context within which CPs can exploit the copyright protection services exported by SPs. Section 5 reports some experimental results. Section 6 discusses related work. Finally, Section 7 reports conclusion remarks.

2 The watermarking procedure

The implemented watermarking procedure is based on the approach proposed in (Malvar & Florêncio 2003) and is specialized for MPEG-2 compressed video streams. In particular, the embedding approach is an improvement of the original SS techniques and is based on the main idea of removing the host signal as source of interference, thus producing a dramatic improvement in the quality of the watermarking process.

2.1 The basic scheme

In the basic scheme (see Figure 1), the insertion of the watermark in the compressed video V is accomplished by extracting the encoded 8×8 blocks of the video and processing them together with the corresponding blocks of the watermarking signal W. In particular, the MPEG-2 bitstream is split into its main components, and only the DCT encoded signal blocks are

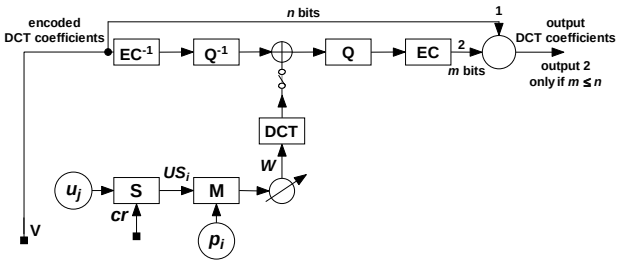

Figure 1: The basic watermarking scheme.

modified. Each encoded block is represented by a sequence of Huffman codes, each representing one (run-level)-pair and, thus, one quantized non-zero DCT coefficient of the current signal block. Therefore, to insert the watermark, each Huffman code is decoded (EC^{-1}) and inversely quantized (Q^{-1}), that is, the mapping from the quantizer index to the quantizer representative is performed. After this processing, a quantized DCT is added to the corresponding DCT coefficient from the transformed W signal, yielding a watermarked DCT coefficient. This is then quantized (Q) and Huffman encoded (EC). Moreover, the watermarking basic scheme is designed not to increase the output bit-rate. Therefore, in Figure 1, the output 2 is selected only if the number of bits used to represent the codeword for the protected video signal is less or equal than the number of bits used to represent the same codeword in the original video signal (Hartung & Girod 1997).

Finally, the watermarking procedure is also completed by a scheme for drift compensation, which is not shown in Figure 1 for the sake of brevity.

2.2 The "on buyer" behavior

The proposed procedure adds a noise-like signal to the encoded video signal processed block by block. The watermark signal (see Figure 1) is generated from a sequence of bits $u_j \in \{-1, 1\}$, which is used to identify a user and is spread (S) by a large factor cr, called *chip-rate*, thus obtaining the spread sequence $us_i = u_j,\ j \cdot cr \leq i < (j+1) \cdot cr$. Then, the noise-like signal is generated by modulating (M) the spread sequence with a binary pseudo-noise sequence $p_i \in \{-1, 1\}$, which, in the proposed solution, has to be unambiguously associated to the protected video. This way, once a protected video has been selected, it is possible to employ the pseudo-noise sequence p_i associated to it to extract the watermark and thus obtain the user sequence $u_j \in \{-1, 1\}$. To this end, the signal of the protected video can be correlated with the p_i sequence over a cr wide correlation window, and the extracted watermark can be then analyzed to obtain the sequence of bits identifying the user who bought the video.

Finally, it is worth noting that the sequences u_j are assigned to identify users according to an anticollusion technique (Trappe, Wu *et al.* 2003) and exploit an error correction code. However, this issue is not elaborated here because this is not a main goal of the paper and for the sake of brevity.

2.3 The improved scheme

The scheme described in Sections 2.1 and 2.2 is based on the simple formula

$$\mathbf{s} = \mathbf{x} + u\mathbf{m} \tag{1}$$

where the vector $\mathbf{x}$ is the host signal, $\mathbf{m}$ is the chip sequence built from p_i, u represents a bit from the u_j sequence, and the vector $\mathbf{s}$ is the watermarked signal. In particular, (1) assumes that one bit of information from the u_j sequence is embedded in the vector $\mathbf{s}$ of cr values according to the common SS techniques. However, the actually implemented watermarking scheme is based on a slight modification to the SS approach, defined in (Malvar & Florêncio 2003) as the linear version of the improved SS technique (ISS). In fact, this variant assumes that the amplitude of the inserted chip sequence can vary by a linear function

$$\mathbf{s} = \mathbf{x} + (\alpha u - \lambda x)\mathbf{m} \tag{2}$$

where $x \triangleq \langle \mathbf{x}, \mathbf{m} \rangle / \langle \mathbf{m}, \mathbf{m} \rangle$ and $\langle \mathbf{x}, \mathbf{m} \rangle$ is the inner product defined as

$$\langle \mathbf{x}, \mathbf{m} \rangle \triangleq \frac{1}{cr} \sum_{i=0}^{cr-1} x_i m_i \tag{3}$$

In particular, (3) also defines the norm whenever it is used, for example, as $\langle \mathbf{x}, \mathbf{x} \rangle$.

The parameters α and λ control the distortion level and the removal of the carrier distortion on the detection statistic. In fact, if $\mathbf{y}$ is the available distorted version of $\mathbf{s}$ obtained by adding to $\mathbf{s}$ a noise $\mathbf{n}$ modeled as an uncorrelated white Gaussian random process, the sufficient statistic available at the watermark extractor r is

$$r = \frac{\langle \mathbf{y}, \mathbf{m} \rangle}{\langle \mathbf{m}, \mathbf{m} \rangle} = \alpha u + (1 - \lambda x) + n \tag{4}$$

where $n \triangleq \langle \mathbf{n}, \mathbf{m} \rangle / \langle \mathbf{m}, \mathbf{m} \rangle$. Therefore, by using the encoder knowledge about the signal, the performance of the watermarking system can be enhanced by modulating the energy of the inserted watermark to compensate for the host signal interference. In particular, the closer λ is made to 1, the more the influence of x is removed from r. The detector is the same as in the SS watermarking techniques, i.e., the detected bit is $sign(r)$. Furthermore, traditional SS techniques can be obtained by setting $\alpha = 1$ and $\lambda = 0$.

The results reported in (Malvar & Florêncio 2003) enable the optimal values of α and λ to be calculated for the watermarking system defined by (2) and under the assumptions made in Sections 2.1 and 2.2. In particular, low values for the error probability (i.e. lower than 10^{-5}) can be achieved by setting

$$\alpha = \sqrt{\frac{cr - \lambda^2 \sigma_x^2}{cr}} \tag{5}$$

and λ close to 1 (i.e. in the range 0.9, 1) under the assumption that cr is large enough and SNR is higher than 10 dB.

2.4 The adaptive behavior

As reported in Section 1, watermarking should have an adaptive behavior. To this end, in the proposed procedure the watermark embedded in a required video depends on the characteristics of both the terminal used to open the video and the quality of the network connection between the user and the CP. This dependence is controlled by two specific functions, Φ and Ψ (see Figure 2), which determine respectively the chip-rate cr and the video output bit-rate. These functions depend on two variables, τ and η, which qualify respectively the user terminal type and network connection.

In a preliminary simplified model, τ essentially captures the terminal visualization capacities, i.e. the video resolution, while η synthesizes the bandwidth and latency of the user network connections. In fact,

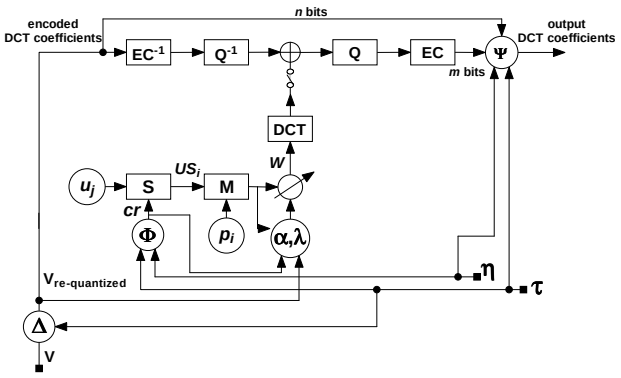

Figure 2: The improved watermarking scheme.

τ can be derived from what declared by users when they interact with the CP web server, while η can be also directly estimated by a CP during the transaction web phase with users. In particular, network connections are roughly differentiated in three main categories: modem and GPRS links, DSL and LAN connections, and T1, T3 lines.

Preliminary tests have shown that cr can usefully vary in the range from $cr_{min}=10,000$ to $cr_{max}=1,000,000$. Therefore, by setting $\tau_{min}=320\times240$ and $\tau_{max}=1024\times768$, Φ is given by

$$\Phi=\begin{cases} cr_{min} & \text{if } \tau<\tau_{min} \\ cr_{max} & \text{if } \tau>\tau_{max} \\ \eta\left(\frac{(cr_{max}-cr_{min})(\tau-\tau_{min})}{\tau_{max}-\tau_{min}}+Kcr_{min}\right) & \text{otherwise} \end{cases} \quad (6)$$

In (6) η and K may respectively assume only three different values, each corresponding to a different kind of user network connection. Table 1 reports the possible values for η and K. In fact, the product $\eta\cdot K$ represents a relative weight that characterizes the network connections.

	η	K
modem or GPRS links	0,5	2
DSL, LAN connections	0,7	10
T1, T3 lines	0,5	100

Table 1: The possible values of η and K in (6).

As reported above and in Figure 2, Ψ controls the video output bit-rate. It specifies the maximum increment percentage that the output bit-rate can induce in the video size. The first conducted tests have shown that such increment can usefully vary in the range from $in_{min}=5\%$ to $in_{max}=50\%$ without compromising the actual video quality. Therefore, Ψ is given by

$$\Psi=\begin{cases} in_{min} & \text{if } \tau<\tau_{min} \\ in_{max} & \text{if } \tau>\tau_{max} \\ \eta\left(\frac{(in_{max}-in_{min})(\tau-\tau_{min})}{\tau_{max}-\tau_{min}}+Kin_{min}\right) & \text{otherwise} \end{cases} \quad (7)$$

As in (6), also in (7) η and K may respectively assume only three different values, each corresponding to a different kind of user network connection. Table 2 reports these values. Moreover, the product $\eta\cdot K$ can be still considered a relative weight characterizing the network connections. However, the weights in Table 2 are less than the corresponding ones reported in Table 1, and this because the possible chip-rate range is larger than the range specifying the increment percentage of the video size.

The behaviors of Φ and Ψ have been determined taking into account that a high value for cr increases

	η	K
modem or GPRS links	0,2	5
DSL, LAN connections	0,4	8
T1, T3 lines	0,3	24,3

Table 2: The possible values of η and K in (7).

the watermark robustness, but at the same time decreases the data rate for watermark. On the other hand, controlling the bit-rate means to determine the fraction of the watermark signal that can be successfully embedded in the protected video: increasing the bit-rate means to increase this fraction and thus improve the robustness of the watermarking, even though the video quality could suffer a degradation.

To this end, it is worth noting that in many known procedures (Hartung & Girod 1997) the watermarking is generally assumed not to increase the output bit-rate. On the contrary, in the proposed procedure, Ψ may increase the bit-rate, since the video size is assumed to change according to both the required protection level and the actual service conditions: the former is essentially identified by τ, while the latter are captured by η. Therefore, once the increment for a video size has been determined, Ψ sets a counter to the increment value. Then, Ψ updates the counter by subtracting from it the difference between the number of bits needed to represent a codeword for the watermarked signal sent to output and the number of bits used to represent the same codeword for the original video signal: positive differences are considered "debts", while negative differences are considered "credits". Thus, when the counter reaches 0, further codewords for the watermarked signal are sent to output only if further credits occur that balance debts. This way, the procedure ensures that the increment of the video size remains constant.

In the proposed procedure, cr may vary to implement the adaptive behavior. As a consequence, to extract the watermark from a video, it is necessary to have the associated sequence p_i as well as the value of cr used to watermark the video. To this end, watermarking is actually performed in two phases. In the former, a cr_v value constantly associated to the video is used to embed the first n values of the sequence u_i. These values are used to identify the chip-rate cr calculated by Φ and that has to be used to watermark, in the latter phase, the remaining part of the video. Thus, given the video, the sequence p_i and the value cr_v can be identified and then applied to retrieve the first n values of the sequence u_j, which identify the cr value to be used to extract the watermark from the remaining part of the video.

The adaptive behavior of the proposed procedure is further improved by assuming that the distributed videos can be characterized by a different quality depending on the visualization capacities of the user terminal. This feature is implemented by stating that the original video quality directly depends on τ. To this end, it is worth noting that the adaptive, on buyer behavior requires a content manipulation to be performed "on the fly", when the web transaction takes place, in order to adapt the video quality and the applied protection to the transaction characteristics. In particular, in order not to reduce the robustness and security level achieved by the watermarking procedure, watermark has to be embedded after the video quality adaptation, and this means that different versions of the available videos should be handled at the CP side. In fact, two main solutions can be adopted by CPs: the former requires that different versions of each video made available by a CP are generated, stored and handled at server side, while the latter

is based on the dynamic generation of such versions from high quality master videos. However, holding one version of a video for each possible quality level is a very heavy solution at server side, especially when the CP server has to address low to high resolution video terminals. On the contrary, the latter solution appears to be more flexible and memory saving, provided that an efficient implementation of the adaptation procedure is used.

Quality adaptation of MPEG-2 videos can be carried out by exploiting one of the two main and well-known techniques: the re-quantization of the DCT coefficients and the cut of the high frequencies, i.e. the AC coefficients (Lei & Georganas 2002). The former is based on the increment of the quantization step in order to pull down ulterior DCT coefficients, while the latter is simply based on eliminating ulterior terms of every DCT 8×8 blocks by cutting the terms relative to the high frequencies. Therefore, both techniques reduce the dimensions of the bit-stream as well as the quality of the video, even if it is demonstrated that the former technique turns out to be more efficient of the latter in that it produces a smaller quantization error.

MPEG-2 video re-quantization is therefore controlled by the further function Δ that determines the increment of the re-quantization step. Preliminary tests have shown that such increment can usefully vary in the range from 0 to $ir_{max}=30\%$. Therefore, Δ is given by

$$\Delta=\begin{cases} ir_{max} & \text{if } \tau \leq 320\times240 \\ 0 & \text{if } \tau > 640\times480 \\ ir_{max}\left(1-\left(\frac{\tau-(320\times240)}{(640\times480)-(320\times240)}\right)^2\right) & \text{otherwise} \end{cases} \quad (8)$$

Obviously, a re-quantization step equals to 0 means that the original master video quality is not modified. Furthermore, the not linear behavior of (8) allows for mostly reducing the quality of the low resolution videos, i.e., the videos that are lightly watermarked.

Finally, it is worth noting that re-quantization is strategic to implement the adaptive behavior of the proposed procedure. In fact, whenever a malicious user tries to obtain a lightly watermarked video by deceptively claiming to be provided with a low resolution video terminal and to be connected by means of a low performance link, he/she ends up obtaining only a re-quantized, low quality video which, even if unprotected, can be neither advantageously played by a high resolution video terminal nor considered interesting to Internet pirates.

3 The implementation of the watermarking procedure

The procedure described in Section 2 has been implemented by mainly exploiting XML based techniques of document structure transformation. In particular, the procedure assumes that the high quality master videos to be protected are all made initially available in a variant of the Bitstream Syntax Description Language (BSDL) (Amielh & Devillers 2002), which allows to describe the MPEG-2 videos by using the XML (see Figure 3). In fact, the elaborated variant enables to both describe the whole bitstream and add a further layer, similar to metadata, able to address the bitstream high-level structure, i.e. how the bitstream is organized in layers or packets of data.

Since CPs store all the master videos, they have to transform the original format of the videos contained in their web repositories if they want to interact with the SPs that implement the proposed watermarking procedure. However, such a transformation appears to be an easy task, since it can be carried out by simply running a specific program directly downloadable from SPs.

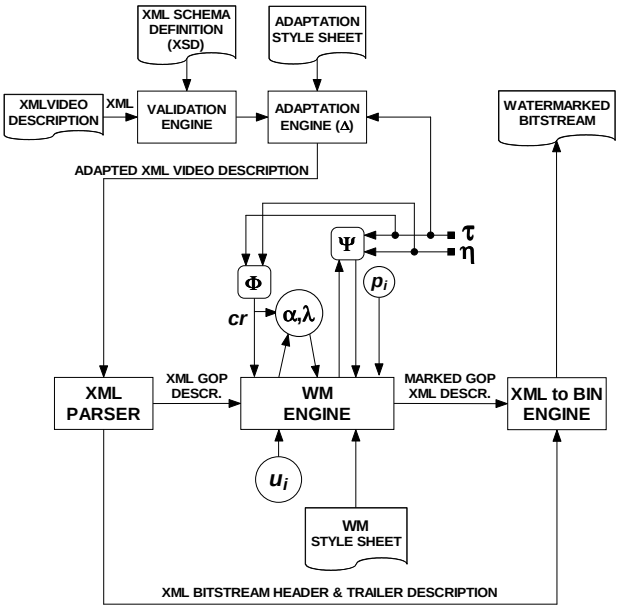

Figure 3: The XML based implementation of the watermarking procedure.

Once an MPEG-2 bitstream is described in XML, all the manipulations to be performed on the video can be carried out as normal XML-to-XML editing operations. In particular, the W3C language XSLT is an efficient way to specify transformations on XML documents by means of style sheets. In fact, an XSLT style sheet contains one or several templates defining the modifications to be applied to the elements or to the attributes matching a set of conditions. Therefore, once received the BSDL description of a video (in the following referred to as BSDLd), an SP can validate it by employing a specific XSD file. Then, the SP can calculate the Δ function and carry out the video re-quantization. This operation, as observed above, is an editing operation on the BSDLd performed by an XSLT transformation engine that applies a parametric adaptation style sheet. Then, the watermarking procedure computes the Φ and Ψ functions and updates the parametric style sheets that control the watermark generation. In particular, the BSDLd is parsed so as to identify only tags whose associated information has to be manipulated. To this end, a SAX parser is used, since it is characterized by an event-based behavior that, differently from a DOM parser, allows for memory saving. On the contrary, the parts of the BSDLd that have not to be manipulated can avoid the watermark XSLT engine (see Figure 3).

Finally, after having generated the new XML description of the watermarked video in the variant of the BDSL, the SP can perform the last operation, that is the XML-to-bitstream conversion, thus generating the protected version of the video.

4 The DRM web platform

Outsourcing a watermarking service involving distinct entities in a web context without compromising security is a complex task. In fact, a correct interoperability among CPs and an SP supplying a watermarking service and acting as a TTP is generally assured only by forcing a tight coupling among these web entities.

To overcome this drawback, web services technologies can be exploited. They have proven to be successful in supporting the creation of new and complex distributed web applications by integrating existing software components provided by distinct web entities (Brunner, Cohen *et al.* 2001). However, when services have to be provided within a secure network context, web service technologies are not sufficient, and it becomes strategic to design a web platform able to sketch an environment of well defined interactions among distinct web entities. This way, it is possible to simplify application integration and promote the reuse of the interaction scheme over multiple web entities and applications.

4.1 The platform architecture

The proposed platform, whose architecture is sketched in Figure 4, consists of two main parts: the former includes the web servers of CPs and represents the "front-end" tier of the platform; the latter, which represents the "back-end" tier of the platform, is composed of the web services implemented by SPs. In particular, in the proposed architecture an SP does not directly expose the web services that it supplies, but hides them behind a "dispatcher", which acts as a unique interface towards CPs for all the web services implemented. Such an interface, designed itself as a web service, takes charge of receiving the service requests from CPs and dispatches them to the actual web services.

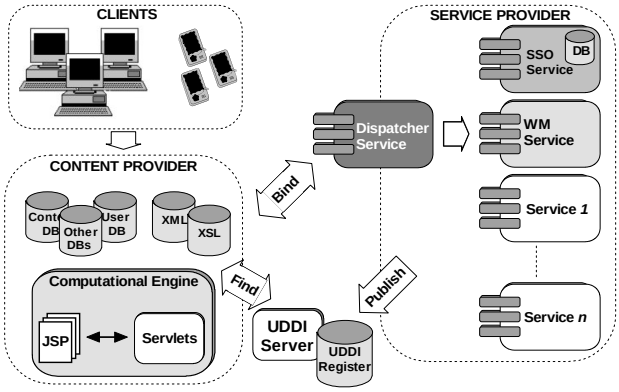

Figure 4: The DRM platform.

The choice of implementing an SP as a dispatcher and a set of web services hidden from users and CPs is motivated by the following considerations: (1) the dispatcher can act as a proxy for the web services; (2) a "single sign-on" (SSO) service can be exploited to control the interactions to and among the web services. In fact, the presence of an SSO service, which can itself be structured as a web service within the back-end tier, enables the dispatcher to sign-on only once in order to gain access to all the web services implemented by the SP. Therefore, even though the adoption of a dispatcher might represent a bottleneck for the back-end tier, the presence of a unique point of authentication/authorization enhances the security and increases the performances of the back-end tier, because the security credentials of the dispatcher do not need to be communicated to web services each time it wants to access them.

4.2 The interaction protocol

Figure 5 shows the interaction protocol assumed by the platform, in which the CP allows users to access its web servers via terminals with a varying degree of functionalities. It exposes a "registration" service finalized both to acquire user information and to implement access control, user tracking and billing. The registration phase and all the subsequent communications involving the exchange of private information, such as payment, take place over SSL/TLS channels. This first phase is also finalized to obtain or generate an XrML (eXtensible rights Markup Language) document associated to the user and the required multimedia document.

When a registered user chooses a content, a servlet running on the CP server decides if to protect it. If the content is, for example, an MPEG-2 video, it may decide to watermark it. Then, the CP can directly contact a known SP that offers a watermarking procedure for that video type or search UDDI registries to discover the demanded service. In this phase, besides the discovery information and according to what reported in Section 3, the CP receives the specific program to carry out the bitstream-to-XML conversions in order to transform the master videos contained in its repositories in a format based on the elaborated variant of the BDSL.

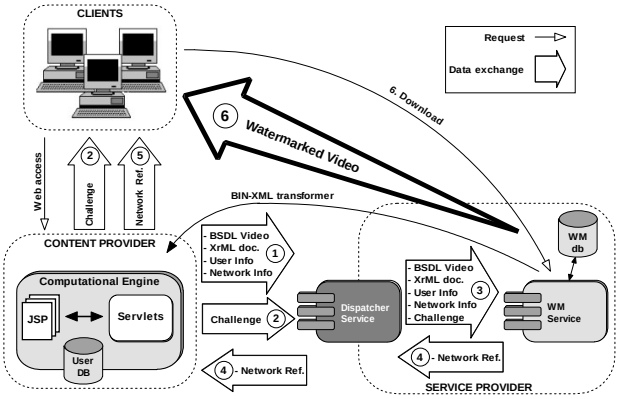

Figure 5: The interaction protocol.

To watermark the selected video according to the procedure described in Section 2, the following steps are to be taken: (1) the CP sends to the SP the video, the user XrML document, and some other information, such as the user profile, his terminal type, and the characteristics of user network connection; (2) the CP generates some data that it communicates both to the SP and the user, and that will be then exploited by the user to issue a challenge to authenticate the SP from which he/she will download the protected video; (3) the SP dispatches the video and all the information received from the CP to the web service implementing the watermarking; (4) the web service exploits the information received to calculate the parameters needed to run the procedure, such as the u_j and p_i sequences and the outputs of the Φ, Ψ and Δ functions, stores the information in the WM database, runs the procedure, and, as soon as it has watermarked a sufficient part of the video, returns its own network reference to the dispatcher, which forwards it to the CP; (5) the CP communicates the network reference of the web service to the user, thus allowing him/her to download the protected video; (6) the user authenticates the web service and starts the download.

It is worth noting that: (1) the communications taking place during the above steps exploit SSL/TLS channels; (2) the information stored by the SP in its database is coded in an internal format; (3) the SP does not return any information about the embedded watermark to the CP, and this makes the readable watermarking procedure blind and based on a not publicly available decoder, thus taking into account also

the buyer rights according to what reported in Section 1; (4) to speed up the service, the download takes place directly between the user and the watermarking server, and the watermarking procedure exploits Java multithreading. In fact, the web service can start watermarking before receiving the whole video, and can also begin to return the video to the user before having completely watermarked it.

5 Experimental results

A preliminary release of the proposed watermarking procedure has been tested within the web platform described in Section 4. In particular, the procedure has been evaluated by performing some relevant attacks that claim to render the embedded watermark not readable. Tables 3, 4 and 5 summarize the results obtained respectively under three different attacks: the IBM attack (Craver, Memon *et al.* 1998), frame dropping and frame averaging. The first attack is considered an "ambiguity attack" in that it attempts to discredit the authority of the watermark by embedding one or several additional watermarks such that it is unclear which was the first, authoritative watermark. However, in this context the IBM attack is exploited to add noise to the video so as to obscure the original watermark. The second attack can be considered a "simple attack" or a "detection-disabling attack", in that it attempts to impair the embedded watermark by manipulations of the whole watermarked data (host data plus watermark), without an attempt to identify and isolate the watermark. The third attack is a "removal attack" in that it attempts to estimate the watermark, separate the watermarked data into host data and watermark, and discard only the watermark (Hartung, Su & Girod 1999).

The videos used for all the tests are coded in MPEG-2, at 30 fps, each with an initial resolution of 1024×768 pixels. Their duration is about 120 seconds. For each attack, in the tables a pair of values is reported under different values of the video terminal resolution (τ) and of the network connection (η): the first value is the rounded bit error rate affecting the watermark extraction, while the second value is the so called PSNR (peak signal-to-noise ratio) value. Both values are calculated as the mean over the tested videos. In particular, the user sequence u_j is assumed 64 bit long, even though only 32 bits have to be considered actually used to identify a user by means of an anticollusion code. In fact, the remaining 32 bits are "check bits" needed to implement an error-correcting code. Therefore, this hypothesis allows a maximum bit error rate (MbER) equal to 9% (6 bit error) to be tolerated. Furthermore, the PSNR has been estimated as the mean calculated over all the I-frames contained in the watermarked and the attacked video. This way, the PSNR can estimate the quality of the two compared videos. In particular, the used PSNR definition has been:

$$10 \cdot \log\left(\frac{255^2}{MSE}\right)$$

where MSE is for the "mean squared error" computed on I-frames belonging to watermarked and attacked videos. To this end, a PSNR equal to 40 decibels is nowadays widely assumed as a lower limit for the video quality in a commercial scenario, according to the current literature (Wang *et al.* 2002). Therefore, an attack can be considered valid only if the MbER is greater than 9% and the PSNR is greater than 40 db. Obviously, the limit of 40 db has not to be considered a hard video quality threshold, but only an estimate. In fact, the developed analisys can be conducted also under a different limit.

	320×240	640×480	1024×768
modem or GPRS	20%/34,2db	13%/41,6db	7%/40,3db
DSL, LAN	17%/32,8db	9%/39,9db	4%/38,6db
T1, T3	13%/30,4db	4%/38,5db	1%/37,1db

Table 3: The watermarking procedure analysis under the IBM attack.

Table 3 shows that the procedure achieves a good performance under the IBM attack. In fact, for low values of τ and η the attack is successful, but the final video quality results low because the video, due to the re-quantization, is not able to contain the further information needed to make the watermark not readable. On the contrary, for high values of τ and η the procedure results to be secure and the attack cannot impair the embedded watermark: the MbERs are prevalently less than 9%. It is also worth noting that the PSNRs tend to assume lower values when τ and η become high, and this because the amount of information embedded in this hypothesis by the watermarking procedure and by the performed attack increases, thus exceeding the video capacity.

	320×240	640×480	1024×768
modem or GPRS	25%/38,3db	18%/44,7db	12%/42,9db
DSL, LAN	22%/37,1db	13%/43,6db	9%/40,3db
T1, T3	16%/35,2db	8%/42,5db	7%/38,7db

Table 4: The watermarking procedure analysis under the frame dropping attack.

The frame dropping attack tries to disable the watermark extraction by removing trunk of frames. In particular, when the dropping rate of video frame is high, errors are introduced to the whole watermark, making the procedure performance poor. However, this also leads to a significant damage to the video, and the results reported in Table 4, obtained under a value of frames dropped about 20%, reflect this condition. In particular, the successful attacks performed under some values of η and τ can be contrasted by increasing the number of the check bits used to implement the error-correcting code.

	320×240	640×480	1024×768
modem or GPRS	11%/45,7db	9%/44,5db	7%/44,1db
DSL, LAN	8%/43,2db	7%/43,4db	4%/43,6db
T1, T3	7%/41,1db	4%/42,7db	1%/42,4db

Table 5: The watermarking procedure analysis under the statistical averaging attack.

In the statistical averaging attack a high number of watermarked frames are collected so as the watermark can be estimated by statistical averaging. The attack has been performed by colluding about the 70% of the available frames and the obtained results are shown in Table 5. In particular, the procedure exhibits a good performance and this is essentially due to its adaptive behavior that can balance the final video quality with the achieved protection level.

Finally, it is worth noting that the PSNR values obtained during the tests above described demonstrate that the procedure can successfully protect the videos as well as reduce the final video quality to the allowed minimum values. In fact, one of the interesting aspect of the procedure, emerged from the test phase, is that, if the re-quantization phase reduces the video quality to a PSNR value close to a predefined lower limit, such that of 40 db, and the subsequent watermark embedding is carried out taking care of

saturating the video capacity without further reducing the final value of the PSNR, attacks to impair the embedded watermark end up obtaining PSNR values much lower than the assumed limit, thus degrading the final video quality.

6 Related work

The literature proposes many solutions for digital watermarking, but very few schemes have been developed to be adapted to an on the fly use in a web context, while no schemes present an adaptive behavior. In practice, most schemes have been designed for still images or uncompressed videos and do not match all the requirements reported in Section 1. Furthermore, no watermarking schemes have been implemented by exploiting XML based technologies, which make the procedure well suited to be integrated in a DRM web platform like the one proposed in Section 4, which, differently from others, such as MOSES (Moses 2002), allows distinct web entities to dynamically interoperate in a secure network context without having to make the provided services compliant to complex frameworks. In particular, this goal is also achieved by two important DRM platforms, the MS Windows Media Rights Manager (WMRM) and the IBM Electronic Media Management System (EMMS). In fact, both are provided with software suites of a lot of components that can interact to provide content owners, businesses, retailers and consumers with solutions for their digital distribution needs. Moreover, both comprise SDKs able to promote the integration of the provided DRM services into existing web applications and enterprise portals. However, both are based on commercial technologies and not freely available products. More precisely, the WMRM presents the following drawbacks: (1) it can only manage multimedia documents saved in specific MS digital formats, such as the Windows Media Video format; (2) users need a player that supports the WMRM, such as the Windows Media Player, to play the protected documents; (3) the web application development is strongly based on the MS .NET technology.

In (Memon & Wong 1998) an interactive buyer-seller protocol for invisible watermarking in which the seller does not know the exact watermarked copy that the buyer receives is presented. The protocol does not allow the seller to create copies of the original document containing the buyers watermark. However, in case the seller finds an unauthorized copy, he/she can identify the buyer from whom this unauthorized copy has originated and furthermore can also prove this fact to a third party by means of a dispute resolution protocol. The watermark embedding protocol is based on public key cryptography and exploits an SS insertion technique. Although the solution proposed in (Memon & Wong 1998) can take advantage from the adoption of the improved SS technique, its exploitation in a web scenario results to be difficult, since the dispute resolution protocol is a 3-party protocol, that is, it requires the buyer to participate in order to prove his/her innocence in case the seller accuses him of making unauthorized copies. Moreover, users have to be provided with certificated public keys if they want to buy the protected documents distributed by CPs, and exchanges of encrypted data among sellers and buyers are needed in order to generate the marks to be inserted in the protected documents. Finally, the copyright violater identification protocol used to discover an unauthorized copy of a document is based on a watermark extraction procedure that needs to compare the original and the unauthorized copy of the document, and this is considered a disadvantage for security in a web environment (Zeng & Liu 1999).

The main drawbacks of the solution proposed in (Memon & Wong 1998) also affect the solution presented in (Pfitzmann & Waidner 1997), which forces users to be provided with certificated public keys and needs exchanges of encrypted data among sellers and buyers to generate the marks to be inserted in the protected documents.

In (Hartung & Ramme 2000) watermarking is presented as an essential component of modern DRM systems, which can have a strong impact on the commerce of multimedia contents. In fact, the authors highlight that secure multimedia applications need to be adapted for modern mobile telecommunications systems. To this end, the proposed adaptive behavior goes just in this direction, making the watermarking process dependent of user terminals and network connections.

In (Fei, Kundur & Kwong 2004) an important study about the performance of the SS and QIM watermarking approaches for still images in the presence of lossy compression is reported. The study shows that SS and QIM based watermarking schemes have different characteristics of robustness to JPEG compression: SS watermarking is more robust to higher levels of JPEG compression, while QIM watermarking does not experience host signal interference which dominates for low compression ratios. Although the reported results concern a scheme where watermarking occurs on still images before lossy compression, the study confirms that the idea exploited by the proposed procedure of removing the host signal as source of interference in the watermark embedding can produce a dramatic improvement in the quality of the protection process. Therefore, the proposed procedure, by adopting and adaptive scheme and exploiting the re-quantization process as a further mechanism to improve the video protection level, can enhance its performance with respect to other SS watermarking schemes without requiring complex watermark decoders or the adoption of hybrid solutions, such as the one proposed in (Fei, Kundur & Kwong 2004).

7 Conclusions

In this paper a watermarking procedure for the copyright protection of MPEG-2 videos distributed in the Internet is described. The watermarking procedure directly acts on compressed video streams and is implemented as an adaptive, on buyer variant of the improved spread spectrum scheme described in (Malvar & Florêncio 2003). The procedure uses an anticollusion code to increase security against average and collusion attacks, and is implemented by exploiting XML based technologies that enable the video quality adaptation and watermark embedding to be considered as normal editing operations on XML files. Furthermore, the paper presents a simplified but flexible DRM web platform implemented by using web service technologies and by which SPs can supply copyright protection services on behalf of CPs in a secure network context. Thus, CPs that exploit the proposed platform can take advantage of a copyright protection system acting as a TTP without having to directly implement it. On the other hand, SPs can follow the proposed XML based approach to implement fully in the XSLT language and Java complex watermarking procedures without having to interface native codes with web services frameworks commonly used to develop web platforms.

The experimental results confirm that a spread spectrum based watermarking procedure can be made robust against a variety of manipulations by performing some improvements that do not penalize efficiency

and flexibility, and this makes the procedure suitable to be exploited in a web context, where an on the fly behavior is required. Moreover, the adaptive behavior of the procedure allows to achieve a trade-off between protection needs and the final quality of the distributed videos. This way, whenever an attack attempts to impair the embedded watermark, the final video quality ends up being degradated, thus making the attacked videos useless in commercial web applications and not interesting to Internet pirates.

References

Amielh, M. & Devillers, S. (2002), Bitstream Syntax Description Language: Application of XML-Schema to Multimedia Content Adaptation, *in* '11th International World Wide Web Conference', Honolulu, Hawaii, USA.

Barni, M. & Bartolini, F. (2004), 'Data Hiding for Fighting Piracy', *IEEE Signal Processing Magazine*, **21**(2), 28-39.

Barni, M., Bartolini, F. & Furon, T. (2003), 'A general framework for robust watermarking security', *Signal Processing*, **83**(10), 2069-2084.

Brunner, R., Cohen, F. *et al.* (2001), *Java Web Services Unleashed*, SAMS Publishing.

Chen, B. & Wornell, G. (2001), 'Quantization index modulation: a class of provably good methods for digital watermarking and information embedding', *IEEE Transaction on Information Theory*, **47**(4), 1423-1443.

Cox, I., Bloom, J. & Miller, M. (2001), *Digital Watermarking: Principles & Practice*, Morgan Kaufman.

Cox, I., Kilian, J., Leighton, F. & Shamoon, T. (1997), 'Secure spread spectrum watermarking for multimedia', *IEEE Transaction on Signal Processing*, **6**(12), 1673-1687.

Craver, S., Memon, N., Yeo, B.-L. & Yeung, M. (1998), 'Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks, and implications', *IEEE Journal on Selected Areas in Communications* (Special issue on Copyright and Privacy Protection) **16**(5), 573–586.

Fei, C., Kundur, D. & Kwong, R. H. (2004), 'Analysis and Design of Watermarking Algorithms for Improved Resistance to Compression', *IEEE Transaction on Image Processing*, **13**(2), 126-144.

Hartung, F. & Girod, B. (1997), Digital Watermarking of MPEG-2 Coded Video in the Bitstream Domain, *in* 'International Conference on Acoustics, Speech, and Signal Processing', Vol. 4, Munich, Germany, pp. 2621–2624.

Hartung, F., Su, J. K. & Girod, B. (1999), Spread Spectrum Watermarking: Malicious Attacks and Counterattacks, *in* E. J. Delp & P. W. Wong, eds, 'Security and Watermarking of Multimedia Contents', Proceedings of SPIE, Vol. 3657, pp. 147–158.

Hartung, F. & Ramme, F. (2000), 'Digital Rights Management and Watermarking of Multimedia Content for M-Commerce Applications', *IEEE Communication Magazine*, **38**(11), 78–84.

Lei, Z. & Georganas, N. D. (2002), 'Rate Adaptation Transcoding for Precoded Video Streams', *in* Proceedings of the 10th ACM international Conference on Multimedia, Juan-les-Pins, France, pp. 127–136.

Lin, C.-Y., Wu, M., Lui, Y.-M. *et al.* (2001), 'Rotation, scale, and translation resilient public watermarking for images', *IEEE Transaction on Signal Processing*, **10**(5), 767-782.

Lubin, J., Bloom, J. & Cheng, H. (2003), Robust, content-dependent, high-fidelity watermark for tracking in digital cinema, *in* E. J. Delp & P. W. Wong, eds, 'Security and Watermarking of Multimedia Contents V', Proceedings of SPIE, Vol. 5020, pp. 536-545.

Malvar, H. S. & Florêncio D. A. F. (2003), 'Improved Spread Spectrum: A New Modulation Technique for Robust Watermarking', *IEEE Transaction on Signal Processing*, **51**(4), 898-905.

Memon, N. & Wong, P. W. (1998), A Buyer-Seller Watermarking Protocol, *in* 'IEEE Workshop on Multimedia Signal Processing', Los Angeles, CA, USA, pp. 278–283.

Moses (2002), Web site of MOSES EC IST project: http://www.crl.co.uk/projects/moses

Pfitzmann, B. & Waidner, M. (1997), Asymmetric Fingerprinting for Larger Collusions, *in* '4th ACM Conference on Computer and Communications Security', Zurich, Switzerland, pp. 151–160.

Trappe, W., Wu, M., Wang, Z. J. & Liu, K. (2003), 'Anti-collusion fingerprinting for multimedia', *IEEE Transaction on Signal Processing*, **41**(4), 1069-1087.

Wang, Y., Ostermann, J. & Zhang, Y. (2002), *Video Processing and Communications*, Prentice Hall.

Zeng, W. & Liu, B. (1999), 'A Statistical Watermark Detection Technique without Using Original Images for Resolving Rightful Ownerships of Digital Images', *IEEE Transaction on Image Processing*, **8**(11), 1534-1548.