Content Protection and Tracking for Digital Rights Management in Digital Libraries

王正豪
國立台北科技大學資訊工程學系助理教授
Jenq-Haur Wang
Assistant Professor, Department of Computer Science and Information Engineering,
National Taipei University of Technology
E-mail：jhwang@csie.ntut.edu.tw

Keywords（**關鍵詞**）：**內容保護**(Content Protection) ；**內容追蹤**(Content Tracking)；**數位圖書館**(Digital Libraries)；**數位版權管理**(Digital Rights Management)

【摘要】

數位圖書館一般採用存取控制與數位浮水印等方式來保護數位內容，然而這些方式有其限制。首先，經過身份驗證與授權之合法使用者可以輕易將內容再次傳播出去。其次，數位浮水印的方法大多無法有效抵擋各種影像處理的攻擊。因此，如何在數位內容傳播至合法使用者後仍能保護其合法使用是目前相當重要的挑戰與議題，數位版權管理系統是目前常見的解決方法之一。然而此類系統只能在受保護的可靠環境(trusted environment)，根據一致的權利政策，提供完整的內容保護，因此有其實用的困難。

在本文中，我們提出一個整合式內容保護與追蹤架構，並結合了靜態的權利控管與動態的內容追蹤，目標在有效偵測網路上非法的侵權行為。首先，我們介紹一種 wrapper-based 權利控管方法，整合了數位浮水印、密碼學、資訊保護技術，及權利模型(rights model)。其次，我們也提出了一個多媒體內容近似複製的偵測追蹤技術，作為內容保護的第二道防線。在受 wrapper 保護的環境中，透過監控各種內容播放程式，數位內容唯有在符合使用規則的情形下才能被存取。更重要的是，我們所提的架構可以很容易與現有的內容播放程式及 DRM 系統整合，實驗結果展示了此架構的有效性與複製偵測的準確度。

【Abstract】

Conventional digital libraries utilize access control and digital watermarking techniques to protect their digital content. These methods have some limitations. First, after passing the identity authentication process, authorized users can easily redistribute the digital assets. Second, it is impractical to expect a digital watermarking scheme to prevent all kinds of attacks. Thus, how to enforce property rights after digital content has been released to authorized users is a crucial and challenging issue.

Digital rights management (DRM) systems have been proposed to address this issue by enforcing the rights access policies in a trusted computing environment. However, DRM systems can only be useful if the computing environment can be protected and compliant to the common rights policy throughout the lifecycle of digital objects.

In this paper, we propose an integrated framework of content protection and tracking that aims to detect unlawful copyright

infringements on the Internet, and combines the strengths of static rights enforcement and dynamic illegal content tracking. First, we introduce a wrapper-based approach to digital rights enforcement for content protection that integrates digital watermarking, cryptography, information protection technology, and a rights model. Also, we present a content tracking mechanism for multimedia-content near-replica detection as the second line of defense. In the rights enforcement environment, the behavior of all content players is monitored and digital content can only be accessed after certain usage rules have been satisfied. Furthermore, the proposed architecture can be easily integrated into any digital content player, or even existing DRM systems in digital libraries. With the protection of the proposed framework, the abuse of digital content can be drastically reduced. Our experiments demonstrate the efficacy of proposed framework and the accuracy of copy detection.

INTRODUCTION

With rapid development of the Internet and computer technology, digital content, including digital images, video, and music, can be distributed instantaneously across the Internet. However, digital content in digital world differs from objects in real world, since it can be easily copied, altered, and distributed to a large number of recipients. This could cause copyright infringement and revenue losses to content owners. Thus, protection of the copyrights and revenues of content owners in digital libraries has become increasingly important in recent years. The National Digital Archives Program (NDAP) [1] in Taiwan has amassed a rich collection of cultural and historical artifacts. These assets have been digitized to enhance their preservation, and make them more accessible to users. The metadata and digital content storage or archival systems both face the problem of piracy. Thus, content holders from museums and archives are sometimes unwilling to release digital content, because their intellectual property rights could be infringed. To protect high-value digital content and avoid digital piracy, we need a system that prevents unauthorized access and manages content usage rights.

For digital content protection, a number of approaches have been proposed. Basic cryptographic schemes are commonly used to encrypt sensitive content. Access control mechanisms are widely deployed to block unauthorized access in archival systems. However, these only address part of the issues. The security of encrypted content depends on the strength of encryption scheme and the privacy of encryption keys. The content would still be redistributed after it's decrypted. Access control schemes could become useless if the system accounts were hacked.

Digital watermarking is the most widely used technique in content protection. A digital watermark, which is an identification code that carries information about the copyright owner, is invisible and permanently embedded in digital data for copyright protection, proof of ownership, and integrity checks of digital content. It can also provide evidence of copyright infringement. Though useful, watermark-based protection systems have some significant limitations. First, watermarking could degrade the quality of digital content. Second, embedded watermarks are not expected to survive under several kinds of attack. In practice, although many techniques have been proposed, watermark-based techniques are not robust enough to prevent malicious users removing watermark via post-processing.

The Digital Rights Management (DRM) system is another popular method for protecting high-value digital assets. DRM is a protocol of hardware and software services and technologies that governs the authorized use of digital content and manages its use throughout the entire lifecycle of the content (as defined by IDC). The primary objective of DRM is to build a digital rights enforcement (DRE) environment that only allows access to protected content under the conditions specified by the content owner. Many DRM and DRE frameworks have been proposed

in recent years. Although these architectures provide a way to construct a copyright protection environment, the security of digital content is not fully addressed. For example, in the area of rights enforcement, authorized users could still distribute digital assets easily after they pass the identity authentication process. Hence, how to enforce the usage rules and protect content owners' property rights after digital content have been released is still a challenge in DRM research.

Recently, the concept of content-based copy detection has been proposed as a complementary solution to conventional DRM systems. The idea is that, instead of hiding additional information in the digital content (such as digital images and videos) for copy detection, the content itself can be employed for the same purpose. A content-based copy detection system works as follows. It starts by extracting features from the original content, and compare them with features extracted from a suspected copy to determine whether the latter is really a copy. Content-based copy detection itself can be used to identify illegal copies, and it can also be used to complement digital watermarking techniques.

Existing content-based image copy detection techniques emphasize on finding unique image features with good performance that could resist a variety of image attacks, but finding a globally effective feature is difficult, and in many situations, domain dependent. Hence, the accuracy of image copy detectors is still restricted.

With respect to video copy detection, most approaches employ high-cost computation techniques to match videos, whereby a fix-sized window that slides frame by frame is used to detect copies. However, the sliding window cannot handle some temporal variations, e.g., fast and slow motion. These drawbacks inevitably impede the practicability of the system.

As current copy protection technologies have certain limitation, in this paper, we seek to address the problem by introducing a novel architecture that integrates a wrapper-based DRM system with an effective content tracking mechanism to discourage attackers and further strengthen the proposed system's security. First, the wrapper-based DRE monitors the user behaviors and enforces the rights policies. Then, the content tracking mechanism utilizes Web content collecting/sampling and content-based copy detection techniques to identify possible copies. The experimental results show good performance in both image and video copy detection.

The remainder of this paper is organized as follows. Section 2 lists related work. The proposed framework for DRM system is described in Section 3. In Section 4, we present some discussions about the wrapper-based approach and show the experimental results to demonstrate the effectiveness of content tracking mechanism. Our conclusions are presented in Section 5.

RELATED WORK

To prevent the abuse of digital content, many approaches have been proposed. For example, cryptographic schemes are commonly used to encrypt sensitive content, and access control mechanisms are built to block unauthorized access in archival systems. However, cryptographic schemes might be compromised if the encryption key is not properly managed. Access control mechanism could become useless if the system accounts get hacked.

Most digital libraries and museums adopt digital watermarking techniques to guard their digital images. Though useful, watermark-based image protection systems are still not robust enough to resist a variety of image attacks. Digital Rights Management (DRM) is a chain of hardware and software services and technologies governing the authorized use of digital content and managing any consequences of that use throughout the entire lifecycle of the content (as defined by IDC). DRM is a new concept that can be used to protect high-value digital assets and

control their distribution and usage. The design of a DRM system must address the following key issues: (1) a digital rights enforcement (DRE) environment, (2) digital rights, and (3) standardization for interoperability. **Guth** proposed a typical DRM system architecture, including several essential components. **Jamkhedkar and Heileman** proposed that DRM should be adopted as a layered framework, whereby various services are offered to users of the digital content at each layer. In addition, **Popescu** et al. proposed a security architecture that enables digital rights management of home networks. The concept of an "authorized domain" is used to authenticate compliant devices, instead of relying on expensive public key cryptographic operations. Although the above works suggest novel architectures for a DRM system, they do not fully address the three issues mentioned earlier.

For example, in the area of rights enforcement, authorized users could still redistribute digital assets easily after they pass the identity authentication process. To overcome this problem, Nicolakis et al. developed a DRM system called MediaRights that protects digital images. However, although this kind of architecture solves the rights management problem, a customized image viewer is not convenient for users. Furthermore, the circulation of digital assets is seriously impaired. Hence, how to enforce the usage rules and protect content owners' property rights after images have been released are the major challenges in DRM research. Several commercial DRM solutions, such as InterTrust, Alpha-Tec, Digimarc, and LTU, are available.

But the system requirements of digital rights management in digital libraries vary enormously and differ from those of industry. It is very unlikely that existing commercial systems can meet the demands of digital libraries. Building a DRM system for digital libraries based on existing commercial solutions without any modification is therefore impractical.

THE PROPOSED APPROACH

Most of the valuable digital content to be protected in archival systems consists of multimedia objects, such as digital images and videos. Figure 1 gives an overview of the proposed integrated framework for a DRM system, which consists of three building blocks: (1) the Digital Rights Enforcement (DRE) Environment, (2) the Digital Watermark Module, and (3) the Content Tracking Module. First, the system packages the content to be protected in a secure manner, and the DRE environment ensures that the usage rules are enforced. We use a wrapper-based DRE technique to protect the digital rights.

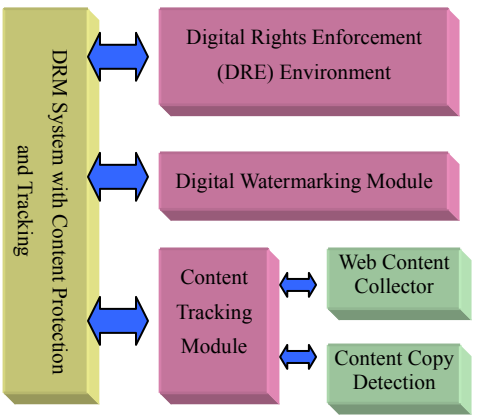

Figure 1. Overview of the proposed framework

When a user downloads digital content from the network and views it on a content player (e.g., a browser), the wrapper automatically monitors the user's behavior. If the rules are violated, or the user refuses to be monitored by the wrapper, the content is rendered unavailable. The second component, the digital watermarking module, can embed an invisible digital watermark into digital content. If necessary, the content holder can extract the watermark to prove ownership and check if there is copyright infringement.

The third component, the content tracking module, can be regarded as the second line of defense. It is composed of two key kernels: a

Web content collector/sampler to collect or sample suspicious content from the Web and an *image/video copy detector* which can determine whether or not suspicious digital content is copyrighted (registered). By integrating a Web image/video search engine or a Web crawler with the content tracking module, illegal use of digital content on the Internet can be detected automatically in a large scale.

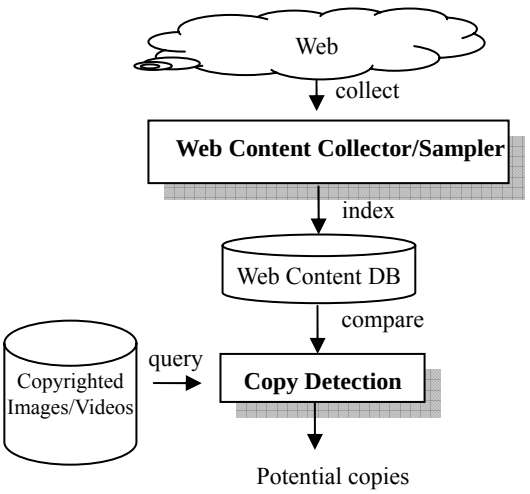

Figure 2. Block diagram of the content tracking module.

As shown in Figure 2, the content tracking module first registers the image/video in the database. Only feature vectors are stored in the database in order to accelerate the detection process and reduce the amount of storage space required. The image/video copy detector then conducts a matching process to determine whether the suspicious digital content collected by Web content collector is copyrighted.

In the following subsections, we will focus on the wrapper-based DRE and the content tracking methods and give individual descriptions.

A Wrapper-based DRE

Figure 3 gives an overview of the wrapper-based DRE, which consists of two building blocks: the server side encoding module called the *Digital Content* (*DC) Packager* and the client side decoding and protection module called the *DC Wrapper*.

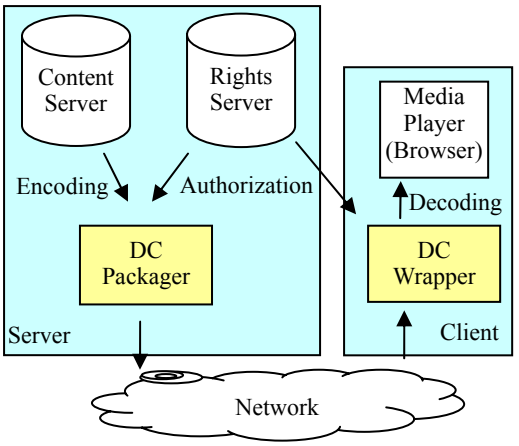

Figure 3. Overview of the Wrapper-based DRE

To facilitate player-independent encapsulation of rights and encryption information, we propose the following wrapper-based approach. First, digital content is separated from its usage rules and they are stored on content server ands rights server, respectively. DC Packager encodes digital content and the related rights usage rules and packages them into a protected content file. Then, DC Wrapper decodes the protected content, launches a content player, and monitors its behaviors.

Rights Expression Model

On the rights server, we allow the content owner to specify and store the access rights of different users on various contents. There are several existing rights models and rights expression languages (RELs), such as XrML (eXtensible rights Markup Language) [2] and ODRL (Open Digital Rights Language) [3]. We use XrML as our REL, which is a general purpose REL with good expressive power.

For content protection in digital libraries, it is only necessary to consider the following restrictions: play/view, print, save, valid duration, and authorized player. A user can only play, print,

or save digital images when he is authorized to do so. The valid duration limits the time that digital content is accessible, while the authorized player ensures that digital images can only be accessed by specific machines/players.

DC Packager

The DC Packager envelops digital images in a protected file.

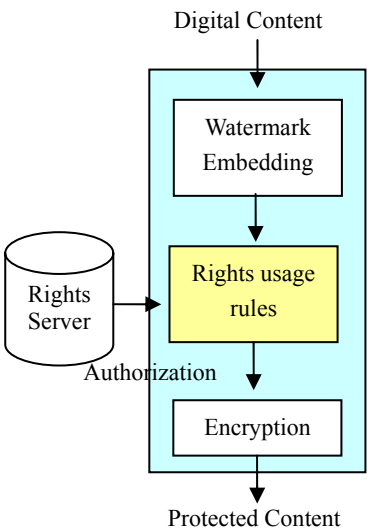

Figure 4. An operational view of the DC Packager

As shown in Figure 4, an invisible digital watermark representing the copyright of NDAP is first embedded into images prior to release. The usage rules derived from the rights server are then combined with the watermarked image to form a new content package, which is then encrypted for security.

The resulting file is called a "protected digital content" file, meaning that the digital image is ready for distribution, and that the usage rules can be enforced with certainty.

DC Wrapper

DC Wrapper enforces the rules related to the use of digital images.

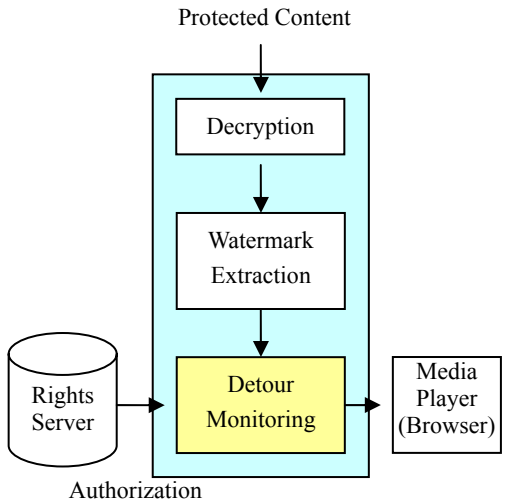

Figure 5. An operational view of the DC Wrapper

As shown in Figure 5, after a user downloads a protected file from the network and views it on a media player (e.g., a browser), DC Wrapper launches automatically and monitors the user's behavior. Furthermore, the digital images are released with specific restrictions on their usage. If these rules are violated, or a user refuses to view images under the monitoring of DC Wrapper, the content is rendered unavailable. DC Wrapper is implemented with a binary instrumentation technique called *Detours*, which intercepts OS-level events and messages by re-writing target function images. Note that the wrapper-based design allows more flexibility in the choice of an underlying content player which is independent of the DRM modules. DC Wrapper decodes encrypted digital content based on predefined rules, and transforms the content into a readable format for the content player.

Note that DC Wrapper monitors the behavior of the player rather than acting as a multimedia player itself. Hence, there is no need to use a customized player to play digital content. The rights information provided by the rights server is employed by DC Wrapper to determine the access rights for a user.

Content Tracking Methods

Image Copy Detection

Previous researchers have tried to find an image feature that can be employed universally for copy detection. Various features have been studied, for example, local, global, DCT-based, wavelet-based, geometrically variant, and geometrically invariant.

Obviously, the accuracy of existing copy detectors relies heavily on the robustness of the feature used, and on a suitable threshold that can balance the false rejection and false acceptance rates. However, it is difficult to find a unique feature that is invariant to all kinds of attack. Another limitation of existing approaches is that they lack a mechanism to exploit useful prior information, such as possible attack models, to boost the copy detection performance – even when such information is easy to generate or acquire.

Hence, instead of extracting the feature vector from a copyrighted image, we use *virtual attacks* as prior guidance to conduct a new copy-detection framework. Typical attacks considered in our approach include signal-processing attacks, geometric attacks, and image-compression attacks. By applying the attacks to a copyrighted image, a set of novel images can be generated. Both the copyrighted and novel images are processed by extracting their features, where the features extracted from the former and the latter are referred to as the *original* and *extended features* in our framework, respectively. Figure 6 shows the concept of copy detection in a 2-dimensional feature space.

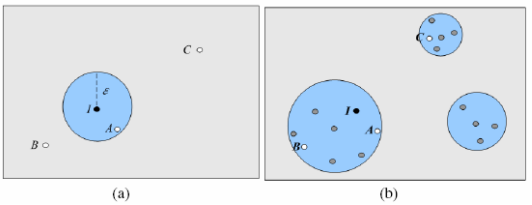

Figure 6. (a) A typical image copy detection algorithm. (b) Using EFS to solve the problem in (a).

In Figure 6(a), *I* denotes the feature vector of a copyrighted image, and *A*, *B*, and *C* are the modified copyrighted images under some malicious attacks. The radius ε denotes the error tolerance for copy detection in the feature space, which is determined by a predefined threshold. It often occurs that after some attack, the modified image, say *A*, can be successfully identified, but the other more heavily modified ones *B* and *C* cannot be detected since they are far away from *A* in the feature space. In our experience, this problem is difficult to solve in practice by simply changing the features being used. Figure 6(b) shows the concept of using *extended feature set* (*EFS*) to enhance the performance of copy detection, where the gray points denote the extended features. In this case, the problem can be solved by grouping features so that the modified images *A*, *B*, and *C* can be correctly identified.

Although modeling copy detection as a one-class classification problem is likely to boost the system's performance, many empirical studies of pattern classification reveal that the classifier can be trained better if more prior knowledge is given. In particular, if some negative examples are available, using them would help build a better classifier than using only positive examples. Therefore, in our approach, not only positive examples (where they are mainly extended features), but also negative examples are used. The negative examples are easy to acquire or generate; for example, they can be obtained from the Internet. Also, a registered image can serve as a negative example of other registered images. Our framework transforms the copy-detection problem into a two-class classification problem. We demonstrate by experiments that our approach generally outperforms the conventional technique when the same feature space is employed.

A popular method for solving the two-classification problem is based on Gaussian Mixture Model (GMM), defined as:

$$f_k(x \mid \theta) = \sum_{j=1}^{k} w_j g(x \mid \lambda_j),$$

where $g(x \mid \lambda_j)$ is a multivariate Gaussian distribution, $\lambda = (u, \Sigma)$ is the Gaussian component parameter set, w_j is the weight of jth component, k is the number of Gaussian components, and $\theta = \{w_j, \lambda_j \mid j = 1, 2, \ldots, k\}$ is the model's parameter set.

To learn the GMM model for each class, we apply the expectation-maximization (EM) algorithm that can converge to a maximum likelihood estimation of the parameter set. The

selected cluster number k is a critical factor in training a GMM. Since we have prior categorical knowledge about our training data, the number of clusters can be set, in advance, as the number of attacks we would like to model. To improve the accuracy, k can also be assigned automatically by maximizing the loglikelihood of the training samples, and estimated via cross-validation. In our approach, we initially set k as the category number, and continue adding clusters until the loglikelihood either (1) starts to decline or (2) keep on increasing but with an amount less than a specific threshold. In Section 4, we conducted some experiments to examine the performance of the proposed framework when a Gaussian mixture classifier is used.

Video Copy Detection

The problem definition of video copy detection is to determine if a given video clip (query) appears in another video clip (target) which is suspected to be a possible copy of copyrighted videos. However, if it does appear, we need to determine its location. The proposed video copy detection module consists of three steps: key frame extraction, candidate clip selection, and sequence matching. Suppose that Q_V and T_V are the query and target video clips, respectively. Q_V is represented as $\{q_{vj} \mid j = 1, 2, \ldots, N\}$, and T_V as $\{t_{vi} \mid i = 1, 2, \ldots, M\}$, where M and N are the number of frames in each video clip, $M >> N$ and t_{vi} and q_{vj} are the ordinal signatures of the corresponding frames. The details of the ordinal signature are as follows: A video frame is partitioned into $n_x \times n_y$ blocks and the average intensity level in each block is computed. In our case, we utilize 3×3 block of each frame for ordinal signature extraction. Then we sort the set of average intensities in ascending order and a rank is assigned to each block. The ranked $n_x \times n_y$ dimensional sequence is then generated. Thus a video frame is represented by a 3×3 matrix of ranked ordinal signatures. We then reshape the matrix to a 9×1 vector. Based on the steps mentioned above, the task of copy detection is to find the subsequences from T_V, whose signature sequences are similar to those of Q_V .

The first step is to extract key frames from video clips. Besides reducing the storage and computational costs, it can also moderate the effects of temporal variations. Let us take the target clip T_V as an example. In order to search the peak or foot of a sequence, we define a 9×9 Laplacian of a Gaussian filter F, which is often used to calculate second order derivatives in a signal:

$$F(x,y) = -\frac{1}{\pi\sigma^4}\left|1-\frac{x^2+y^2}{2\sigma^2}\right| e^{\frac{x^2+y^2}{2\sigma^2}},$$

The second order derivatives reveal signal transitions, which can be chosen as key frames.

We then convolute F and T_V to obtain a vector A and find the local extreme on A, as shown in Figure 7. The extracted key frames are denoted as $T_K = \{t_{k1}, t_{k2}, \ldots, t_{km}\}$. For the query clip Q_V, we repeat the above procedure to extract Q_V's key frame sequence $Q_K = \{q_{k1}, q_{k2}, \ldots, q_{kn}\}$.

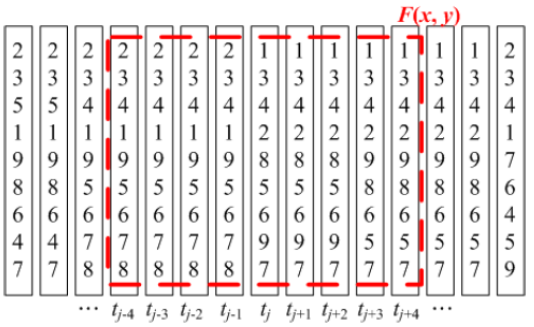

Figure 7. An illustration showing the convolution of the filter F and the target video T.The dashed square indicates the range of F, and t_j is the ordinal signature of the j-th frame in T.

After the key frames have been extracted, the key frame sequence of T_K is still very long. To avoid an exhaustive search of the long sequence, we do a rough scan of T_K to find subsequences that may be copies of Q_K. First we search for the start and end indices of candidates CI_{start} and CI_{end} in T_K. These candidates are frames that are similar to the first and end frames of Q_K (i.e., q_{k1} and q_{kn}). Then we search for the second candidate lists CI_{start} and CI_{end}. A subsequence $C = \{t_{ks}, t_{ks+1}, \ldots, t_{ke}\}$ in T_K is reported as a candidate clip according to following conditions: First, the list

of candidate clips is sorted in ascending order of the start and end indices. Second, the candidate with the smallest indices is selected. Third, candidate clips that are too long or too short are filtered out.

Finally, the sequence matching is done by the Dynamic Time Warping (DTW) algorithm which is applied to compute the similarity between the query example Q_K and the candidate clip C. Since DTW can compensate for differences in length, it is suitable for dealing with temporal variations in videos. We define the following distance function:

$$dist(Q_K, C) = cost(n, l),$$

where n and l are the frame number of Q_K and C respectively, and $cost(n, l)$ is a recursive function:

$$cost(1,1) = \|qk_1 - tk_1\|,$$
$$cost(n,l) = \|qk_n - tk_l\| + min\{cost(n-1,l), cost(n,l-1), cost(n-1,l-1)\},$$

where $||l - n|| \leq$ the maximum warping distance, which is normalized to determine whether C is a copy of Q_K.

EXPERIMENTS AND DISCUSSIONS

In this section, we conducted two experiments to evaluate the performance of content-based copy detection. In the first case, the detection results of image content tracking are presented; while in the second, video data from the National Digital Architecture Program in Taiwan is used to verify the effectiveness of our method. Then, discussion on the wrapper-based DRE is also given.

Performance of Image Detection

We took Kim's approach – DCT ordinal measure as the basis for comparison. In this approach, an input image is divided into 8×8 equal-sized sub-images. Only AC coefficients of the 8×8 DCT coefficients are used as the ordinal measure. We thus generated a 63-dimensional image feature vector.

In the first test, one hundred copyrighted images were registered in the database and used as queries to determine how many modified versions could be successfully detected. A standard benchmark, Stirmark 4.0, was used to generate novel testing data. The image replicas were randomly generated by StirMark 4.0 with two kinds of attacks: *pre-learned* and *novel* attacks (that are not modeled in training). Specifically, we adopted 7 categories of pre-learned image attacks (convolution filtering; cropping; JPEG compression; median filtering; noise adding; scaling; and rotation), and 6 categories of novel attacks, including affine transformation, self-similarity, removal of lines, PSNR, rotation+ rescaling (abbreviated as RRS), and rotation+cropping (abbreviated as RC). Thus, we generated 124 near-replicas for each copyrighted image. In addition to image replicas, 15,000 randomly selected unrelated images were also included in the testing data set, giving a total of 27,400 images for testing.

To evaluate the performance, the precision rate, recall rate, and F-measure are used:

$$F - Measure = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}}.$$

The results in Table 1 show that our framework outperforms that of the DCT ordinal measure. The EFS for the Gaussian mixture model achieves very high precision and recall rate of 96.56% and 93.54% respectively, while the F-measure is 95.03.

Table 1. Average precision and recall rates by using extended features and pure DCT ordinal measures. The response time consists of both the feature extraction and classification times.

Algorithm	Avg. Precision	Avg. Recall	F-Measure
DCT ordinal measures	93.13%	54.79%	68.99
Gaussian Mixture Classifier with EFS	96.56%	93.54%	95.03

The above experiment shows the overall performance of our method. To test the robustness against different attacks, we conducted another smaller-scale experiment in which only the three images shown in Figure 8 were used. This allowed us to further compare the performance of EFS with conventional copy detection method. The results are summarized in Table 2.

Figure 8. Three selected color images in the digital museum (512*512 pixels): a container, a rare book and a painting.

Table 2. Recognition rates of the Gaussian Mixture Classifier (including novel attacks)

Pre-learned	Testing Item	Container	Rare Book	Painting
ˇ	Convolution Filtering * 2	2(2)	2(2)	2(2)
ˇ	JPEG * 14	14(14)	14(14)	14(14)
ˇ	Median Filtering * 4	4(4)	4(4)	4(4)
ˇ	Noise * 12	12(10)	12(9)	12(8)
	Self-Similarities * 3	3(3)	3(3)	3(3)
	PSNR * 10	10(10)	10(10)	10(10)
ˇ	Scaling * 10	10(10)	10(10)	10(10)
ˇ	Cropping * 13	9(1)	8(2)	11(0)
ˇ	Rotation * 18	17(0)	16(0)	18(0)
	Affine * 8	7(7)	8(6)	7(6)
	Removing Lines * 10	10(8)	10(8)	10(8)
	RRS * 10	9(1)	9(1)	9(0)
	RC * 10	9(1)	9(1)	9(0)
Recognition Rate (DCT ordinal measures)		(71+70+65) / (124*3) = 55.38%		
Recognition Rate (Gaussian Mixture Classifier with EFS)		(116+115+119) / (124*3) = 94.07%		

Note: The first column indicates whether the type of image attack was pre-learned, while the second column shows the attack model and how many times it was applied. For example, “Noise *k” means that the noise attack was applied to the image k times. In the remaining columns, “m(n)” indicates that the number of image replicas successfully detected by our Gaussian mixture method and by the pure DCT ordinal measures method was m and n respectively.

We also applied some *novel attacks* (i.e., attacks not modeled in the training phase) to examine the performance of our approach. The results show that the images’ resistance to geometric attacks (cropping, rotation, scaling) was significantly enhanced by our approach; on the average, more than half of the manipulated geometric images were correctly identified in the experiment. In Table 2, the first column indicates whether the image attacks were pre-learned. Clearly, for those novel attacks we did not model in advance, our approach still achieves an acceptable performance and outperforms the pure DCT ordinal measure method.

Performance of Video Detection

We experimented with approximately 106,333 frames of video data from the NDAP’s digital video library of social culture in Taiwan. The video format is MPEG-1 NTSC, for which the resolution is 352×240 and frame rate is 29.97 fps. To test the performance of the proposed approach, the video clips were modified in different ways to generate eight copies for brightness, histogram equalization, changing the resolution to 176×120, changing the frame rate to 15 fps and 10 fps, slow motion (0.5×), fast motion (2×), and hybrid modification (changing to 176×120 resolution, 10 fps, and 2× fast motion). We randomly selected 100 video clips (100×1000 frames in total) as query clips for each type of copy. Hence there are 800 queries in the experiment to verify the performance of our video copy detection module.

Table 3. The F-measure of brightness, equalization, and frame size changing (spatial variations), and frame rate changing, slow and fast motion (temporal variations) copy in Hua's, Kim's and our proposed approach

	Bright	Equ.	176×120	10fps	15fps	0.5×	2×	Hybrid
Hua	89.98	94.87	90.13	94.25	96.01	53.27	75.94	65.52
Kim	93.61	95.89	93.14	76.54	85.90	25.86	43.30	40.27
Our	94.26	96.19	94.24	93.87	95.60	83.55	94.06	83.38

We compared the results of the proposed approach with Hua's and Kim's approaches using F-measure. Table 3 shows the F-measure of all cases, and our approach greatly outperformed the other two. According to the experiment results, we see that our method performs slightly better than Hua's and Kim's for spatial variation attacks such as brightness, equalization and frame size change. For frame rate changes, our method performs better than Kim's but slightly worse than Hua's. However, our method achieves a much better performance (average F-measure is 88.81) for the attacks of fast and slow motion than those of the others (average F-measure is 64.61 and 34.58). To conclude, our method achieves better overall performance for the hybrid case, and is effective not only for spatial-variation but also for temporal-variation attacks.

Discussion on Wrapper-based DRE

Our proposed wrapper-based DRE has three major advantages. First, the wrapper-based approach can be a standalone system, or it can be integrated into existing content players, and even commercial DRM systems. Second, the difficult problem of enforcing usage rules when playing digital content is addressed by DC Wrapper. It monitors the behavior of the content player, and prevents illegal access to digital content. Third, the proposed architecture enables two or more intellectual property rights protection systems to cooperate and complement with each other.

Since the DC Wrapper is implemented using a binary interceptor approach, it depends on the underlying operating system. This is a trade-off between better control over content access and platform independence when considering the integration with various existing DRM systems. Intercepting the message in OS-level is more robust and compatible to various applications than in the non-standardized application level. Just like any other system security tools, there is no 100% secure DRM system which can always survive all kinds of attacks.

For example, a malicious media player could possibly bypass all the usage rules. However, developing such a malicious player is so complicated and time-consuming that it is not technically feasible for an average user. It is also what we try to do to raise the barrier for the abuse of digital content.

CONCLUSIONS

Protecting digital content presents serious technical challenges that the existing approaches have not overcome. The distribution of digital content requires content protection and rights management in order to ensure trust between the parties involved. Trusted computing platforms and the integration of DRM components into the digital libraries would probably encourage content providers to release precious digital assets. In this paper, the proposed integrated framework provides a solution for digital content protection of digital libraries. First, with the wrapper-based DRE technique, a digital rights enforcement environment can be built to maintain the usage rules of digital content that provides stronger protection for digital images, and thereby drastically reduces the piracy of digital content. Second, with the help of content tracking mechanism, pirated digital content altered from original images and videos can be effectively identified. Also, the introduced copy detection techniques have been demonstrated to be more accurate than conventional approaches. By employing such a complementary design, the abuse of valuable digital content can be greatly

reduced, and further discourage the copyright infringements.

NOTE

[1] http://www.ndap.org.tw/

[2] http://www.xrml.org/

[3] http://odrl.net/

REFERENCES

Berrani, S.-A., Amsaleg, L., and Gros, P. (2003). Robust Content-based Image Searches for Copyright Protection, *Proceedings of the 1st ACM International Workshop on Multimedia Databases*, 70-77.

Bhat, D.-N. and Nayar, S.-K. (1998). Ordinal Measures for Image Correspondence, *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20 (4)*, 415-423.

Chang, E.-Y., Wang, J.-Z., Li, C., and Wiederhold, G. (1998). RIME: a Replicated Image Detector for the World-Wide Web, *Proceedings of the SPIE Multimedia Storage and Archiving Systems*, 58-67.

Duhl, J. and Kevorkian, S. (2001). Understanding DRM Systems, *An IDC Research White Paper*, IDC.

Figueiredo, M. A. F. and Jain, A.-K. (2002). Unsupervised Learning of Finite Mixture Models, *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (3)*, 381–396.

Guth, S. (2003). A Sample DRM System, *Proceedings of the Digital Rights Management Conference*, 150-161.

Hampapur, A., Hyun, K.-H., and Bolle, R. M. (2002). Comparison of Sequence Matching Techniques for Video Copy Detection, *Proceedings of the SPIE Conference on Storage and Retrieval for Media Databases*, 194-201.

Hsiao, J.-H., Chen, C.-S., Chien, L.-F., and Chen, M.-S. (2006). Image Copy Detection via Grouping in Feature Space based on Virtual Prior Attacks, *Proceeding of the International Conference on Image Processing*, 1989-1992.

Hsiao, J.-H., Wang, J.-H., Chen, M.-S., Chen, C.-S. and Chien, L.-F. (2005). Constructing a Wrapper-based DRM System for Digital Content Protection in Digital Libraries, *Proceedings of the 8th International Conference on Asian Digital Libraries*, 375-379.

Hua, X. S., Chen, X., and Zhang, H. J. (2004). Robust Video Signature based on Ordinal Measure, *Proceedings of the International Conference on Image Processing*, 685-688.

Hunt, G. and Brubacher, D. (1999). Detours: Binary Interception of Win32 Functions, *Proceedings of the 3rd USENIX Windows NT Symposium*, 135-143.

Jamkhedkar, P. A. and Heileman, G. L. (2004). DRM as a Layered System, *Proceedings of the 4th ACM Workshop on Digital Rights Management*, 11-21.

Kim, C. (2003). Content-based Image Copy Detection, *Signal Processing: Image Communication, 18*, 169-184.

Kim, C. and Vasudev, B. (2005) Spatiotemporal Sequence Matching for Efficient Video Copy Detection, *IEEE Transactions on Circuits and Systems for Video Technology, 15(1)*, 127-132.

Lin, T.-B. and Huang, S.-K. (2004) OpenDReaMS: A Generic DRM Wrapper for COTS Readers, *Proceedings of the 3rd Workshop on Digital Archives Technologies*, 289-295.

Lu, C.-S., Hsu, C.-Y., Sun, S.-W., and Chang, P.-C. (2004). Robust Mesh-based Hashing for Copy Detection and Tracing of Images, *Proceedings of IEEE International Conference on Multimedia and Expo, 1*, 731-734.

Nicolakis, T., Pizano, C. E., Prumo, B. and Webb, M. (2003). Protecting Digital Archives at the Greek Orthodox Archdiocese of America, *Proceedings of the 2003 ACM Workshop on Digital Rights Management*, 13-26.

Petitcolas, F. (2000). Watermarking Schemes Evaluation, *IEEE Signal Processing Magazine, 17 (5)*, 58-64.

Popescu, B. C., Crispo, B., Kamperman, F. L. A. J., Tanenbaum, A. S. (2004). A DRM Security Architecture for Home Networks, *Proceedings of*

the 4th ACM Workshop on Digital Rights Management, 1-10.

Yan, K., Sukthankar, R., and Huston, L. (2004). Efficient Near-Duplicate Detection and Subimage Retrieval, *Proceedings of the 12th Annual ACM International Conference on Multimedia, 13*, 869-876.

Yuan, J., Tian, Q., and Ranganath, S. (2004). Fast and Robust Search Method for Short Video Clips from Large Video Collection, *Proceedings of the International Conference on Pattern Recognition*, 866-869.