Watermarking

Arun Kejariwal

Protecting one's work from being hijacked

The recent growth of Internet usage and networked multimedia systems has necessitated the need for the protection of digital media. This is especially critical for the protection and enforcement of intellectual property rights. Copyright protection involves the authentication of object (text/image/video) ownership, and the identification of illegal copies of a (possibly forged/fake) object. Techniques are needed to prevent the copying, forgery and unauthorized distribution of images and video. In the absence of above, placing images or video sequences on a public network puts them at risk of theft and alteration.

The need for watermarking emanates from the following: A designer has created an image and wants to make it available on the network. When unauthorized copies or forgeries of the image appear elsewhere on the network, the designer needs to prove his ownership of the image. One also needs to determine if and by how much the image has been changed from the original. This way the person can prove ownership by illustrating the difference between the forged image and the original.

Overview

Digital watermarking provides protection of intellectual property in the digital world. Just as plagiarism runs rampant in the physical world, unauthorized copying of datawhether it be audio, visual, or videoexists in the digital world. Digital watermarking attempts to copyright the digital data that is freely available on the World Wide Web to protect the owner's rights. As opposed to traditional, printed watermarks, *digital watermarks* are transparent signatures. They are integrated within digital files as noise, or random information that already exists in the file. Thus, the detection and removal of the watermark becomes more difficult. Typically, watermarks are dispersed throughout the entire digital file such that the manipulation of one portion of the file does not alter the underlying watermark.

History

Wwatermarking stems from the study of "Steganography". The word comes from the old Greek language and can be translated as "cover writing". Steganography was basically a way of transmitting hidden (secret) messages between allies and was used as early as 1000 BCE. First references to steganography appear in Homer's "Iliad" and "Histories of Herodotus" (440 BCE).

Characteristics

A watermark is designed to permanently reside in the host data. When the ownership of data is in question, the information can be extracted to completely characterize the owner. To achieve maximum protection of intellectual property with watermarked media, several requirements must be satisfied:

Imperceptible: The watermark should be imperceptible so as not to affect the viewing experience of the image or the quality of the audio signal.

Undeletable: The watermark must be difficult or even impossible to remove by a malicious cracker, at least without obviously degrading the host signal.

Statistically undetectable: A pirate should not be able to detect the watermark by comparing several watermarked signals belonging to the same author.

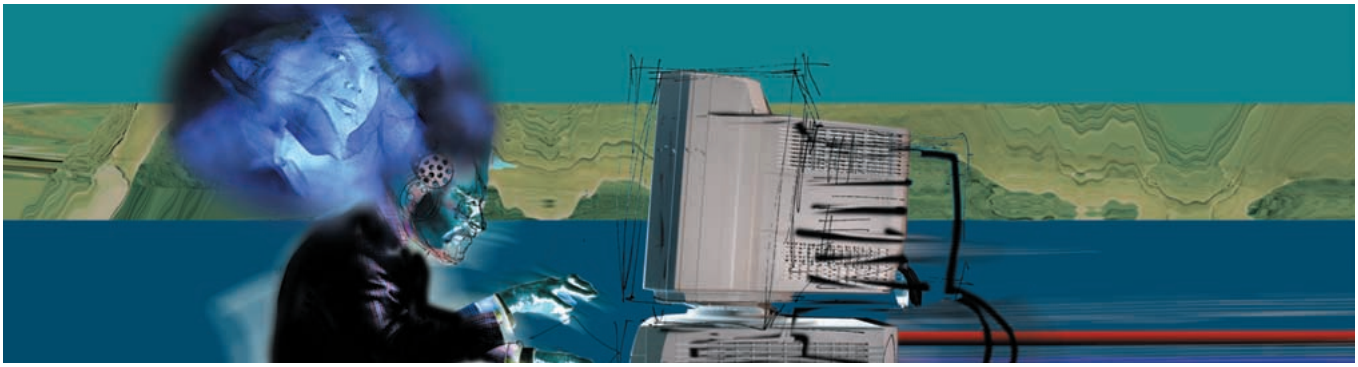
Robustness: The watermark should be able to survive lossy compression techniques like JPEG, which is commonly used for transmission and storage. The watermark should be retrievable even if common signal processing operations are applied, such as signal enhancement, geometric image operations and noise filtering.

Unambiguous: Retrieval of the watermark should unambiguously identify the owner, and the accuracy of identification should degrade gradually in the face of attacks.

Types of watermarks

Visible: Visible watermarks are designed to be easily perceived by the viewer, and clearly identify the owner. The watermark must not detract from the image content itself, however. Most research currently focuses on invisible watermarks, which are imperceptible under normal viewing conditions.

Fragile: Fragile watermarks are designed to be distorted, or to be broken, under the slightest changes to the image. Semi-fragile watermarks are designed to break under all changes



0278-6648/03/$17.00 © 2003 IEEE

that exceed a user-specified threshold.

Spatial: These are constructed in the image spatial domain, and embedded directly into an image's pixel data.

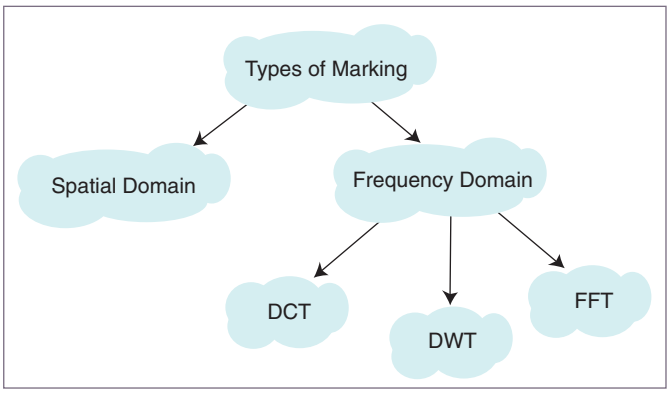


Fig. 1 Types of watermarking

Spectral (or transform-based) watermarks are incorporated into an image's transform coefficients (Discrete-Cosine Transform (DCT) wavelet).

Image-adaptive: Image-adaptive watermarks are usually transform-based and very robust. They locally adapt the strength of the watermark to the image content through perceptual models for human vision. These models were originally developed for image compression.

Blind: Blind watermarking techniques can perform verification of the mark without use of the original image. Other techniques rely on the original to detect the watermark. Many applications require blind schemes; these techniques are often less robust than non-blind algorithms.

Public & private watermarking: In public watermarking, users of the content are authorized to detect the watermark while in private watermarking the users are not authorized to detect the watermark.

Asymmetric & symmetric watermarking: Asymmetric watermarking (also called asymmetric key watermarking) is a technique where different keys are used for embedding and detecting the watermark. In symmetric watermarking (or symmetric key watermarking), the same keys are used for embedding and detecting watermarks.

Steganographic &non-steganographic watermarking: Steganographic watermarking is a technique where content users are unaware that a watermark is present. In non-steganographic watermarking, the users are aware of the presence of a watermark. Steganographic watermarking is used in fingerprinting applications while non-steganographic watermarking techniques can be used to deter piracy.

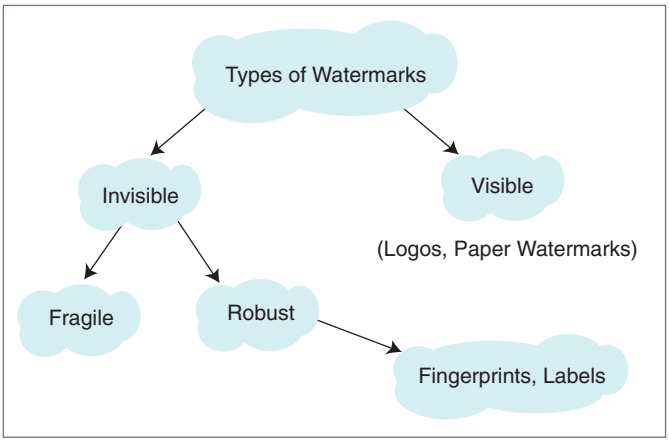


Fig. 2 Types of watermarks

Image watermarking

Human eyes are more sensitive to noise in the lower frequency range than in its higher frequency range counterpart. (The energy of most natural images is concentrated on the lower frequency range.) Therefore, the quantization table applied in lossy compression always reflects the human visual system that is less sensitive to quantization noise at higher frequencies. In order to invisibly embed the watermark and survive the lossy data compression, a reasonable trade-off is to embed the watermark into the middle-frequency range of the image.

To prevent an expert from extracting the hidden information directly from the transformed domain, the watermarks are embedded by modifying the relationship of the neighboring blocks of middle-frequency coefficients of the original image. This is done instead of embedding by an adaptive operation. "Adaptive" watermarks may be detected by using adaptive filters - as the watermark is based on the original image characteristics only.The former is better as the coefficient governing relationship is known to the creator only.

The oldest technique of embedding data into images is the Least Significant Bit (LSB), and it is implicitly based on masking. What LSB does initially is to set the least significant bit of each pixel to 0. This method satisfies the perceptual transparency property, since only the least significant bit of an 8-bit value is altered. Data can be embedded into the image by choosing the desired values of 0's and 1's for the LSBs. This method was initially designed to work for gray scale images. But it can be easily extended to color images by treating each plane as the single plane in the former. This technique is useful for detecting size modifications or when some editing may have been done to the image.

DCT-based watermarking

In this technique, the original image is divided into 8x8 blocks of pixels. Then, the 2-D DCT is applied independently to each block. After that, coefficients of the middle-frequency range are picked from the DCT coefficients.

An example of defining the middle-frequency is shown is Fig. 3. A 2-D sub-block mask is used to compute the residual pattern from the chosen middle-frequency coefficients. For example, if a = b = c = 0, d = _1; x = 1, then the polarity is a binary pattern. This pattern represents the coefficients at the position of the current block that is larger (polarity = 1) or less (polarity = 0) than the coefficient at the corresponding position of the previous block.

Let the digital watermark be a binary image. A fast 2-D pseudo random number traversing method is used to permute the watermark so as to disperse its spatial relationship. In addition to this pixel-based permutation, a block-

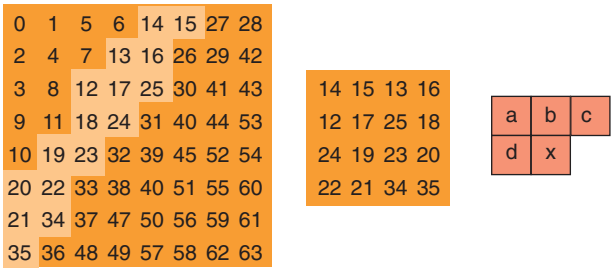


Fig. 3 An example of middle-frequency components and a residual block

based permutation according to the variances of both the image and the watermark is used. Although the watermark is embedded into the middle frequency coefficients, for those blocks with little variances (i.e. the blocks containing the low frequency contents), the modification of DCT coefficients will introduce quite visible artifacts. To improve the invisibility in this image-dependent permutation, both variances of image blocks and watermark blocks are sorted and mapped accordingly.

After the residual pattern is obtained-for each marked pixel of the permuted watermarkthe DCT coefficients are modified according to the residual mask. This way the corresponding polarity of the residual value is reversed. Finally, we inverse the DCT of the associated result to obtain the watermarked image. Figure 4 shows the embedding steps of intraframe watermarking.

Now the extraction of the watermark requires the original image, the watermarked image and the digital watermark. Why? First of all, both the original image and the watermarked images are DCT transformed since we made use of the chosen middle-frequency coefficients and the residual mask to obtain the residual values. Thus, the exclusive-or (XOR) operation would need to be performed on these two residual patterns to obtain a permuted binary signal. Then the person would need to reverse both the block- and the pixel-based permutations to get the extracted watermark. Figure 4 shows the extracting steps of intraframe watermarking.

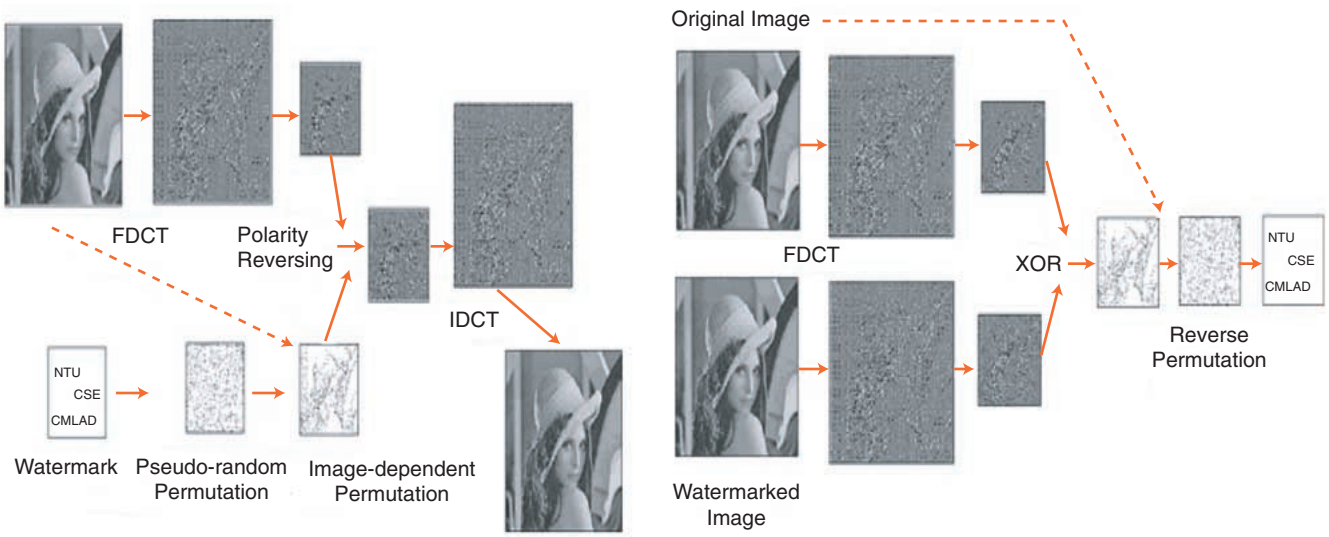


Fig. 4 Embedding and extracting a watermark

Other applications

Audio watermarking. In this case, time and frequency masking properties of the human ear are used to conceal the watermark and make it inaudible. The greatest difficulty lies in synchronizing the watermark and the audio file. But techniques like the introduction of new audio formats that overcome this problem have been proposed.

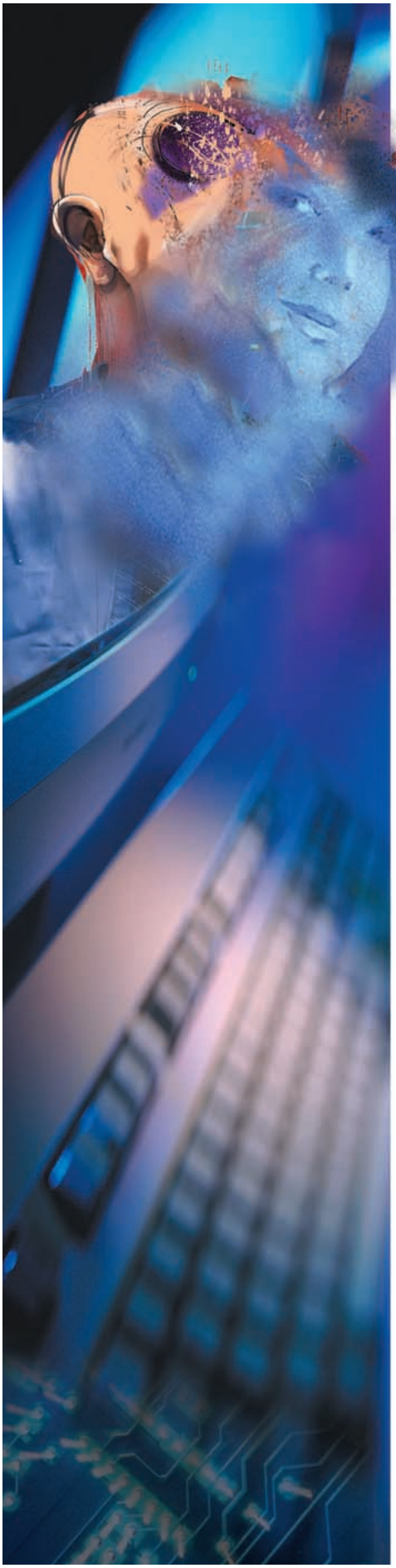
Text watermarking. This problem, which was one of the first to be studied in the information hiding area, can be solved at two levels. At the printout level, information can be encoded in the way the text lines or words are separated. (This action facilitates the survival of the watermark even if photocopied.) At the semantic level (necessary when raw text files are provided), equivalences between words or expressions can be used. However, special care has to be taken not to destroy the possible intention of the author.

Fingerprinting. This is similar to the previous application and allows acquisition devices (such as video cameras, audio recorders, etc) to insert information about the specific device (e.g., an ID number) and date of creation. This can also be done with conventional digital signature techniques. But with watermarking, it becomes considerably more difficult to excise or alter the signature. Some digital cameras already include this feature.

Hardware/software watermarking. This presents a good paradigm to understand how almost every kind of data can be copyright protected. If one is able to find two different ways of expressing the same information, then one bit of information can be concealed, something that can be easily generalized to any number of bits. This is why it is generally said that a perfect compression scheme does not leave room for watermarking. In the hardware context, Boolean equivalences can be exploited to yield instances that use different types of gates and that can be addressed by the hidden information bits. Software can be also protected not only by finding equivalences between instructions, variable names, or memory addresses, but also by altering the order of non-critical instructions without changing the semantics of the program. All this can be accomplished at the compiler level.

Types of watermark attacks

Simple attacks (other possible names include waveform attacks or noise attacks") are conceptually simple. They attempt to impair the embedded watermark by manipulating the whole watermarked data (host data plus watermark), without trying to identify and isolate the watermark. Examples include linear and general non-linear filtering, waveform-based compression (JPEG, MPEG), addition of noise, addition of a cropping, quantization in the pixel domain, conversion to analog, and correction.



Detection-disabling attacks (other names include "synchronization attacks) try to break the correlation and make the recovery of the watermark infeasible for a watermark detector. This is done mostly by geometric distortion like zooming, shift in spatial or temporal (for video) direction, rotation, shear, cropping, pixel permutations, subsampling, removal or insertion of pixels or pixel clusters, or any other geometric transformation of the data. A typical property of this type of attacks is that the watermark remains in the attacked data. Typically, it can be recovered with increased intelligence (and thus, complexity) of the watermark decoder.

Ambiguity attacks (other possible names include "confusion attacks, deadlock attacks, inversion attacks, fake-watermark attacks and fake-original attacks) attempt to confuse by producing fake data. In an ambiguity attack, the attacker tries to fake a watermark and an object such that the watermark is embedded in the alleged "original" object.

Removal attacks attempt to: a) analyze the watermarked data, b) estimate the watermark or the host data, c) separate the watermarked data into host data and watermark, and d) discard only the watermark. Examples are collusion attacks, denoising, certain nonlinear filter operations and compression attacks using synthetic modeling of the image (e.g. using texture models or 3-D models).

Common attack techniques

Additive noise. This may stem in certain applications from the use of digital to analog (D/A) and analog to digital (A/D) converters or from transmission errors. However, an attacker may introduce perceptually shaped noise (thus, imperceptible) with the maximum unnoticeable power. This action will typically increase the threshold at which the correlation detector works.

Cropping. This is a very common attack since in many cases the attacker is interested in a small portion of the watermarked object, such as parts of a certain picture or frames of a video sequence. With this in mind, in order to survive, the watermark needs to be spread over the dimensions where this attack takes place.

Rotation and scaling. This has been the true battle horse of digital watermarking, especially because of its success with still images. Correlation-based detection and extraction fail when rotation or scaling are performed on the watermarked image because the embedded watermark and the locally generated version do not share the same spatial pattern anymore. An exhaustive search with different rotation angles and scaling factors does yield the correlation peak, but it is prohibitively complex.

Conclusion

Digital watermarking is an effective technique for protecting intellectual property (IP) rights by embedding information in digital multimedia data. It bears a huge commercial potential as it is widely deployed in consumer electronic devices. Digital watermark technology can be used in consumer electronic devices such as digital still cameras, digital video cameras, set top boxes (STB), digital versatile disc (DVD) players and MPEG-1 Audio Layer-3 (MP3) players. As a result, it can protect information in controlled access (pay-per-view broadcasts), prevent illegal replication and embedding ownership information in images captured in digital still/video cameras.

Read more about it

- A. B. Kahng et. al, "Watermarking Techniques for Intellectual Property Protection" Design Automation Conference, pages = 776-781, 1998
- R. ChandraMouli et. al, "A Distributed Framework for Steganalysis" ACM Multimedia Workshops 2000 : 123-126

F. Hartung and M. Kutter, "Multimedia Watermarking Techniques" *Proceedings of the IEEE,* vol. 87, no. 7, pp. 1079-1107, July 1999.

About the author

Arun Kejariwal is a second year graduate student at Center for Embedded Computing Systems, University of California, Irvine. He received his B. Tech. degree in Electrical Engineering from Indian Institute of Technology, New Delhi, India. His interests are compilers and embedded systems. He is a Student member of IEEE