How to achieve robustness against scaling in a real-time digital watermarking system for broadcast monitoring

HOW TO ACHIEVE ROBUSTNESS AGAINST SCALING IN A REAL-TIME DIGITAL WATERMARKING SYSTEM FOR BROASCAST MONITORING

P. Termont, L. De Strycker, J. Vandewege,
M. Op de Beeck, J. Haitsma, T. Kalker, M. Maes, G. Depovere

University of Gent, INTEC, Belgium
Philips Research Laboratories Eindhoven, The Netherlands

ABSTRACT

In the European Esprit project VIVA (Visual Identity Verification Auditor) a real-time digital watermarking system for broadcast monitoring has been investigated and implemented. On top of the usual requirements for watermarks, the VIVA watermarking system has to satisfy an additional number of constraints. One of the most important constraints in a broadcast environment is the robustness of the watermark against scaling. This paper describes how robustness against scaling is achieved in the VIVA project. Furthermore, a real-time implementation of the algorithms is discussed. Experimental results prove the effectiveness of the algorithms.

1. INTRODUCTION

Using digital watermarking a mechanism of IPR (Intellectual Property Right) protection can be provided. In the European Esprit project VIVA the feasibility of a professional television broadcast surveillance system using digital watermarking has been proven. A real-time watermark embedder and detector are implemented in software and the experimental results prove their possible exploitation in a professional environment. More information on the VIVA watermarking system for broadcast monitoring can be found in [1], [2] and [3].

The VIVA watermarking technology for copy protection is referred to as JAWS (Just Another Watermarking System). Using the JAWS watermarking algorithms, the embedded watermark fulfils its basic requirements. It is unobtrusive (i.e. there is no perceptual degradation of the quality), easily detectable by dedicated software and hardware and very difficult to remove by malicious and deliberate attacks or by common processing. Furthermore, the JAWS watermarking technology is optimized for the high picture quality needed in a broadcast environment, makes it possible to embed a 36-bit payload every second and is designed such that the false alarm probability is very low as well. Finally, the JAWS watermarking algorithms ensure that the watermark is robust against all processing steps it will be subject to in the broadcast chain, such as digital to analogue conversion, editing, compression and scaling. The excellent performance of the JAWS watermarking technology is also discussed in [4] and [5].

Making a watermark robust against scaling is not straightforward and can be considered as one of the most difficult issues in the development of a digital watermarking system for broadcast monitoring. However, the VIVA watermarking technology would never be used in a professional broadcast surveillance system, if the watermarking algorithms were not able to retrieve the correct watermark from a scaled watermarked video stream. If the watermark didn't survive scaling, the monitoring system could be easily disrupted and all its possible applications would disappear.

In this paper, the robustness of a watermark against scaling is addressed in detail. The algorithms used in the VIVA project to deal with scaled video streams are explained and their implementation on a state-of-the-art digital signal processor is discussed. Finally, some first experimental results, which prove the effectiveness of the algorithms, are presented.

2. ALGORITHM OVERVIEW

2.1. Watermarking algorithms

During the development of the JAWS technology, the algorithms are improved in order to meet all requirements for broadcast monitoring. The 'first generation' JAWS watermark embedding and detection algorithms are explained in [5] and their implementation is discussed in [6]. In these watermarking algorithms, two important issues remained to be solved before a successful

0-7803-6297-7/00/$10.00 © 2000 IEEE

application of the broadcast monitoring system could be attempted. A first problem was the fact that these watermarking algorithms didn't support robustness against scaling of the video. Secondly, the payload had to be increased from 8 bits per second to 36 bits per second to obtain a sufficient number of different video identifiers. In order to deal with scaling and to provide a higher payload, the 'first generation' algorithms were further improved, leading to the 'second generation' JAWS algorithms. In order to distinguish the upgraded version of JAWS from the 'first generation' algorithms, we will refer to the upgraded version of JAWS as JAWS+. The JAWS+ algorithms and their implementation are addressed in this paper.

In short, the watermarking embedding process of both the JAWS and JAWS+ algorithms casts a watermark on a video sequence by adding a pseudo-noise pattern in the spatial domain. This pseudo-noise pattern is derived from a set of N basic pseudo-noise patterns and a key, which carries the information the user wants to embed into the video. The embedding of the pseudo-noise pattern is done with a certain embedding depth. This embedding depth determines the energy of the watermark. The watermark detection procedure can be described as the computation of a threshold correlation value. The incoming frames are folded and summed. This result is then correlated with each of the N basic pseudo-noise patterns used in the embedding procedure. From the correlation value a decision variable is extracted. If the decision variable is larger than a certain threshold the video is said to be watermarked and the information embedded in the video stream can be extracted. Otherwise it is very likely that no watermark has been embedded.

Only the JAWS+ algorithms resist independent horizontal and vertical scaling. This can be understood as follows. For the 'first generation' JAWS algorithms, the robustness is such that embedded watermarks are not lost when watermarked video is scaled. If the scaling parameters are known, then appropriate rescaling before detection can retrieve the embedded watermark and the associated payload. In a theoretical sense JAWS is therefore scale invariant. However, this can only be exploited if the scaling parameters can somehow be retrieved.

In the JAWS+ detection algorithm an efficient method for scaling parameter retrieval is found by exploiting the inherent symmetry of the embedded watermark. Figure 1 shows that the pseudo-noise pattern, carrying the embedded information, is added to every quarter of the image, creating horizontal and vertical symmetry. For every pixel P in the original watermark, there is a corresponding pixel P' in the horizontal direction and a corresponding pixel P'' in the vertical direction. The distance from pixel P to edge A is the same as the distance between pixel P' and edge B. Analogously, pixel P is as far from edge C as pixel P'' is from edge D. By scaling the watermarked video stream, the position of the corresponding pixels towards the edges changes and the symmetry is disturbed. Figure 1 shows that, if the original size of the image is for example reduced, the distance from pixel P to edge A will increase, whereas pixel P' will come closer to edge B. A similar conclusion can be drawn for pixel P and P'' by looking in the vertical direction. The more the video stream is scaled, the greater this effect will be.

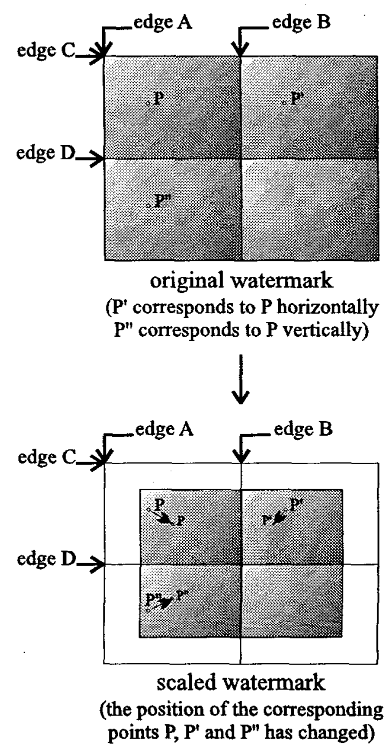

Figure 1 : Scaling of the watermark

The positions of corresponding pixels can now be retrieved by using the same correlation method as for the detection procedure. However, instead of folding the accumulated incoming video frames and correlating the result with the N basic pseudo-noise patterns, the left half of the accumulated video frames is correlated with the right half to find the positions of the corresponding pixels in the horizontal direction and by correlating the upper half with the lower half the positions of the corresponding pixels in the vertical direction can be determined. From these positions, both the horizontal and vertical scaling parameter can be calculated.

This method of scale retrieval works adequately if the basic pseudo-noise patterns are large enough (e.g. ¼ image). If the basic pseudo-noise patterns are too small, the accuracy of the retrieved scaling parameters will not be sufficient and if the incoming video stream is rescaled using these imprecise scaling parameters, no watermark will be detected. The JAWS algorithms use 2 basic pseudo-noise patterns of size 128x128, which makes it possible to carry a payload of 8 bits per second. Because of the size of the basic pseudo-noise patterns, JAWS is not robust against scaling. In the JAWS+ algorithms, 4 basic pseudo-noise patterns with a size of ¼ image are defined, which means that the algorithms are able to support robustness against scaling. By choosing 4 patterns of ¼ image it is also possible to carry a payload of 36 bits per second. This is explained in [4].

2.2. Real-time implementation

In the VIVA system, the multimedia processor TriMedia TM-1000 DSPCPU of Philips Semiconductors is used to implement both the JAWS and JAWS+ real-time watermark embedding and detection algorithms. The TriMedia processor is a powerful DSP with a state-of-the-art VLIW (Very Long Instruction Word) architecture and a speed of 4 BOPS (Billion Operations Per Second). One of TriMedia's key features is the seamless interface to peripherals like video and audio equipment. The implementation of the algorithms is done using a PCI plug-in card with one TriMedia.

The implementation of the JAWS+ watermark embedder is based on the 'first generation' version and straightforward. This is understood from [6].

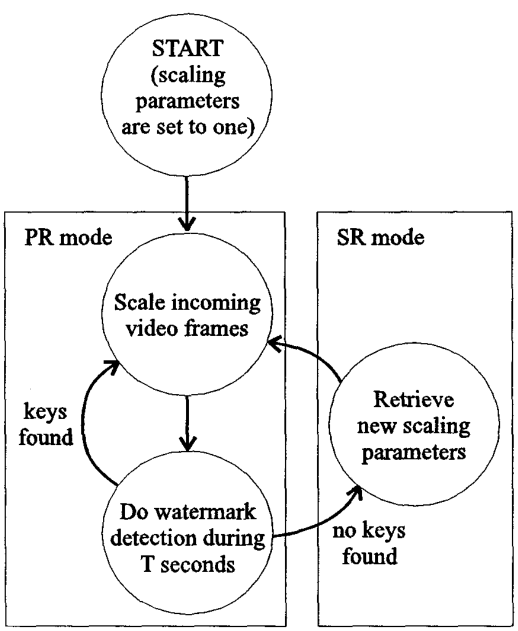

Figure 2 : Two-phase detection proces

The implementation of the algorithm of the JAWS+ watermark detector with robustness against scaling is done in a two-phase approach. Figure 2 illustrates the watermark detection process. The 36-bit watermark detector has two different operational modes, the payload retrieval mode (PR) and the scale retrieval mode (SR). In the PR mode, the video is scaled in real-time in accordance with the scaling parameters in effect and accumulated in a buffer. A watermark detection is then made every 0,5 seconds. If no watermark is found during a certain number (T) of consecutive seconds, the watermark detector switches from the PR mode to the SR mode. In the SR mode, the detector accumulates video and uses the autocorrelation method explained above to retrieve the scaling parameters. Once the scaling parameters are found, the detector goes into PR mode again using the measured scaling parameters. This two-phase approach can be justified by stating that the quality of the content will be severely degraded if scaling parameters change very quickly over time.

3. EXPERIMENTS AND RESULTS

In order to find out if the scaling parameters can be retrieved and if the watermark can be detected in the rescaled video stream, the algorithms have been tested in a laboratory environment with a real-time video stream.

The basic configuration consists of two personal computers, each with a TriMedia PCI plug-in card inside. An incoming video stream is processed on the first PCI card and a watermark is added to each video frame. The watermarked video is then scaled. On the second PC with PCI card, the two-phase detection process is implemented.

The scaling parameters are changed every 12 seconds. The horizontal scaling factor (H) and vertical scaling factor (V) are modified as follows (remark that both scaling factors don't need to have the same value – horizontal and vertical scaling can be independent) :

1) 0-12 sec : H = 0.8; V = 0.8
2) 12-24 sec : H = 0.9; V = 0.9
3) 24-36 sec : H = 1 ; V = 1 (no scaling)
4) 36-48 sec : H = 1.1; V = 1.1
5) 48-60 sec : H = 1.2; V = 1.2
6) 60-72 sec : H = 1.2; V = 0.8
7) 72-84 sec : H = 0.8; V = 1.2

Detection starts in the PR mode. The scaling parameters are set to one by default (H=1 ; V=1). This means that it is assumed that the incoming video is not scaled. As the incoming video is scaled, no watermark will be detected at first. The detector therefore starts the SR mode to retrieve possible scaling parameters. Once these new scaling factors are determined, the detector goes back to PR mode, rescales the incoming video and retrieves the necessary information from the watermarked video stream. If the scaling parameters are changed again, the detector won't find a watermark and will start a new search for scaling factors.

Figure 3 gives the retrieved scaling parameters. The accuracy of the scaling parameters is better than 0.5 %. This is sufficient to find the watermark in the rescaled video stream. Also, the accuracy of the horizontal scaling parameter is better than than the accuracy of the vertical scaling parameter. This is due to the fact that the horizontal dimension of a video frame is larger than its vertical dimension.

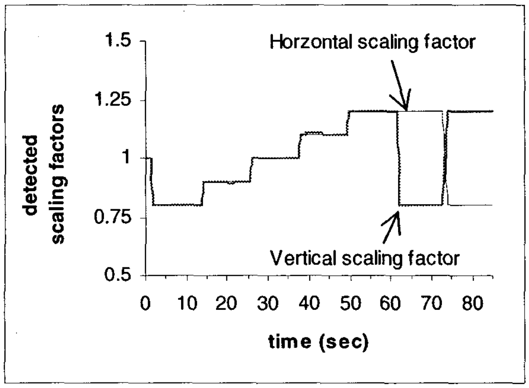

Figure 3 : Detected scaling factors

Figure 4 gives the watermark detection results. As soon as the correct scaling parameters are used to rescale the incoming video stream, the strength of the detected watermark is always larger than the threshold, indicating that a watermark is present. This threshold is chosen such that the probability of false positives is about 10^{-7}. More details on the threshold setting are presented in [7]. Remark that the decision variable of the watermark detection reaches its maximum when the incoming video stream is not scaled (H=1 and V=1 ; 24-36 sec).

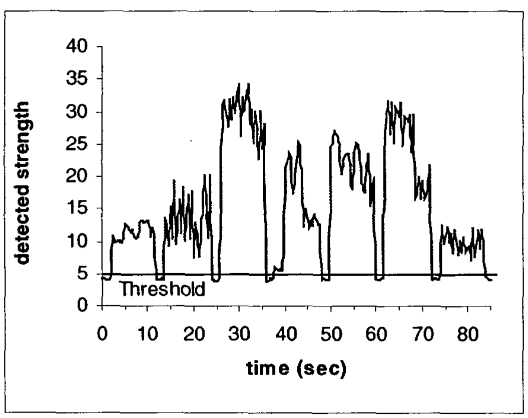

Figure 4 : Strength of the detected watermark

These results prove that the algorithms, which have been developed in the VIVA project, support robustness of the watermark against scaling.

4. CONCLUSIONS

In this paper a real-time digital watermarking system for broadcast monitoring and the robustness of the watermark against scaling has been discussed. Some experimental results prove that a watermark can be retrieved from a scaled video stream in real-time.

Acknowledgements

The authors would like to acknowledge the support through the VIVA project from the Esprit program of the European Commission.

References

[1] VIVA home page :
http://www.intec.rug.ac.be/Research/Groups/hfhsdesign/viva/index.htm

[2] P. Termont, L. De Strycker, J. Vandewege, J. Haitsma, T. Kalker, M. Maes, G. Depovere, A. Langell, C. Alm and P. Norman, "Performance measurements of a real-time digital watermarking system for broadcast monitoring", ICMCS '99, Florence, Italy.

[3] G. Depovere, T. Kalker, J. Haitsma, M. Maes, L. De Strycker, P. Termont, J. Vandewege, A. Langell, C. Alm, P. Norman, G. O'Reilly, B. Howes, H. Vaanholt, R. Hintzen, P. Donnelly and A. Hudson, "The VIVA project : digital watermarking for broadcast monitoring", ICIP '99, Kobe, Japan.

[4] M. Maes, T. Kalker, J. Haitsma and G. Depovere, "Exploiting Shift Invariance to Obtain a High Payload in Digital Image Watermarking", ICMCS '99, Florence, Italy.

[5] T. Kalker, G. Depovere, J. Haitsma and M. Maes, "A Video Watermarking System for Broadcast Monitoring", Proc. of IS&T/SPIE '99, San Jose, California.

[6] L. De Strycker, P. Termont, J. Vandewege, J. Haitsma, T. Kalker, M. Maes, G. Depovere, "An implementation of a real-time digital watermarking process for broadcast monitoring on a TriMedia VLIW processor", IPA'99, Manchester, UK.

[7] T. Kalker, J.P. Linnartz and G. Depovere, "On the Reliability of detecting Electronic Watermarks in Digital Images", Eusipco '98, Island of Rhodes.