VLSI Architectures of Perceptual Based Video Watermarking for Real-Time Copyright Protection

Saraju P. Mohanty, Elias Kougianos,Wei Cai, Manish Ratnani

VLSI Design and CAD Laboratory (VDCL)

University of North Texas, Denton, TX 76203.

Email-ID: saraju.mohanty@unt.edu.

UNT
UNIVERSITY OF NORTH TEXAS
Discover the power of ideas

Summary and Conclusions

- For effective digital rights management (DRM) of multimedia in the framework of embedded systems, we present a watermarking algorithm and VLSI architecture that can insert a broadcaster's logo in video streams in real-time to facilitate copyrighted video broadcasting and internet protocol television (IP-TV).
- The watermarking process is performed in the frequency domain and the system's maximum throughput is 43frames/sec at a clock speed of 100MHz.
- Further research is under way to extend the real time performance of the system to HDTV and higher resolutions and to improve the PSNR. Advanced MPEG-4 features, such as N-bit resolution, advanced scalable textures, and video objects will be utilized.

VLSI Architectures of Perceptual Based Video Watermarking for Real-Time Copyright Protection

Saraju P. Mohanty[1], Elias Kougianos,Wei Cai, Manish Ratnani
University of North Texas, Denton, TX 76203.
Email-ID: saraju.mohanty@unt.edu.

Abstract

- *For effective digital rights management (DRM) of multimedia in the framework of embedded systems, both watermarking and cryptography are necessary.*
- *We present a watermarking algorithm and VLSI architecture that can insert a broadcaster's logo in video streams in real-time to facilitate copyrighted video broadcasting and internet protocol television (IP-TV) which when realized in silicon can be deployed in any multimedia producing appliance to enable DRM.*
- *The watermarking process is performed in the frequency domain and the system's maximum throughput is 43frames/sec at a clock speed of 100MHz which makes it suitable for real-time digital video broadcasting emerging applications such as IP-TV.*

Introduction, Motivation, and Contributions

There is a need for real-time copyright logo insertion in emerging applications, such as internet protocol television (IP-TV). This is demonstrated in Fig. 1. The visible-transparent watermarking unit accepts broadcast uncompressed video and the broadcaster's logo. The output is real-time compressed video with the logo embedded. This situation arises in IP-TV and digital TV broadcasting when video residing in a server has to be broadcast by different stations and under different broadcasting rights.

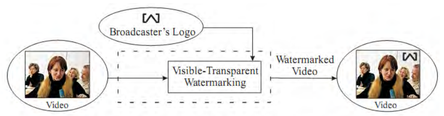


Figure 1. Real-time logo insertion through watermarking.

MPEG-4 is the mainstream exchangeable video format in the Internet today because it has higher and flexible compression rate, lower bit rate, and higher efficiency while providing superior visual quality. Existing works are targeted towards invisible watermarking, not useful for logo insertion. Other existing works are for images not for video.

Microsoft®, Real® and Apple® support the MPEG4 standard and already have embedded MPEG-4 decoders into some of their products, and there are even free software implementations available, such as the Xvid codec. This motivated us to consider MPEG-4 as the target video compression framework in our research.

Novel contributions of this paper are as follows:

- A perceptual-based adaptive visible watermarking algorithm suitable for video broadcasting.
- VLSI architectures for real-time watermarking in the context of compressed video (MPEG-4).
- Simulink and FPGA prototyping of the VLSI architectures which can be integrated in multimedia producing appliances (e.g. digital camera, network processor).

The Proposed Watermarking Algorithm

Algorithm 1 The proposed MPEG-4 watermarking algorithm

1: Convert RGB color frames to YC_bC_r frames for the input video.
2: Resample YC_bC_r frames according to 4:2:0 sampling rate.
3: Split Y frame and watermark image into 8×8 blocks.
4: Perform DCT for each 8×8 block to generate DCT coefficients.
5: Perform perceptual analysis of the host video frame.
6: Compute scaling and embedding factor for different blocks.
7: Each block of Y DCT matrix is watermarked with an 8×8 watermark DCT matrix at same location as at DCT domain.
8: Perform 2-D IDCT for each 8×8 watermarked matrix to transform it back to Y color pixels.
9: Buffer watermarked Y component, non watermark C_b and C_r frames which holds a GOP.
10: Split Y component into 16×16 blocks and C_b and C_r into 8×8.
11: Perform motion estimation for Y component.
12: Obtain the motion vectors (MV) and prediction errors of residual frame for motion compensation (MC) for Y component.
13: Obtain motion vector and prediction error for C_b, C_r components and residual frame for motion compensation.
14: Perform 2-D DCT on blocks of different frames.
15: Quantize 2-D DCT coefficient matrix.
16: Zigzag scan quantized 2-D DCT coefficient matrix.
17: Entropy coding re-ordered 2-D DCT coefficient matrix and motion vector.
18: Build structured compreseed stream from the buffer.

We now present a watermarking algorithm that performs the broadcaster's logo insertion as watermark in the DCT domain. The proposed watermarking algorithm is presented as a flow-chart in Fig. 2.

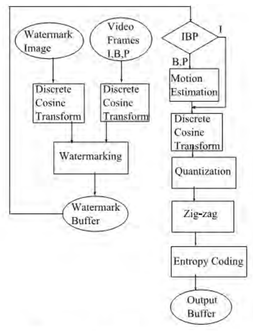


Figure 2. The flow of the proposed video watermarking algorithm.

The Proposed VLSI Architectures

The proposed data path architecture that can perform watermarking in the MPEG-4 video compression framework is shown in Fig. 3.

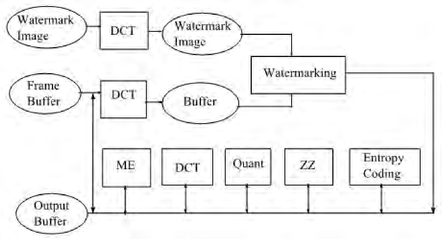
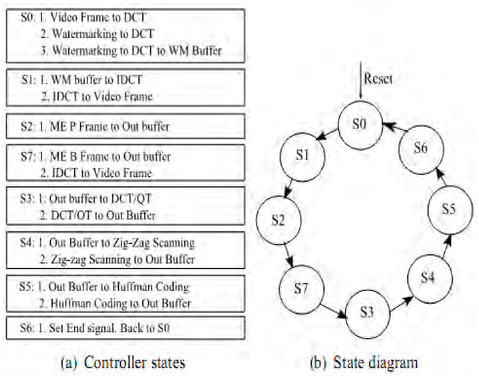


Figure 3. Datapath for MPEG-4 watermarking architecture (bus width 12 bits).

The controller generates address and control signals to synchronize other components. It is realized as a finite state machine (FSM) as shown in Fig. 4. In this FSM several sub-states have been merged to 8 states for simplicity of design.

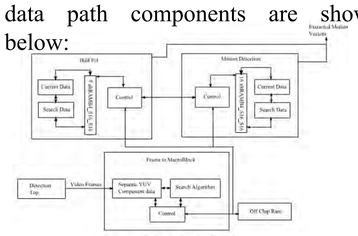


(a) Controller states (b) State diagram

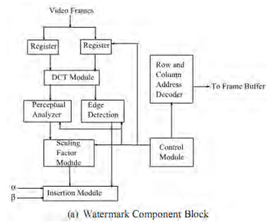
Figure 4. State diagram of the controller for MPEG-4 watermarking architecture.

System Level Modeling and Prototyping

The proposed architectures of some data path components are shown below:



(a) Motion Estimation Module



(a) Watermark Component Block

Table 1. FPGA synthesis report of the VLSI architectures

Components	Elements	Registers	MUXes
2D DCT	1477	157	0
Quantization	2363	0	1
Zigzag	1030	786	0
Watermark	24	0	0
Frame Buffers	7713	6156	0
Motion Vector Buffer	667	520	0
Watermark Buffer	4043	3048	0
RGB to YCbCr	1416	0	8
Motion Estimation	8987	4900	0
Controller	575	157	0
Overall Architecture	28322	16532	9

Testing of Watermarking Quality

$$MSE = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{k=1}^{3}|p(m,n,k) - q(m,n,k)|^2}{3 \times M \times N}$$

$$PSNR = 10 \times \log_{10}\left(\frac{(2^i-1)^2}{\text{MSE}}\right).$$

Table 2. Video quality metrics for watermark - 1

Clips	$PSNR(dB)$	$RMSE$	Compression Ratio		
			Average	Range	Estimate
Bird	21.1	22.3	21.4	16 to 39	16
Dinner	22.2	19.7			
Iphone	22.2	20.1			
LGphone	22.8	23.1			

Table 3. Video quality metrics for watermark - 2

Clips	$PSNR(dB)$	$RMSE$	Compression Ratio		
			Average	Range	Estimate
Bird	21.7	20.9	21.2	16 to 39	16
Dinner	21.9	20.4			
Iphone	21.9	21.0			
LGphone	21.4	22.2			

We performed exhaustive simulations to assess watermarking quality with a large variety of watermark images and video clips. For brevity we present selected examples of watermarked video in Figs. 7 and 8.

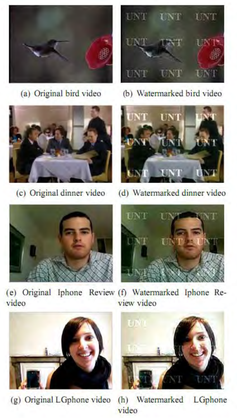


Figure 7. Sample watermarked videos using watermark - 1.

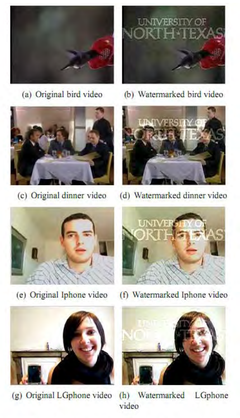


Figure 8. Sample watermarked videos using watermark - 2.

We present performance statistics with reference to existing hardware based watermarking for video in Table 4. We note that our implementation is the only one capable of achieving real-time video watermarking and compression at rates exceeding existing broadcast standards.

Table 4. Comparative perspective with other video watermarking hardware

Works	Design Type	Watermarking	Domain	Features
Maes, [9]	FPGA/IC	Invisible-Robust	Spatial	$17k$ Gates
Mathai [10]	Custom IC	Invisible-Robust	Wavelet	1.8V
Tsai [16]	Custom IC	Invisible-Robust	Spatial	NA
This Work	FPGA	Visible	DCT	$43 frames/sec$

Summary, Conclusions, and Future Works

- We presented a visible watermarking algorithm using FPGA technology for MPEG4 video. The algorithm and its inherent are suitable for real-time applications such as video broadcasting, IP-TV and digital cinema.
- Further research is under way to extend the real time performance of the system to HDTV and higher resolutions and to improve the PSNR.
- In future, advanced MPEG-4 features, such as N-bit resolution, advanced scalable textures, and video objects will be utilized.

VLSI Design and CAD Laboratory (VDCL)

UNIVERSITY OF NORTH TEXAS™