Video Authentication for H.264/AVC using Digital Signature Standard and Secure Hash Algorithm

Nandakishore Ramaswamy
Qualcomm Inc
5775 Morehouse Dr,
Sam Diego, CA 92122. USA
nandakishore@qualcomm.com

K. R. Rao
The University of Texas at Arlington
416 Yates Street, Nedderman Hall,
Rm 530
Arlington, TX 76010, USA
rao@uta.edu

ABSTRACT
Multimedia authentication techniques are required to prove the validity of legal multimedia content and establish the identity of the content creator. Digital signatures are one way of authenticating multimedia content. Typically digital signatures use cryptographic techniques to ensure the integrity of the multimedia bitstream. This paper proposes a hard video authentication and sender verification scheme for video sequences compressed using H.264/AVC Main Profile by using digital signatures and cryptographic hash. Features from the transform domain are used as the authentication data for a macroblock. The algorithm can also detect the cause of authentication failure and point out the location of the tampered frames in case of frame tampering. Results demonstrate that the proposed scheme is highly robust to temporal and spatial manipulations.

Categories and Subject Descriptors
E.4 [**Coding and Information Theory**]: Nonsecret encoding schemes

I.4 [**Image Processing and Computer Vision**]: Compression

General Terms
Algorithms, Security.

Keywords
H.264/AVC, Secure Hash, Multimedia Security, Digital Signature Standard, Video authentication.

1. INTRODUCTION
Recent advances in developing powerful processors and increasing software sophistication have made it easier to alter and forge digital video content without leaving any trace [1]. Authentication methods need to be established for verifying the integrity and creator of the digitized video. Two major ways of authentication exist for digital multimedia content [2]. One is a soft authentication scheme which allows modification of the multimedia content as long as it does not degrade perceptual quality. The other is a hard authentication scheme which does not allow any modification to the video bitstream and can be considered as a form of lossless authentication. Digital signatures are one way of achieving hard (lossless) authentication.

H.264/AVC is the latest video coding standard developed by the Joint Video Team (JVT) consisting of experts from ITU-T and ISO/IEC [3]. This standard has several new features such as increased compression efficiency, error resiliency and support for a wide range of applications which make it suitable for adoption by the industry. Fidelity range extensions composed of four high profiles further improve the compression efficiency based on additional functionalities [12, 13]. JVT documents can be accessed from [14].

In this proposed algorithm, the Secure Hash Algorithm (SHA) [4] and Digital Signature Standard (DSS) [5] have been used to validate the video and verify the sender. This algorithm also reduces computational complexity by authenticating features extracted from a macroblock rather than authenticating the whole macroblock. Since the human visual system is more sensitive to luminance frames than chrominance frames [6] and a change in the spatial domain results in a change in the transform domain; only the luminance frames in transform domain are considered for authentication of the video.

1.1 H.264/AVC transform structure
In the H.264/AVC standard [8], a 4x4 integer approximation of the DCT is used to remove the correlation from the spatial domain. This transform is used by INTRA 4x4 and INTER 4x4 prediction modes.

For the INTRA 16x16 mode, which is used in predicting the entire 16x16 macroblock, the 4x4 integer DCT is applied to all the 16 blocks contained in the macroblock. An additional 4x4 transform (Hadamard) transform is used to further decorrelate the 16 DC coefficients obtained after the integer 4x4 DCT. Most of the energy of the block after DCT is present in the low frequency components.

NOSSDAV'06, May 22-23, 2006, Newport, Rhode Island, USA

2. PROPOSED ALGORITHM

The proposed method [10] is an extension of the work done by Lou and Liu [7] where digital signatures are used to authenticate images compressed by JPEG [15]. The authors in [7] propose that the DC or the mean value of every quantized luminance block be used as the feature data of an 8x8 block for generation of digital signatures. Although the DC value represents the mean of every block and contains most of the energy, the block values can be changed without changing the mean (DC) of the block. The DC value alone might not be an appropriate criterion for authenticating every block. Digital video has an additional dimension (temporal). Characteristics of digital video should be taken into account while applying this scheme to digital video. The proposed method authenticates a group of pictures (GOP) by generating a unique digital signature for every GOP.

2.1 Digital Signature Generation

Figure.1 shows the digital signature generation part. For INTRA 4x4 and INTER macroblocks, the quantized DC coefficient and the first two quantized AC coefficients (low frequency coefficients in zig-zag scan order) surrounding the DC value of every 4x4 block are taken as the feature data for the macroblock. For INTRA 16x16, all the non zero quantized Hadamard transform coefficients and the first two quantized AC coefficients in zig-zag scan order surrounding the DC value form the feature data for this type of macroblock. These feature data are collected in a buffer for every coded macroblock in every frame until the end of the GOP is reached. In H.264/AVC, the end of GOP is indicated by an instantaneous decoder refresh (IDR). At the end of the GOP, the values present in the buffer are hashed using SHA to produce a 160 bit message digest. This digest along with the sender's private key are used in producing a unique digital signature. This digital signature is encrypted using RSA encryption [9] and sent as supplemental information in the video bitstream.

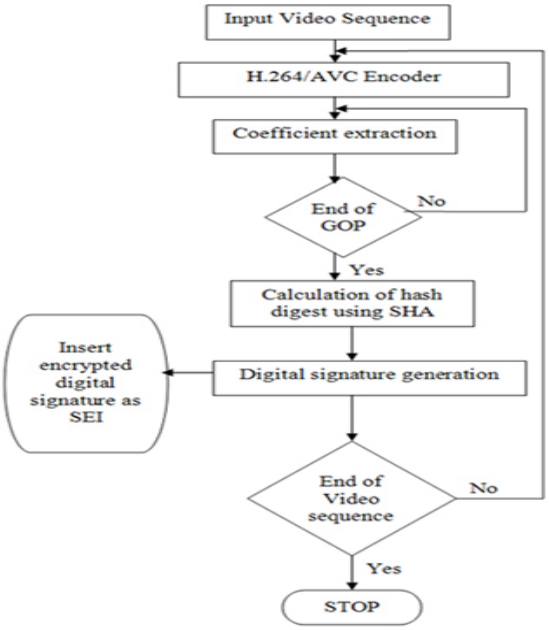

Figure.1 Proposed digital signature generation scheme

2.2 Digital Signature verification

In the verification part, the H.264/AVC decoder stores in a buffer the DC coefficient and the first two AC coefficients in zig-zag scan order before inverse quantization and inverse transform of every 4x4 block of a macroblock for INTRA 4x4 and INTER modes. For INTRA 16x16 modes, all the non zero Hadamard coefficients are also stored in the buffer.

Figure.2 describes the signature verification process. When the end of GOP is detected, the data in the buffer is hashed to give a message digest. The digest and the sender's public key along with the digital signature sent by the encoder are used in verifying the digital signature. The output of the verification process is a binary result indicating if the video has been tampered. However in the case of authentication failure it is not possible to identify whether frame tampering or sender forgery is the cause of failure.

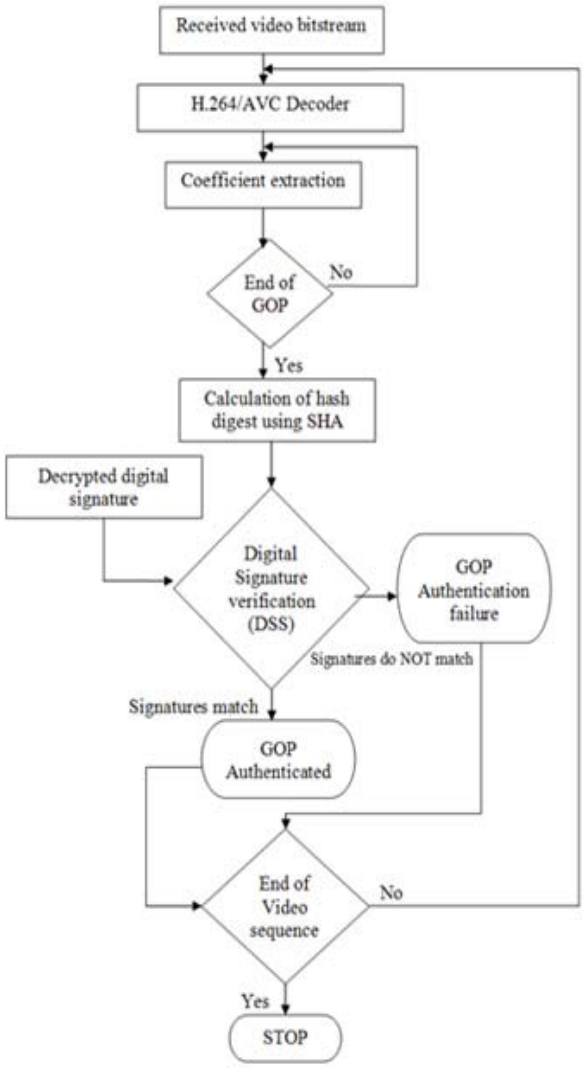

Figure.2 Proposed digital signature verification scheme

2.3 Authentication failure analysis

In order to find out the reason for authentication failure, we propose that the hash (SHA) of every frame also be computed at the encoder apart from the digital signature. The decoder can request the encoder to send the hash digests of all the frames contained in a GOP in the event of an authentication failure. To detect sender forgery, the decoder calculates the hash of every received frame in a GOP and matches it with the corresponding hash sent by the encoder. If the hash digests of all the frames in a GOP match and yet the signature verification fails then conclusion can be drawn that sender forgery is the cause of authentication failure. On the other hand, if the hash digests of the encoder and decoder do not match it can be ascertained that the corresponding frame has been tampered. Figure.3 depicts the signature analysis process.

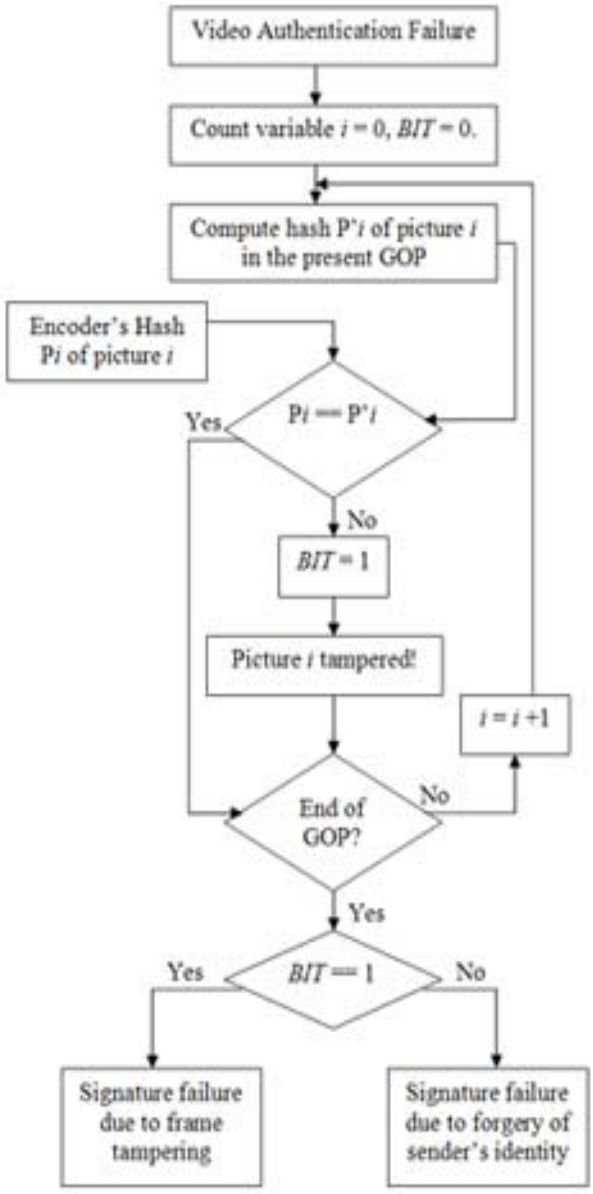

Figure.3 Authentication failure analysis

2.4 Signature and Hash embedding

The signature and hash are embedded as Supplemental Enhancement Information (SEI) in the H.264/AVC bitstream [3]. Since decoding of SEI by the H.264/AVC decoder is optional, embedding it as SEI ensures backward compatibility with decoders without support for this authentication scheme.

2.5 Storage requirements

The signature consists of 2 numbers, each of 160 bits. The hash requires 160 bits. Let x be the total number of frames in a GOP and y be the total number of GOP in the video. Since a signature is generated for every GOP, signatures for the whole video would require y*320 bits. The total bandwidth required by the hash digests for authentication failure analysis of the whole video would be y*x*160 bits. The digital signature and hash digests are encrypted before transmission by a 1024 bit key. Thus a total of 1024* (y*x + 1) bits would be required for authenticating the entire video sequence where 1024 bits are for the digital signature and 1024*y*x bits are for the hash digests.

3. RESULTS

The H.264/AVC software [11] (Version JM 7.5c) was modified to implement this algorithm. The software was configured to produce an I frame once every 10 frames, thus making the total number of frames in the GOP equal to 10. Figures 6 - 11 demonstrate the different types of tampering (DC attack, cropping, rotation, inversion, frame-reordering and change in quantization parameter) our proposed algorithm can detect.

Figure.6 shows the advantages of including AC coefficients in the authentication process. The top left corner of the figure shows the distortion achieved by tampering six 4x4 blocks without changing the mean (DC) value of the 4x4 block.

Original Block of pixels (Location 0,0)				Modified Block of pixels (Location 0,0)			
34	139	219	226	165	154	153	200
34	140	218	225	110	106	104	133
33	139	217	225	120	117	115	132
29	136	218	221	224	221	211	188
Sum of all pixels = 2453				Sum of all pixels = 2453 (unchanged)			

Figure.4 DC Attack

We refer to this type of tampering as DC attack. The values of the 4x4 block located at (0, 0) are shown in Figure 4 for more clarity.

Figure.5 Original Foreman frame

Figure.6 Foreman frame after DC attack

Figure.7 Cropped Foreman frame

Figure.8 Rotated Foreman frame

Figure.9 Inverted Foreman frame

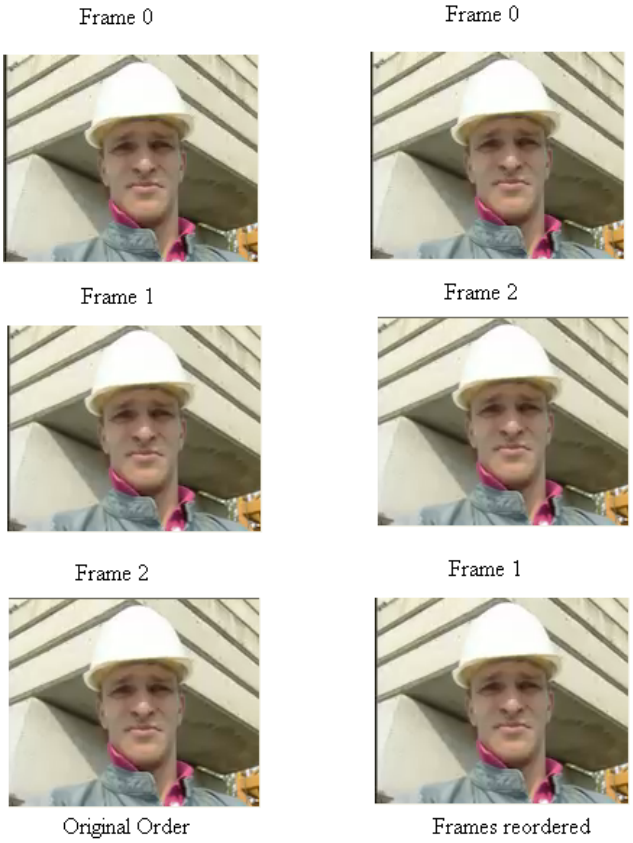

Figure.10 Frame reordering attack

Figure.11 Frame 0 quantized with different QP parameters (Left is QP29 and right is QP35).

The reason for authentication failure for the attacks mentioned in the former part of this section is that when the frames or the sender's public key are tampered with, the hash produced at the decoder is different than the one produced by the encoder as the

SHA is a one way hash and the probability of getting a same hash with two different sets of data is close to nil [4].

3.1 Comparison with previous work

The major contribution of our work is that it can detect various spatial and temporal manipulations and point out the reason for the authentication failure at reduced complexity. Table 1 shows the comparison between the previous work [7] and proposed work.

Table 1. Comparison with previous work

Parameter	Previous Work [7]	Current work
Media	Images compressed by JPEG and videos compressed by Motion JPEG	Videos compressed by H.264/AVC
Authentication	Soft	Hard
Features extracted in transform/spatial domain	Transform	Transform
Feature Extraction	Luminance only	Luminance only
Type of feature data for every block	DC only	DC + AC
Number of feature coefficients required	Fixed (E.g. for QCIF 396 coefficients required - See Figure 12)	Variable (E.g. for QCIF - see Figure 12)
Digital Signature	1 per 8x8 block	1 per coded video sequence
Detect spatial manipulations	Yes (See Figures 6, 7, 8 and 11)	Yes (See Figures 6, 7, 8, 9 and 11)
Detect temporal manipulations	No	Yes (See Figure 10)
Detect sender forgery	Yes	Yes
Detect tampered location	Yes (Block level)	Yes (Frame level)
Self recovery for tampered locations	Yes	No
Detect quantization changes	Yes	Yes
Computational complexity	High (For QCIF size, 396 encryption passes per frame required for generating digital signature for video sequence)	Low (For QCIF size, 1 encryption pass per frame and 2 encryption passes per video sequence required for generating digital signature for video sequence)

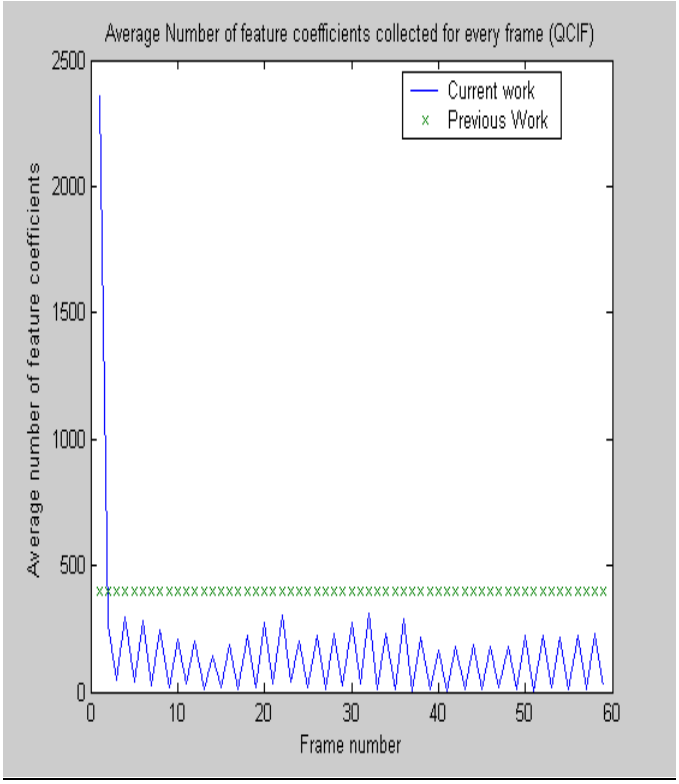

Figure.12 Number of feature coefficients collected for every frame

4. CONCLUSIONS

Features unique to a macroblock are extracted and are used in the authentication process. For Intra and Inter 4x4 blocks, the DC coefficients and the first two AC coefficients along zig-zag scan have been taken as feature data whereas for INTRA 16x16 macroblock, the HT coefficients and two AC coefficients in zig-zag scan order from every 4x4 block are used as feature data. The DSS is used for generating the digital signature and RSA for encryption of the digital signature.

From the above results and discussions, it can be seen that our proposed algorithm can detect malicious temporal and spatial manipulations to the video, identify the frames where these manipulations have taken place and also point out the fraudulent intentions of an imposter attempting to forge the identity of the genuine sender, for videos encoded by H.264/AVC. Every GOP is authenticated separately which provides encoder the ability to re-transmit only the corresponding GOP in case of authentication failure. The process of identification of tampered frames by calculating the hash of individual frame leads to an increase in

computational complexity and also an increase in the bits transmitted by the encoder.

5. ACKNOWLEDGEMENTS

We would like to thank the anonymous reviewers for their constructive suggestions that help to improve the quality of this paper.

6. REFERENCES

[1] Chin-Yung Lin, "Watermarking and digital signature techniques for multimedia authentication and copyright protection", PhD Thesis, Columbia University, 2001.

[2] B.B. Zhu, M.D. Swanson and A.H. Tewfik. "When seeing isn't believing [multimedia authentication technologies]", IEEE Signal Processing Magazine, Vol.21, pp. 40- 49, Mar. 2004.

[3] H.264/AVC International Standard

ITU-T Rec. H.264 | ISO/IEC 14496-10 version 3

[4] Secure Hash Standard

Federal Information Processing Standards

Publication-180-1

http://www.itl.nist.gov/fipspubs/fip180-1.htm

[5] Digital Signature Standard

Federal Information Processing Standards

Publication-186-1

http://www.itl.nist.gov/fipspubs/fip186-1.htm

[6] I.E.G. Richardson, "H.264 and MPEG-4 Video Compression". Chichester, West Sussex: Wiley, 2003.

[7] D.-C. Lou and J.-L. Liu, "Fault resilient and compression tolerant digital signature for image authentication", IEEE Trans. on Consumer Electronics, Vol. 46, pp. 31-39, Feb. 2000.

[8] Special Issue on H.264/AVC

IEEE Transactions on Circuits and Systems for Video Technology, vol 13, pp. 557-725, July 2003

[9] Rivest, R.L., Shamir, A. and Adleman, L.M., "A method for obtaining digital signatures and public-key cryptosystems", Communications of the ACM vol. 21, pp. 120-126, Feb. 1978.

[10] N. Ramaswamy, "A video authentication scheme using Digital Signature Standard and Secure Hash algorithm for H.264/AVC Main Profile". MS Thesis, The University of Texas at Arlington, Aug. 2004.

[11] H.264/AVC Reference Software

http://bs.hhi.de/~suehring/tml/

[12] G. Sullivan, P. Topiwala and A. Luthra, "The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions," SPIE Conference on Applications of Digital Image Processing XXVII, vol. 5558, pp. 53-74, Aug. 2004.

[13] S.-K Kwon, A. Tamhankar and K.R. Rao, 'Overview of H.264 / MPEG-4 Part 10" ISME, Hong Kong, Oct. 2004.

[14] JVT Documents

ftp://standards.polycom.com

http://ftp3.itu.int/av-arch/jvt-site

[15] M. Ghanbari, " Standard codecs: Image compression to advanced video coding," IEE, UK, 2003