7-2013

Technique for Authenticating H.264/SVC Streams in Surveillance Applications

Wei ZHUO

Robert H. DENG
Singapore Management University, robertdeng@smu.edu.sg

Jialie SHEN
Singapore Management University, jlshen@smu.edu.sg

Yongdong WU

Xuhua DING
Singapore Management University, xhding@smu.edu.sg

See next page for additional authors

DOI: https://doi.org/10.1109/ICMEW.2013.6618259

7-2013

Technique for Authenticating H.264/SVC Streams in Surveillance Applications

Wei ZHUO

Robert H. DENG
Singapore Management University, robertdeng@smu.edu.sg

Jialie SHEN
Singapore Management University, jlshen@smu.edu.sg

Yongdong WU

Xuhua DING
Singapore Management University, xhding@smu.edu.sg

See next page for additional authors

TECHNIQUE FOR AUTHENTICATING H.264/SVC CODESTREAMS IN VIDEO SURVEILLANCE APPLICATIONS

Zhuo Wei, Yongdong Wu, Robert H. Deng

ABSTRACT

Surveillance codestreams coded by H.264/SVC (scalable video coding), which consists of one base layer and one or more enhancement layers, supply flexible and various quality, resolution, and temporal (sub)codestreams such that clients with different network bandwidth and terminal devices can seamlessly access them. In this paper, we present a robust authentication scheme for them in order to insure the integrity of SVC surveillance codestreams, named AUSSC (Authenticating SVC Surveillance Codestreams). AUSSC exploits cryptographic-based authentication for base layer and content-based authentication for enhancement layers. For content-based authentication, AUSSC extracts full features from the first frame of each GOP (group of picture) and partial features from "active" macroblocks of other frames. Performance analysis indicates that AUSSC is robust to content-preserving manipulations and sensitive to content-changing manipulations of enhancement layers. Compared with cryptographic-based and watermarking-based authentication schemes, experimental results show that AUSSC causes less computation complexity and smaller compression overhead. Thus, it appears that AUSSC is suitable for real time SVC surveillance applications.

Index Terms— H.264/SVC, Authentication, Surveillance application

1. INTRODUCTION

The scalable extension of H.264, referred to as scalable video coding (SVC) [1], is composed of one base layer, which is compatible with the H.264 advance video coding (AVC), and one or more enhancement layers which improve the video in one of three scalability dimensions (time, quality and resolution). With video surveillance becoming an integral part of our security infrastructure, the industry is currently starting to use SVC to compress digital video for surveillance applications such that clients with different network bandwidth and terminal devices can seamlessly access various SVC surveillance (sub)codestreams. For example, for a home SVC surveillance system as shown in Figure 1, family members may take a look at their home by mobile devices on their way home or view home security by IPAD/laptop at public place (e.g., library). Moreover, they can furthermore clearly check their home by computer at office. However, under open network situations, any layer of SVC surveillance codestreams can be modified by sophisticated processing tools such that the surveillance content can be changed without leaving any visible traces for human eyes. Therefore, surveillance data have virtually no value as legal proofs since doubts would always exist.

Authentication scheme aims to thwart any unauthorized manipulations by verifying the integrity and source of data. It has the standard requirements, e.g., security, computational efficiency and communication efficiency. For authenticating SVC surveillance codestreams, an authentication scheme should further satisfy the following properties. Firstly, it preserves the scalability of the original SVC surveillance codestreams. That is, it authenticates an SVC codestream once at the source, but allows verification of three-dimensional (sub)codestreams. Secondly, it is able to pinpoint the tampered regions if tampering indeed occurred. In addition, it is robust or resilient to content-preserving manipulations which do not change the semantic meaning of a codestream and it is able to detect content-changing manipulations which modify the semantic meaning of the codestream.

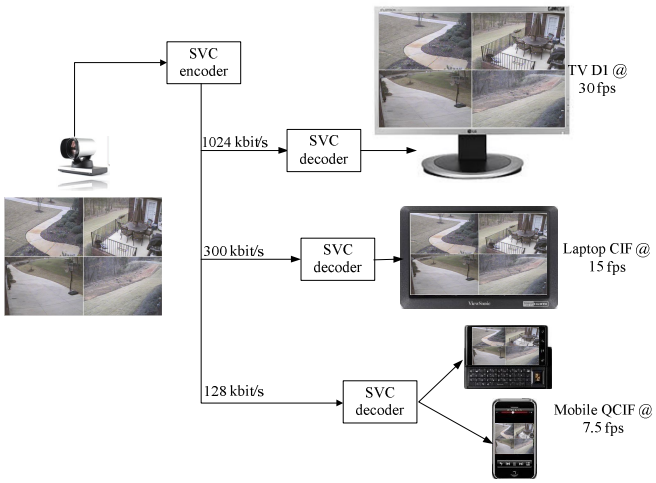

Fig. 1. An SVC home surveillance system.

In this paper, we propose a novel authentication scheme for SVC surveillance codestreams, named AUSSC (Authenticating SVC Surveillance Codestreams), which integrates authentication operations into the SVC coding process. According to SVC layer characteristic, base layer acts as the reference layer to quality/spatial enhancement layers. Due to the fact that base layer is the basement of SVC codestreams, it must be transmitted to clients at any adaptation sessions. AUSSC exploits cryptographic-based authentication for base layer codestream in case of any bit changing. It further involves the authentication on temporal scalability since base layer codestream contains frame order and time stamp. On the other hand, qualtiy/spatial scalability supplements enhanced information for the base layer in order to produce various higher quality/resolution images. Since those images have the same content as the original images used by encoders, AUSSC can take content-based features extracted from original images to authenticate qualtiy/spatial scalability. We name the first frame of a GOP as key-frame while other frames of the GOP as non-key frames. With hierarchical prediction proerty, AUBBS extracts full features of key-frames but only "partial" features from "active" macroblocks of non-key frames. Analysis indicates that AUSSC is secure and can preserve three-

dimensional SVC scalabilities. Compared with other authentication schemes, experimental results show that AUSSC causes the least communication overhead.

The rest of this paper is organized as follows. We review existed authentication algorithms of SVC codestreams in Section 2 and present preliminary in Section 3. Then, AUSSC is explained in Section 4. Experiments and analysis are given in Section 5. At last, we draw our conclusion in Section 6.

2. RELATED WORKS

Authentication schemes for SVC codestreams can be classified into three major types: cryptographic-based authentication, watermarking-based authentication and content-based authentication.

Cryptographic-based authentication scheme Cryptographic-based authentication is generally to perform compression first, and then authentication. For instance, Yu in [2] and Mokhtarian et al. in [3] proposed similar hash chain schemes for SVC codestreams. The schemes hash each enhancement layer and attach the hash value to the lower layer of the same frame. Cryptographic-based authentication schemes are very sensitive to content modifications, including content-changing manipulations, content-preserving manipulations and bit errors caused by transmission or storage noise. Proposed schemes are also not transparent to users because they totally depend on the SVC layer structure. Furthermore, since they must execute hash function for each layer, their computation complexity and communication overhead are proportional to the number of layers.

Watermarking-based authentication scheme Watermarking-based authentication embeds a reference object, e.g., image or message, into an SVC codestream. Grois et al. in [4] reviewed recent watermarking-based authentication schemes for SVC. As the reference object and the SVC codestream are mixed together, the embedded object will be tampered as long as the SVC codestream is maliciously tampered. For example, Meerwald and Uhl in [5] designed a robust watermarking-based authentication by embedding the same watermark into both base layer and enhancement layers for quality/spatial scalability. For the sake of robustness and security, watermarking-based authentication schemes must embed the reference object into each layer of SVC. Otherwise, the non-watermarked layers can be easily tampered without being detected. Moreover, the capacity of embedding watermarking is very limited in enhancement layers because most quantized coefficients of enhancement layers are equal to zero.

Content-based authentication scheme Content-based authentication [6] separates from compression. It ensures the authenticity of multimedia features such as edges, the feature of Matrix transform, and the feature of transform domain. To authenticate a codestream, a content provider extracts its multimedia features, generates a reference object with the extracted features, and delivers the reference object to end users via a secure channel. Upon receiving the video codestream and the reference object, an end user extracts the video features as the provider did, and checks whether the extracted features match those in the reference object. In such schemes, they have their own robustness range for content-preserving and content-changing manipulations [6] so as to satisfy authentication requirements of general multimedia, such as MPEG series and H.264. However, since SVC codestreams can supply various quality and resolution images, especially, the lowest quality of SVC codestreams is outside robustness range of features, content-based authentication fails to simultaneously protect both base layer and enhancement layers.

3. NMF-NMF-SQ

The NMF (Non-negative Matrix Factorization) algorithm [7] is able to decompose a non-negative matrix into two non-negative matrix factors. Monga and Mihcak in [8] proposed a robust and secure image hashing methods, named NMF-NMF-SQ hashing, as follows.

- Given an image $\mathbf{I}$, pseudorandomly select p overlapping subimages $\mathbf{A}_i$ with size $m \times m$, $1 \leq i \leq p$;
- Perform a rank r_1 NMF transform on each subimage ($r_1 \ll m$), $\mathbf{A}_i \approx \mathbf{W}_i \times \mathbf{F}_i^T$, where $\mathbf{W}_i$ and $\mathbf{F}_i$ are $m \times r_1$ matrices, $\mathbf{F}_i^T$ is the transpose of $\mathbf{F}_i$;
- Randomly arrange the matrices $\mathbf{W}_i$ and $\mathbf{F}_i$ into a new image $\mathbf{J}$ with size $m \times 2pr_1$;
- Perform a rank r_2 NMF transform on $\mathbf{J}$, $\mathbf{J} \approx \mathbf{W} \times \mathbf{H}$, where the size of $\mathbf{W}$ is $m \times r_2$ and the size of $\mathbf{H}$ is $r_2 \times 2pr_1$;
- Concatenate the columns of $\mathbf{W}$ with the rows of $\mathbf{H}$ as a hash vector $\mathbf{h}$. Denote the length of $\mathbf{h}$ as v;
- Generate pseudorandom weight vectors $\{\mathbf{t}_i\}_{i=1}^{u}$ ($u \leq v$) with a secret k_e, where each $\mathbf{t}_i$ is the length of v. Let $V_i = \langle \mathbf{h}, \mathbf{t}_i \rangle$ be the inner product of vector $\mathbf{h}$ and vector $\mathbf{t}_i$. The hash is $\{V_1, \cdots, V_u\}$.

NMF-NMF-SQ is very robust to a large class of perceptually insignificant manipulations. For example, it can tolerate the JPEG compression with quality factor $QF = 1\%$ [6]. As H.264/SVC utilizes a similar integer Discrete Cosine Transform (DCT) as the JPEG standard, the robustness property is conjectured to be applicable to H.264/SVC codestream too [9]. Based on the tampering detection dataset CASIA [10], our experimental results on H.264 codestreams indicate that NMF-NMF-SQ indeed keeps excellent robustness and sensitivity property when QP (quantization parameter) of an image is no more than 38.

4. AUTHENTICATION AND VERIFICATION

AUSSC seamlessly integrates authentication/verification operation into the SVC coding process.

4.1. Authentication

AUSSC takes use of cryptographic-based authentication to guarantee the integrity and authenticity of base layer and utilizes content-based authentication to ensure enhancement layers. Figure 2 illustrates the flow of authentication of an SVC codestream.

4.1.1. Authentication of base layer

Given the encoded frame Φ, the provider takes its base layer Φ_b and a key k_b shared by provider and receiver as input to produce MAC ϕ as

$$\phi = \mathcal{H}(k_b, \Phi_b) \tag{1}$$

where $\mathcal{H}(\cdot)$ is a standard one-way hash function (e.g., SHA-1). AUSSC takes use of SEI (Supplement Enhancement Information) NALU to encapsulates the hash, which is the same method as the one in [3]. Each AU (Access Unit) owns its SEI NALU.

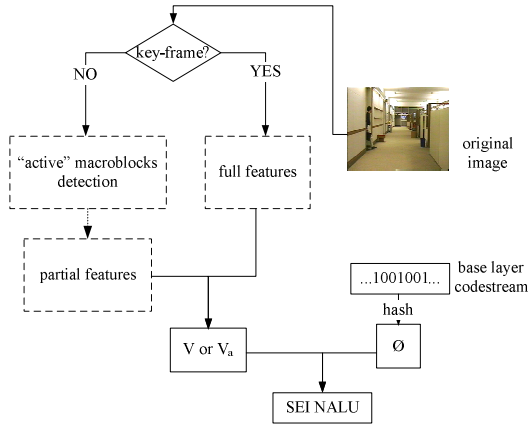

Fig. 2. The flow of authentication.

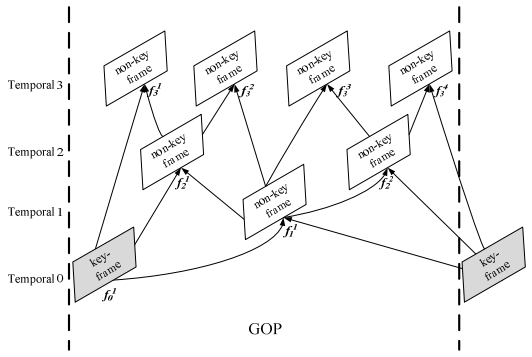

Fig. 3. Hierarchical prediction structure of SVC temporal scalability.

4.1.2. Authentication of enhancement layers

In order to authenticate SVC enhancement layers, AUSSC extracts content-based features from original images, which may be full features of key-frames or partial features of non-key frames as shown in Figure 2.

Full features of key-frame Since key-frames are the most important reference frames and always act as synchronous frames, AUSSC exploits NMF-NMF-SQ hash to extracts full features from original images which should be of the same resolution as base layer does. As NMF-NMF-SQ requirements, a pseudorandom weight vectors $\{\mathbf{t}_i\}_{i=1}^{u}$ $(u \leq v)$ should be generated in order to compress the hash vector $\mathbf{h}$ by inner product. In this article, we utilize RC4 which takes the secret key k_e and an IV (initial vector) as input to generate $\{\mathbf{t}_i\}_{i=1}^{u}$ $(u \leq v)$ for each key-frame. AUSSC chooses the slice header of base layer as IV. The inner product's result is NMF-NMF-SQ hash vector, named V. Its length is u and each element occupies 12 bits. At last, V is written into its SEI NALU.

Partial feature of non-key frame With the hierarchical prediction structure of SVC temporal scalability as shown in Figure 3, AUSSC authenticates the non-key frames one by one. Firstly, AUSSC authenticates the non-key frame f_1^1. Then, higher temporal layers' non-key frames, e.g., f_2^1, f_2^2, which take use of key-frames and/or the authenticated non-key frames as reference frames are authenticated. Analogously, the highest temporal layer's non-key frames, e.g., f_3^1, f_3^2, f_3^3 and f_3^4, are gradually processed. Different from extracting full features of key-frames, AUSSC only extracts partial features from "active" fields of non-key frames in order to reduce its computation complexity and communication overhead. Because adjacent frames normally have lots of non-changing fields inside a GOP, especially for surveillance applications, that is, the corresponding fields have the same content among those frames. As a consequence of the fact that the integrity of reference frames have been authenticated, the non-changing fields of current non-key frame can also be authenticated.

In AUSSC, an "active" field corresponds to a 16×16 macroblock which is the coding unit of H.264 and can be judged by SVC base layer's encoding information, such as code mode and *cbp* (coded block pattern). *cbp*'s value of a macroblock indicates if there are 4×4 subblocks which contain non-zero coefficients (i.e., prediction residuals). In AUSSC, Intra period must be greater or equal to GOP size, hence only P/B can be the non-key frame. Given a non-key frame (i.e., P/B frame) and one of its macroblock, we judge "active" or "static" as follows. If code mode is Intra mode or PCM mode, the macroblock coding is independent of reference frames. Thus, the macroblock must be authenticated alone, then AUSSC sets it as "active" one; if code mode is Direct/Skip mode, the macroblock is coded without sending residual error or motion vector. Hence, decoders can directly reconstruct it based on reference frames, then AUSSC sets it as "static" one; if code mode is Inter mode, when *cbp* is non-zero, AUSSC sets it as "active" one, otherwise, AUSSC sets it as "static" one because the content of macroblock or its subblocks can be found from reference frames based on mvs. For the *Hall* SVC codestream, it contains one base layer ($QP40$, CIF) and two quality enhancements ($QP30$ and $QP20$, CIF). Figure 4 illustrates the detection results of "active" macroblocks.

Assuming a non-key frame has M of "active" macroblocks, AUSSC takes the following steps to authenticate them. Firstly, 256 pixels of each "active" macroblock are permutated by a pseudorandom sequence. Let $E_{k_e}(\cdot)$ be the permutation algorithm. Denote $\mathbf{B} = \{b_0, \cdots, b_{255}\}$ an "active" macroblock. To permute $\mathbf{B}$ using k_e, the sender determines a permutation function $\pi_n : [0, n-1] \rightarrow [0, n-1]$ as follows.

Step 1. Compute $\mathbf{C} = \{c_0, c_1, \cdots, c_{n-1}\}$ where $c_i = E_{k_e}(i, R)$ for $0 \leq i \leq n-1$, and R is a string related to slice header of base layer and frame number.

Step 2. Sort the n ciphertexts in the ascending order, such that $c_{i_0} < c_{i_1} < \cdots < c_{i_{n-1}}$, where $0 \leq i_j < n$.

Step 3. Define $\pi_n(i_j) = i$. In other words, c_i is replaced with c_{i_j}.

π_n is a secure pseudo-random permutation if an encryption algorithm (e.g., AES) is used. Secondly, M of encrypted "active" macroblocks are reorganized into a new subimage A_1 $(16 \times 16 * M)$. Thirdly, AUSSC performs NMF on A_1 with rank r_1 in order to produce two matrixes: $16 \times r_1$ and $r_1 \times 16 * M$, then reorganizes them into a new subimage A_2 $(16 \times (M+1)r_1)$. At last, AUSSC further performs NMF on A_2 with rank r_2 and produces two matrixes: $16 \times r_2$ and $r_2 \times (M+1)r_1$. The concatenation of columns of these two matrixes gives the features of non-key frames, named V_a, which is also an input to the SEI NALU.

4.2. Verification

After receiving an SVC (sub)codestream, if it contains both base layer and enhancement layers, AUSSC should verify base layer and enhancement layer of the received (sub)codestream as shown in Figure 5.

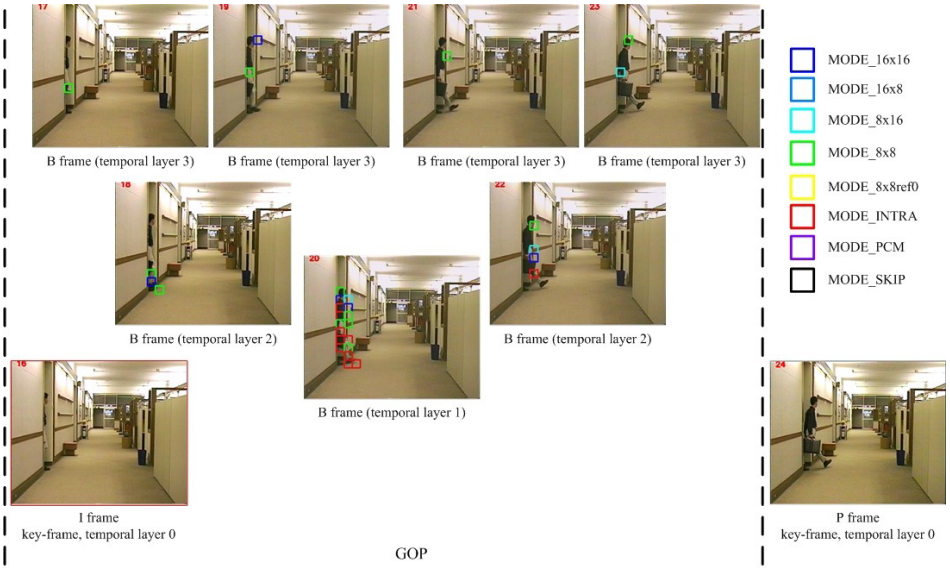

Fig. 4. Detection of "active" macroblocks.

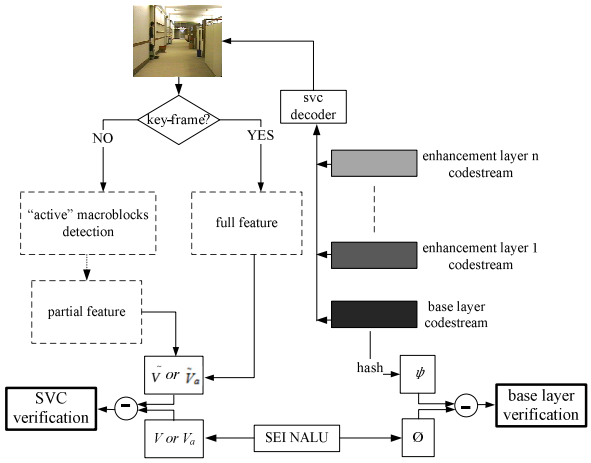

Fig. 5. The flow of verification.

4.2.1. *Verification of base layer*

For the base layer Ψ_b of a frame Ψ, AUSSC first calculates its MAC value ψ as

$$\psi = \mathcal{H}(k_b, \Psi_b). \tag{2}$$

If $\psi = \phi$, AUSSC framework accepts the base layer's codestream. It further involves the authentication on temporal scalability because the time stamp and frame number are authenticated by MAC, hence, AUSSC can detect frame reordering attack in which the temporal order of frames may be changed. Otherwise, the codestream of the base layer is tampered and AUSSC directly rejects the base layer and all the enhancement layers of Ψ.

4.2.2. *Verification of enhancement layer*

In order to verify received enhancement layers, AUSSC utilizes two different content-based authentication techniques for key-frame and non-key frame.

Verification of key-frame AUSSC takes NMF-NMF-SQ hash to verify the feature integrity of a key-frame. As what the provider does, with the shared key k_e, the receiver extracts feature $\tilde{V}$ from the key-frame decoded from base layer and the received enhancement layers. If the received codestream contains spatial enhancement layers, the key-frame should be downsampled before feature extraction. AUSSC calculates the error $e = \| \tilde{V} - V \|$. If e satisfies robustness range of content-based feature, the key-frame is accepted; otherwise, it is regarded as modified one and AUSSC directly reject the GOP. The detail of detection theoretic analysis and the probability of miss and false alarm can be found in reference [8].

Verification of non-key frame Based on the hierarchical prediction structure, AUSSC furthermore verifies received non-key frames one by one. If the received base layer's codestream is accepted, its coding information (i.e., code mode and *cbp*) are verified. AUSSC takes the same method as shown in Subsection 4.1.2 to differentiate "active" macroblocks from "static" macroblocks. If a received SVC codestream contains spatial enhancement layers, the resolution of non-key frames should be downsampled to the same as base layer's in order to locate "active" macroblocks. After detection of "active" macroblocks, AUSSC extracts partial features $\tilde{V}_a$ with the same way as shown in Subsection 4.1.2. The distance $e_a = \| \tilde{V}_a - V_a \|$ indicates if "active" macroblocks are modified. On the other hand, since "static" macroblocks do not contain residuals, the same content of the macroblock (e.g., code mode is $MODE\_SKIP$ or $MODE\_16 \times 16$) or its subblocks (e.g., code mode is $MODE\_16 \times 8$, $MODE\_8 \times 16$ or $MODE\_8 \times 8$) can be found in reference frames based on mvs. In order to verify "static" macroblocks, AUSSC performs the subtraction of "static" subblocks and corresponding subblocks of reference frames to verify content integrity. Our experiments on *Hall* sequence indicate that results of all pairs of pixel substraction approximate to zero.

Table 1. Communication overhead (bytes)

	quality scalability				spatial scalability			
	original length	key-frame	no-key frame	overhead %	original length	key-frame	no-key frame	overhead %
Hall 38 key frames	1661133	4978	18171 ($M = 6048$)	1.39	1583646	3154	11253 ($M = 3742$)	0.91
Bridge-close 63 key frames	4652164	8253	38919 ($M = 12964$)	1.01	4391955	5229	13749 ($M = 4574$)	0.43
Bridge-far 63 key frames	2583726	8253	1911 ($M = 628$)	0.39	2521044	5229	393 ($M = 122$)	0.22

5. EXPERIMENTS AND ANALYSIS

The experiments on Ubuntu 10.04 are carried out on a PC with 2.53GHz Intel dual-core processor. We choose *Bridge-far* (500 frames), *Bridge-close* (500 frames), and *Hall* (300 frames) sequences as our experiments, which are encoded into SVC codestreams by JSVM 9.19 [11]. We set GOP size and Intra period as 8, and the QPs of enhancement layers are no more than 38. For quality scalability experiments, the encoded SVC codestreams contain three layers (i.e., QP_{40}, QP_{30}, and QP_{20}, CIF(352×288)), and **u** is set as 64. For spatial scalability experiments, the encoded SVC codestreams consist of one base layer (QP_{40}, QCIF (176×144)) and two enhancement layers (QP_{35} and QP_{20}, CIF), and **u** is set as 32. In addition, the parameters of NMF-NMF-SQ are $p = 10$, $m = 50$, $r_1 = 2$, and $r_2 = 1$ for QCIF. We design our experiments based on $NMFlib$[1], and exploit ***nmf_alspg*** (alternating least squares using a projected gradient method) to compute the factorization.

Security AUSSC inputs k_b and Φ_b and outputs an MAC which ensures that a receiver who knows the secret key k_b can detect any changes to the base layer. On the other hand, besides the base layer, SVC codestreams also supply various quality and resolution (sub)codestreams to clients using different enhancement layers. For key-frames, as required by the NMF-NMF-SQ algorithm, $\mathbf{u}\{\mathbf{t}_i\}_{i=1}^{u}$ pseudorandom weight vectors are generated using RC4 with the secret key k_e and an initialization vector IV. When a stream ciphers such as RC4 is used, the initialization vector IV should be unique for each AU such that the resulting pseudorandom vectors do not repeat themselves. In AUSSC, IV is generated as

$$IV = \mathcal{F}(H_n, H_s). \tag{3}$$

where $\mathcal{F}$ is a one-way function, H_n represents the SVC scalable information (e.g., temporal identifier), and H_S denotes the slice header of base layer which is protected by MAC. Because the header information is in clear text, IV can be deduced from the SVC codestream at the decoder/verifier side. For non-key frames, the pseudo-random permutations of "active" macroblocks are generated using a block cipher such as AES. It's well known that block ciphers can be regarded as secure pseudo-random permutations and it is computationally infeasible to distinguish the output of a block cipher from that of a truly random permutation [12]. This implies that an attacker can not forge V_a without the knowledge of the secret key for the block cipher, although the attacker know the content of macroblocks from base layer.

Computation complexity Authentication computation cost of AUSSC consists of base layer's authentication cost t_b and enhancement layer's authentication cost t_e. t_e can be key-frame cost t_e^k and non-key frame cost t_e^n. t_b is MAC computation cost. Usually, $t_b \ll t_e$ and can be omitted. For t_e^k, according to [8],

$$t_e^k = p \cdot o(m^2 r_1) + o(2mpr_1r_2) + o(mr_2 + 2pr_1r_2). \tag{4}$$

where the first term is the rank r_1 NMF cost on $p \cdot m \times m$ matrices, the second term is the rank r_2 NMF cost on $m \times 2pr_1$ matrix, and the third term is due to pseudorandom statistics obtained from the resulting NMF-NMF vector of length $mr_2 + 2pr_1r_2$. For instance, experimental results of spatial scalability show that the three terms in Eqn. 4 are 3464.5 μs, 210 μs, and 62.84 μs, respectively. Then, t_e^k is 3737.34 μs. As for t_e^n, it is related to the number M, and can be calculated as follows,

$$t_e^n = e_M + o(16M16r_1) + o(16(M+1)r_1r_2). \tag{5}$$

Where e_M is the sum of permutation time for all "active" macroblocks, it can be ignored because it is far less than the second and third items; the second item is the rank r_1 NMF cost on $16M \times 16$; the third item is the rank r_2 NMF cost on $16 \times (M+1) \cdot r_1$. For example, if M is equal to 20, the second and third items of Eqn. 5 are 1402 μs and 270 μs, respectively. Therefore, t_e^n is 1672 μs.

On the other hand, verification is an inverse way of authentication, its cost should contain t_b, t_e^k, and t_e^n. In addition, the computation cost t_e^s of verifying "static" macroblocks also should be added. For example, with M being 20, t_e^s of QCIF are about 64.63 μs and 129.26 μs for a P frame and for a B frame, respectively.

Compression overhead Communication overhead of each AU is $l = l_b + l_e + l_h$ bytes, where l_b denotes a fixed-length overhead for base layer authentication, l_e is the size of the enhancement layers features, and l_h is the header size for an SEI. Typically, we select $l_b = 16$ and $l_h = 19$, hence,

$$l = 16 + l_e + 19 = 35 + l_e. \tag{6}$$

l_e can be the size of key frame's features l_e^k or non-key frame's features l_e^n. Table 1 describes the experimental results of quality and spatial scalabilities. The first column indicates that there are 38 of key-frames for *Hall* and 63 key-frames for *Bridge-close* and *Bridge-far*. The second and sixth columns show the length of original SVC codestreams. The third and seventh columns describe compression overhead of key-frames, where l_e^k is 131 bytes (i.e., $l_e^k = 96$) for quality scalability and 83 bytes $l_e^k = 48$ for spatial scalability. The fourth and eighth columns show the value of M and the communication overhead caused by non-key frames for quality and spatial scalability. The fifth and ninth columns give the communication overhead in the form of percentage. As a result, the average overhead of quality and spatial scalabilities are 0.93% and 0.52%.

Scalability When users only receive base layer codestream, AUSSC verify it by cryptographic-based authentication. When users further

[1] http://www.ee.columbia.edu/ grindlay/code.html

Table 2. Comparison with other H.264/SVC authentication schemes

	robustness/sensitivity	tampered location	authentication operation	dependence on SVC structure	communication overhead
Cryptographic authentication	sensitive to any bit change	no	all layer hash	yes	quality scalability 2.19% spatial scalability 2.62%
Watermarking authentication	semi-fragile	yes	all layer watermarking	yes	quality scalability 8.66% spatial scalability 2.93%
AUSSC	sensitive to base layer change robust to enhancement layers	yes	base layer hash content-based features	no	quality scalability 0.93% spatial scalability 0.52%

receive different quality and/or spatial enhancement layers, AUSSC verify them by content-based authentication. Therefore, AUSSC preserves the scalability of the original SVC surveillance codestreams.

Robustness and sensitivity The base layer is authenticated by cryptographic-based authentication, thus AUSSC is sensitive to any bits changing. On the other hand, AUSSC authenticates the content-based features to ensure integrity of enhancement layers. Because the features are robust to content-preserving manipulations and sensitive to content-changing manipulations. Therefore, transcoding on enhancement layers, e.g., reducing bit-rates/resolution of SVC codestreams, can still be accepted, however, the content-changing operations on enhancement layers can be rejected.

5.1. Comparison with other authentication schemes

As AUSSC exploits the characteristics of SVC to ensure its authenticity, it is robust to content-preserving manipulations but sensitive to content-changing manipulations for enhancement layers. However, cryptographic-based authentication is sensitive to any bit change of SVC and watermarking-based is semi-fragile to content-preserving manipulations. In other words, AUSSC achieves a good balance between robustness and sensitivity as shown in the second column of Table 2. The third column indicates that cryptographic-based authentication can not locate tampered areas. Furthermore, the fourth column of Table 2 shows that AUSSC only depends on base layer and content-based features of the highest quality/resolution images, while cryptographic-based authentication and watermarking-based authentication must involve every layer of SVC to prevent the attacks on unprotected layers. In the fifth column of Table 2, cryptographic-based authentication and watermarking-based authentication depend on layer prediction relationship of SVC in order to construct hashing chain or embed watermarking. AUSSC is independent of SVC structure since AUSSC only considers the authentication of base layer's codestream and content-based features of SVC. Hence AUSSC is transparent to users. The last column illustrates the communication overhead. In our SVC experiments, GOP size is 8 and encoded SVC sequences have three layers (one base layer and two enhancement layers). The cryptographic-based authentication [3] appends 960 bytes overhead per GOP (i.e., each frame has 120 bytes overhead). Hence, the average overheads of the scheme in [3] are 2.19% and 2.62% of the original codestream for quality and spatial scalability, respectively. In general, AUSSC produces the smallest communication overhead as shown at the last column of Table 2. In addition, AUSSC's communication overhead is constant, while the counterparts of cryptographic-based authentication [3] and watermarking-based authentication [5] increase with the number of enhancement layers. For example, with GOP size being 8, each frame will carry 40 bytes more overhead in [3] when an SVC sequence contains one more enhancement layer.

6. CONCLUSION

In this paper, we proposed a robust authentication scheme for SVC surveillance codestreams, named AUSSC. According to the characteristics of SVC architecture, AUSSC exploited cryptographic-based and content-based authentication techniques for base layer and quality/resolution enhancement layers, respectively. Based on the hierarchical structure of temporal scalability, AUSSC extracted full features of key frame while only partial features of non-key frames. Our experimental results showed that AUSSC caused less compression overhead than cryptographic-based and watermarking-based authentication, and had low computation complexity. Thus, it appears that AUSSC is suitable for real time SVC surveillance applications. In addition, it can be robust to content-preserving manipulations and sensitive to content-changing manipulations, and it can further locate tempered fields.

7. REFERENCES

[1] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the h.264/avc standard," *IEEE Transactions on Circuits and System for Video Technology*, vol. 17, no. 9, pp. 1103–1120, 2007.

[2] H. Yu, "Scalable streaming media authentication," *IEEE International Conference on Communications*, pp. 1912–1916, 2004.

[3] K. Mokhtarian and M. Hefeeda, "Authentication of scalable video streams with low communication overhead," *IEEE Transactions on multimedia*, vol. 12, no. 7, pp. 730–742, 2010.

[4] D. Grois and O. Hadar, "Recent advances in watermarking for scalable video coding," *Wateramarking, Intech Open Access Publisher*, 2012.

[5] P. Meerwald and A. Uhl, "Robust watermarking of h.264/svc-encoded video: Quality and resolution scalability," *International Workshop on Digital Watermarking*, pp. 156–169, 2010.

[6] S. H. Han and C. H. Chu, "Content-based image authentication: current status, issues, and challenges," *International Journal of Information Security*, vol. 9, no. 1, pp. 19–32, 2010.

[7] D. D. Lee and H. S. Seung, "Algorithms for non-negtive matrix factorization," *Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference. MIT Press*, pp. 556–562, 2000.

[8] V. Monga and M. K. Mihcak, "Robust and secure image hashing via non-negative matrix factorizations," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 376–390, 2007.

[9] Q. Sun, D. He, and Q. Tian, "A secure and robust authentication scheme for video transcoding," *IEEE Transactions on Circuits and Systems for video technology*, vol. 16, no. 10, pp. 1232–1244, 2006.

[10] CASIA, "Tampered image detection evaluation database," *http://forensics.idealtest.org/*, 2010.

[11] JSVM, "Joint scalable video model software, http://ip.hhi.de/imagecom g1/savce/downloads/svc-reference-software.htm," 2011.

[12] J. Katz and Y. Lindell, "Introduction to modern cryptography," *Chapman & Hall/CRC*, 2008.