Pyramid Based Interpolation for Face-Video Playback in Audio Visual Recognition

Dereje Teferi and Josef Bigun

School of Information Science, Computer, and Electrical Engineering (IDE)
Halmstad University, P.O.Box 823, SE-301 18
Halmstad, Sweden
{Dereje.Teferi,Josef.Bigun}@ide.hh.se

Abstract. Biometric systems, such as face tracking and recognition, are increasingly being used as a means of security in many areas. The usability of these systems depend not only on how accurate they are in terms of detection and recognition but also on how well they withstand attacks. In this paper we developed a text-driven face-video signal from the XM2VTS database. The synthesized video can be used as a means of playback attack for face detection and recognition systems. We use Hidden Markov Model to recognize the speech of a person and use the transcription file for reshuffling the image sequences as per the prompted text. The discontinuities in the new video are significantly minimized by using a pyramid based multi-resolution frame interpolation technique. The playback can also be used to test liveness detection systems that rely on lip-motion to speech synchronization and motion of the head while posing/speaking. Finally we suggest possible approaches to enable biometric systems to stand against this kind of attacks. Other uses of our results include web-based video communication for electronic commerce.

1 Introduction

Biometrics is the study of automated methods for uniquely identifying and recognizing humans based upon one or more intrinsic physiological or behavioral traits. The traits used for recognition include, face, fingerprints, hand geometry, gait, handwriting, iris, retina, voice etc. Biometric systems are in use in applications such as security, finance, banking etc [1], [2].

The need for increased security is becoming apparent at all levels as transaction fraud and security breaches became more and more a threat. It is increasingly reported, in research and development, that biometrics systems have high level of accuracy in identifying and recognizing people. However, its application is not as widespread as expected. One drawback is that biometric data of a person (such as face, speech, etc) are not secret and cannot be replaced anytime the user wants to or whenever they are compromised by a third party. The problem of spoofing is minimal if the authentication system works with the help of a human supervisor as in border control where the presented trait can be visually checked to see if it is genuine or fake. However, the risk is high

for remotely controlled biometric applications such as banking and e-commerce that use the internet [3]. The risk of spoofing on automated biometric systems can be reduced significantly by combining multiple traits into the system and incorporating liveness detection.

Biometric system attacks may occur in a number of points throughout the process of enrollment, identification or verification (Fig 1). The attacks could be set off at the sensor, network, algorithm, the template database etc. These vulnerable points of a biometric system are discussed in detail in [4]. This work presents a possible playback attack targeting points two of the verification process.

Since the purpose of this research is not to attack biometric systems but to prevent them from such, we conclude with solutions to differentiate playback videos from that of live ones.

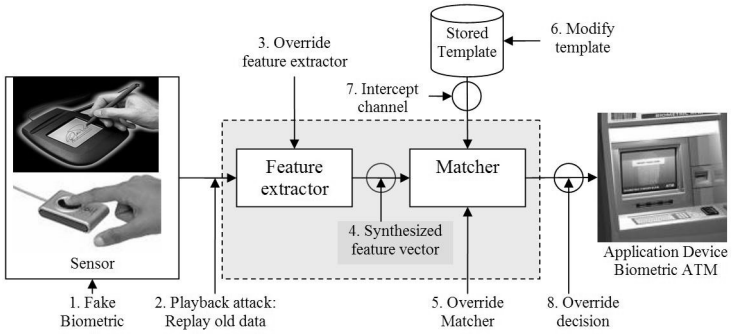

Fig. 1. Possible attack points of a biometric system

Technological advances in audio-video capture and processing have enabled the development of sophisticated biometric systems. As such it has also presented opportunities for more vicious system attacks. Impersonation can be done, for example, by audio-video playback and application of image processing techniques without the presence of the real client.

The actual presence of the client can be assured to a certain degree by liveness detection systems. Liveness detection is an anti-spoofing mechanism to protect biometric systems [5], [4]. It is performed by, for example, face part detection and optical flow of lines to determine liveness score [5], analysis of fourier spectra of the face image [6], lip-motion to speech synchronization[7], body temperature, on the spot random queries such as pronouncing of random sequences of digits, iris size scanning under varying illumination etc.

Text-prompted liveness detection and audio-video based recognition systems use random digits as their text prompts. This is because random numbers are easier to read for the client than random texts and numbers are easier to synthesize. Accordingly, the digit speaking image sequences of the XM2VTS database

are used in this work for the experiment. XM2VTS is a 295 subject audio-video database that offers synchronized video and speech data for research in multimodal recognition systems [8]. In this paper we developed a text prompted face-video synthesis for testing the capability of liveness detection and audio-visual person verification systems against attacks.

We use HMM based speech recognizer to identify the locations, in order of 10^{-7} of a second, of each digit spoken by the subject. The pre-recorded image sequence is then reshuffled according to system prompt (or digits entered). However, the process of shuffling creates discontinuities between digits in the new image sequence. This discontinuity between the last frame of a digit and the first frame of the next digit is minimized by interpolating frames in between using a pyramid based multiresolution frame interpolation method. This method reported a better result, for the attacker, by reducing the *block effect* and *false match* as illustrated in Fig 6.

2 Speech Recognition

Human speech is defined by the structure of the vocal tract and the use of the vocal cords. Speech is a biometric trait that can be used to uniquely identify people. The digit speaking audio database of the XM2VTS database is used in this work. Vectorial representation of the speech is performed using the Mel-Frequency Cepstral Coefficients(MFCC) as implemented in [9].

Hidden Markov Model (HMM) can be used to process any time series such as that of speech [10]. We use Htk (a portable toolkit for building and manipulating Hidden Markov Models) to model the speech recognition system in our work 2. Htk is mainly designed to build HMM models for speech recognition.

The waveform files are converted into a sequence of discrete equally spaced acoustic parameter vectors using MFCC according to a defined configuration. The configuration in our HMM model has a sampling period of 10 ms, a window size of 25 ms while the number of cepectral coefficients is set to 12. A prototype HMM model with 39 feature vectors and 3 left-right states is defined and used for the training.

The final HMM model constructed from a series of trainings is used for the speech recognition as shown in 2. A speech signal is passed to the HMM model for recognition of the digits. The result - a transcription file - is used to reshuffle the image sequences according to the system prompt.

3 Motion Estimation for Frame Interpolation

Motion estimation is the process of finding optimal motion vectors that describe the movement of pixels or blocks of pixels from frame to frame in an image sequence. A number of motion estimation techniques are discussed in [11], [12], and [13]. We use motion estimation to calculate motion vectors for interpolating new frames to reduce the discontinuity in the image sequence. The discontinuity

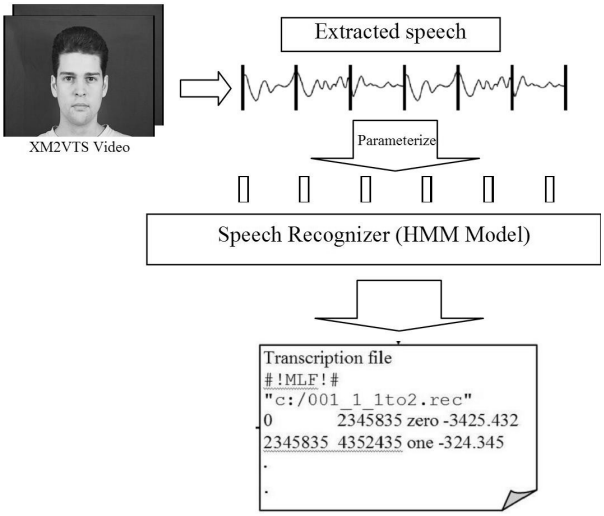

Fig. 2. Speech Recognition System

occurs due to the rearrangement of sequence of frames according to the prompted text.

Motion estimation is a common technique used in video compression. A video usually have some redundant data between adjacent frames within a certain period of time $(t_1 - t_0)$. The redundancy is small for fast paced and complex motion of objects and background in the video and high otherwise. This redundancy is exploited to compress the video. That is a reference frame (sometimes known as the independent frame) is taken from a sequence every n frame apart. Then the middle frames are predicted from these frames as well as from previously predicted frames by the codec. The codec actually uses the motion vector and prediction error to reconstruct the *dependent* frames. Forward prediction, where the new frames are predicted from previous frames, or backward prediction, where the new frames are predicted from future frames, or both can be used for estimation of the new frames in the middle. Many codecs use this method for video compression.

Typical audio visual recognition systems use head-shoulder video as an input. Moreover, it is not too difficult to acquire video of a person uttering the 10 digits. Given such a sequence an attacker could proceed as discussed below.

Assuming the video is captured with a stationary camera, the background will be near-constant. Therefore, little information is lost or added between adjacent frames, such as teeth, mouth and eyes. That is, there is a high probability that most part of a frame exist in another frame although translated to a different location. First, the points in motion are extracted using absolute difference between the two frames (Fig 3).

$$AD = |\mathbf{F}(k,l) - \tilde{\mathbf{F}}(k,l)| \tag{1}$$

Now that we know the points in motion, the motion vector (MV) is calculated only for blocks around these points. For each block around a point in motion on frame $\mathbf{F}$, we look for its parallel pattern in frame $\tilde{\mathbf{F}}$ within a local neighborhood by using block matching algorithm.

(a) Frame 1 (b) Frame 2 (c) Pixels in Motion

Fig. 3. Points in Motion between frame $\mathbf{F}$ and $\tilde{\mathbf{F}}$

3.1 Block Matching Algorithm

Block matching is a standard technique used by video compression techniques to encode motion in image sequences. A review of block matching algorithms is given in [13], [12], and [14]. The motion vector calculated from the block matching algorithm is used for frame interpolation. This method of using motion of objects between frames for interpolation is referred to as motion compensated frame interpolation (MCFI) [15].

Non-overlapping blocks are created over the points in motion of frame $\mathbf{F}$ (*the frame at time* t_0) and a search area is defined on frame $\tilde{\mathbf{F}}$ (*the frame at time* t_1). The search area is larger than the size of the block by *expected* displacement e of the object in motion (Fig 4). Then, Schwartz inequality is applied to find the most parallel pattern for the block in frame $\mathbf{F}$ from the search area in frame $\tilde{\mathbf{F}}$.

Let $\mathbf{f}$ and $\tilde{\mathbf{f}}$ be vectorial representations of patterns from frames $\mathbf{F}$ and $\tilde{\mathbf{F}}$ and $<,>$ be the scalar product defined over the vector space. Then, we have

$$| < \mathbf{f}, \tilde{\mathbf{f}} > | \leq \|\mathbf{f}\| \|\tilde{\mathbf{f}}\|$$

$$cos(\theta) = \frac{| < \mathbf{f}, \tilde{\mathbf{f}} > |}{\|\mathbf{f}\| \|\tilde{\mathbf{f}}\|} = \frac{|\mathbf{f}^T \tilde{\mathbf{f}}|}{\|\mathbf{f}\| \|\tilde{\mathbf{f}}\|} \leq 1 \tag{2}$$

where $\mathbf{f} = (f_1, f_2, ..., f_{k-1}, f_k, f_{k+1}, ...)$ and $\tilde{\mathbf{f}} = (\tilde{f}_1, \tilde{f}_2, ..., \tilde{f}_{k-1}, \tilde{f}_k, \tilde{f}_{k+1}, ...)$ are vector forms of the 2D pattern $\mathbf{f}$ and $\tilde{\mathbf{f}}$ from frames $\mathbf{F}$ and $\tilde{\mathbf{F}}$ respectively and $cos(\theta) \in [0, 1]$ is the similarity measure between the patterns.

The most parallel pattern $\tilde{\mathbf{f}}$ in the frame $\tilde{\mathbf{F}}$ is found by maximizing $cos(\theta)$. This can be done by repetitive scalar products. That is the pattern $\mathbf{f}$ from frame

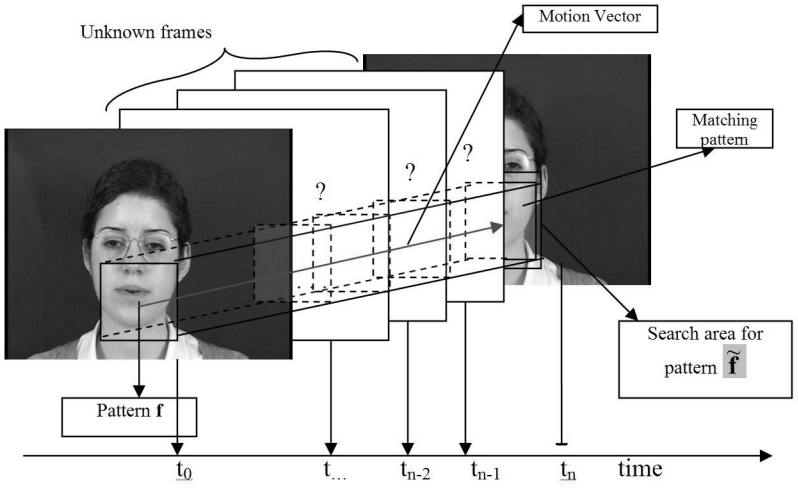

Fig. 4. Motion Vector and unknown frames in a sequence

$\mathbf{F}$ is glided over a neighborhood of *expected* parallel local neighborhood of $\tilde{\mathbf{f}}$ in the frame $\tilde{\mathbf{F}}$ and the most similar pattern is selected as a match.

The motion vector for the point at the center of pattern $\mathbf{f}$ is calculated as the displacement between pattern $\mathbf{f}$ and pattern $\tilde{\mathbf{f}}$ as illustrated in Fig 4. That is:

$$MV(k,l) = x + iy \tag{3}$$

Where x and y are the horizontal and vertical displacements respectively of the block/pattern $\mathbf{f}$, *(k,l)* is the index for the center of pattern $\mathbf{f}$ and $i = \sqrt{-1}$.

3.2 Pyramid Based Approach for Block Matching

Using large block size while calculating the motion vectors above gives rise to what is known as *block effect* in the interpolated frame. Whereas making the block size small may result in multiple similarities and a probable choice of a *false match* for the motion vector.

To solve this problem, a pyramid based frame interpolation method which makes use of both large as well as small size blocks hierarchically is employed to calculate the motion vector in the right direction. That is, initially large block size is used to get a crude directional information on which way the block is moving. Here it is unlikely to get a false match as the size of the block is very large. This motion vector is used to determine the appropriate search area in the next step of the the block matching process so that the probability of a false match is minimal. The search area is also reduced to a smaller local neighborhood because we have information on where to look for the pattern. Then, we reduce the size of the block by a factor of n and calculate the motion vector again (Fig 5). The motion vector calculated for each block is again used to determine where

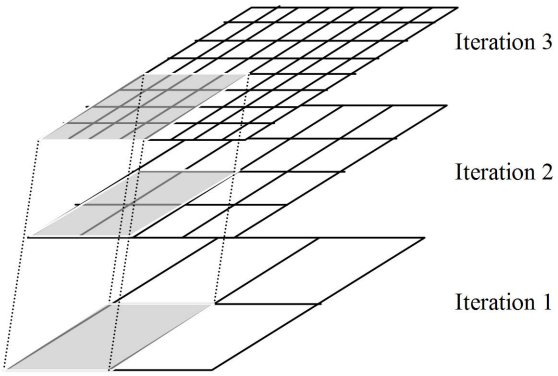

Fig. 5. Pyramid based iteration where n=4 and number of iterations =3

the search area should be in the next iteration. This process is repeated until a satisfactory result is achieved in the interpolated frame.

The final complex valued motion vector matrix is used to interpolate the new frames between $\mathbf{F}$ and $\tilde{\mathbf{F}}$. The number of frames to be interpolate d depends on the norm of the motion vector and is determined at run-time. For a length 2 time units, the actual frame interpolation is done by dividing the motion vector at point (k,l) by 2 and moving the block in frame $\tilde{\mathbf{F}}$ centered at $(k+x,l+y)$ to the new frame at $(k+x/2,l+y/2)$. Then the block at the new location in frame $\tilde{\mathbf{F}}$ is moved back the same distance. That is, let $\mathbf{F}$ be the frame at t_0, $\tilde{\mathbf{F}}$ the frame at t_1 and $\mathbf{F}^{'}$ be the frame at $\frac{(t_1-t_0)}{2}$, then

$$\mathbf{F}^{'}(k+x/2, l+y/2) = \tilde{\mathbf{F}}(k+x, l+y) \tag{4}$$

$$\mathbf{F}^{'}(k,l) = \tilde{\mathbf{F}}(k+x/2, l+y/2) \tag{5}$$

where x and y are the real and imaginary parts of the motion vector at (k,l).

Consecutive interpolations are made in analogous manner. The motion vector is adjusted and a new frame is created as necessary between frames $\mathbf{F}$ and $\mathbf{F}^{'}$ as well as between $\mathbf{F}^{'}$ and $\tilde{\mathbf{F}}$. This process continues until all the necessary frames are created.

4 Video Synthesis

Audio data is extracted from the input audio-video signal and forwarded to the speech recognizer. The recognizer uses the HMM models for the recognition and returns a transcription file containing the start and end time of each digit spoken in the audio signal. A search for the prompted text is done against the transcription file and the time gap of each prompted digits within the video signal is captured. Then, the image sequences are arranged according to the order of the prompted text.

(a) Original Frame 1 (b) Original Frame 2

(c) Using large blocks (d) Using small blocks (e) Using pyramid method

Fig. 6. Results of the various frame interpolation methods

The discontinuity between the image sequences of each digit is minimized by interpolating frames using the pyramid based multiresolution frame interpolation technique summarized in section 3. The interpolated frames are attached to a silence sound and are inserted to their proper locations in the video signal to decrease the discontinuity of utterances. Finally, the video is played to represent the digit sequence prompted by the system.

5 Experiment

Th experiments are conducted on all the digit speaking face videos of the XM2VTS database. The accuracy of the text-prompted video signal is mainly dependent on the performance of the speech recognition system. The accuracy of our HMM based speech recognition system is 94%. The pyramid based frame interpolation algorithm gives optimal result when the final block size of the pyramid is 3. The discontinuity of the reshuffled video signal is reduced significantly as evaluated by the human eye, the authors. The time it takes a person to speak a digit is enough to interpolate the necessary frames between digits. Moreover, a laptop computer as a portable DVD can be used to playback the video for an audio visual recognition system.

However, there is still visible blur around the eye and the mouth of the subject. This is due to the fact that differences between the frames are likely to appear around the eye and the mouth, such as varied state of eye, teeth, and mouth opening etc (Fig 6). Such changes are difficult to interpolate in real time as they do not exist in both the left and right frames.

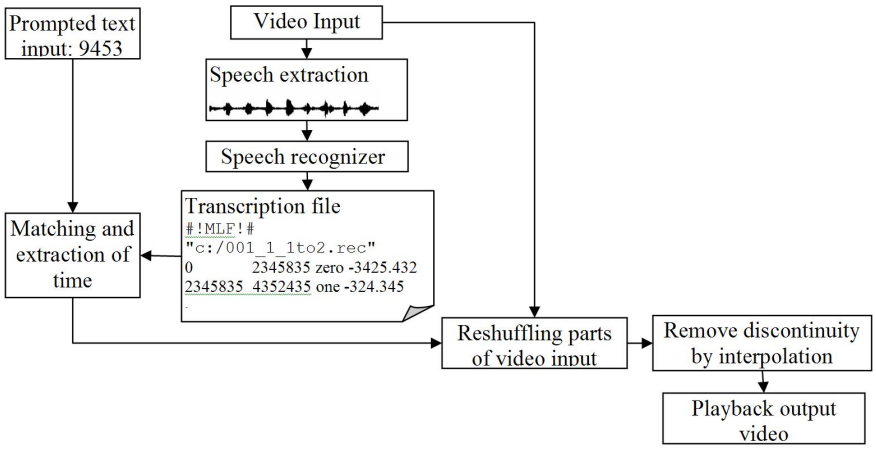

Fig. 7. Process flowchart

Biometric authentication and liveness detection systems that make use of motion information of face, lip and text prompted audio-video are easy targets of such playback attacks described here.

6 Conclusion

The risk of spoofing and impersonation is forcing biometric systems to incorporate liveness detection. Assuring liveness especially on remotely controlled systems is a challenging task. The proposed method shows a way to produce playback attacks against text-prompted systems using audio and video in real time. The result shows that assuring liveness remotely by methods that rely on apparent motion can be targets of such attacks. Our results suggest the need to increase the sophistication level of biometric systems to stand up against advanced playback attacks.

Most video-based face detection and recognition systems search for the best image from a sequence of images and use it for extracting features assuming it will yield a better recognition. However, this could have a negative impact as some of those blurry or faulty frames that are dropped could be our only hope to tell if a video-signal is a playback or a live one.

Therefore, we conclude that audio visual recognition systems can withstand such playback attacks by analyzing the area around the eyes and the mouth of the new frames between prompted text or digit to identify if they are *artificial*.

Acknowledgment

This work has been sponsored by the Swedish International Development Agency (SIDA).

References

1. Jain, A., Ross, A., Prebhakar, S.: An Introduction to Biometric Recognition. IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics **14(1)** (January 2004)
2. Ortega-Garcia, J., Bigun, J., Reynolds, D., Gonzalez-Rodriguez, J.: Authentication Gets Personal with Biometrics. IEEE Signal Processing Magazine **21(2)** (March 2004) 50–62
3. Faundez-Zanuy, M.: Biometric Security Technology. IEEE Aerospace and Electronic Systems Magazine **21(6)** (2006) 15–26
4. Ratha, N.K., Connell, J.H., Bolle, R.M.: Enhancing Security and Privacy in Biometrics-Based Authentication Systems. IBM Systems Journal **40(3)** (2001) 614–634
5. Kollreider, K., Fronthaller, H., Bigun, J.: Evaluating Liveness by Face Images and the Structure Tensor. In: AutoID 2005: Fourth Workshop on Automatic Identification Advanced Technologies.IEEE Computer Society. (October 2005) 75–80
6. Li, J., Wang, Y., Tan, T., Jain, A.K.: Live Face Detection Based on the Analysis of Fourier Spectra. In Jain, A.K., Ratha, N.K., eds.: Biometric Technology for Human Identification. Proceedings of the SPIE, Volume 5404. (August 2004) 296–303
7. Faraj, M., Bigun, J.: Person Verification by Lip-Motion. In: Computer Vision and Pattern Recognition Workshop (CVPRW). (June 2006) 37–45
8. Messer, K., Matas, J., Kitler, J., Luettin, J., Maitre, G.: XM2VTSDB: The Extended M2VTS Database. In: 2nd International Conference on Audio and Video-based Biometric Person Authentication (AVBPA). (1999) 72–77
9. Veeravalli, A.G., Pan, W., Adhami, R., Cox, P.G.: A Tutorial on Using Hidden Markov Models for Phoneme Recognition. In: Thirty-Seventh Southeastern Symposium on System Theory, SSST 2005. (2005)
10. Young, S., Evermann, G., Gales, M., Hein, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P.: The htk Book. for Version 3.3. http://htk.eng.cam.ac.uk/docs/docs.shtml (April 2005)
11. Bigun, J.: Vision with Direction: A Systematic Introduction to Image Processing and Computer Vision. Springer-Verlag Berlin Heidlberg (2006)
12. Jain, J., Jain, A.K.: Displacement Measurement and its Application in Interframe Image Coding. IEEE Transactions on Communication COM 29 (December 1981) 1799–1808
13. Cheng, K.W., Chan, S.C.: Fast Block Matching Algorithms for Motion Estimation. In: ICASSP-96: IEEE International Conference on Acoustic Speech and Signal Processing. Volume 4(1). (May 1996) 2311–2314
14. Aly, S., Youssef, A.: Real-Time Motion Based Frame Estimation in Video Lossy Transmission. In: Symposium on Applications and the Internet. (January 2001) 139–146
15. Zhai, J., Yu, K., Li, J., Li, S.: A Low Complexity Motion Compensated Frame Interpolation Method. In: IEEE International Symposium on Circuits and Systems. ISCAS 2005. Volume 5. (May 2005) 4927 – 4930