Error Concealment of Video Realistic Avatar for Virtual Conferencing System

Han-Wen Liang, Chao-Kuei Hsieh[1], and Yung-Chang Chen[2]
Department of Electrical Engineering, National Tsing Hua University, Taiwan
[1] *frost@benz.ee.nthu.edu.tw*
[2] *ycchen@ee.nthu.edu.tw*

ABSTRACT

Avatars are usually employed in virtual conferencing system as user's agent in a 3-D virtual space. In previous research, system architecture is established by integrating a 3-D model-based coder and a 2-D video coder under very low bit-rate video coding. It makes the avatar have more realistic expressions. In this paper, we develop a new error concealment scheme for the 2-D video coder as the enhancement layer, which is expected to be more suitable in the virtual conferencing system. A VQ-like algorithm is proposed to cluster the FAPs and its partial texture differences of each ROI according to the minimum quantization error. With the codevectors of each cluster, we can predict the partial texture difference by the nearest one to the received FAPs and generate a texture map. With the proposed scheme, we can alleviate the defect caused by the loss of the enhancement layer and reconstruct a vivid 3D virtual facial model while inefficient bandwidth is available.

1: INTRODUCTIONS

All submissions should be formatted according to the IEEE Computer Society guidelines. Please follow the steps outlined below when submitting your paper. ALL MANUSCRIPTS MUST BE IN ENGLISH (Maximum 6 pages) In recent years, video conferencing system is one of the most important applications in multimedia communication. However, it has the limitation of the demand of high bandwidth. Therefore, model-based coding [6,7] is usually used to reduce the bit-rate consumption in virtual conferencing system, and further, it makes possibility for users residing far away to attend a conference by using avatars in 3-D shared virtual space.

The concept of model-based coding is to parameterize a talking head, including facial expressions and head motions. The decoded parameters at the receiver are used to animate the 3-D facial model and reconstruct the talking head. Thus, a very low bit rate video coding can be achieved. For this purpose, the Facial Animation Parameters (FAP) has been defined by MPEG-4 to provide a common form of description and transmission for talking head related applications.

However, there are still some disadvantages of model-based coding approach. Facial texture is transmitted once during the session startup, and then the facial model is deformed by the animation parameters. Therefore, the synthetic facial image can not be reconstructed as vivid as the real one. In [1], a virtual conferencing system architecture has been established by integrating 3-D model-based coding and 2-D video coding to create video realistic avatars, where facial texture, transmitted by 2-D video coder, is updated on the texture map by the differences between the real video and synthetic image in the head region. But, under a very-low bit rate bandwidth condition, the difference image, transmitted as an enhancement layer, might be lost. Moreover, like the residues in conventional video codec, the loss will cause serious error propagation. An error concealment procedure is unavoidable.

There are many researchers studying error concealment techniques. Most papers stress on the concealment scheme to be compliant with different video standards, such as H.263, MPEG-1, 2, and 4. The concealment techniques use the spatial interpolation [3,4], temporal interpolation [5], or both [8].There are still some schemes for nonstandard-compliant coding techniques. Examples of this kind of work by Turaga and Chen [2], who build a model for region of interest, and use this model to replenish any missing information.

In this paper, we will develop a new VQ-liked error concealment scheme for the enhancement layer by using the relationship between FAPs and the differences between expressive texture map and neutral one, which is expected to be more suitable in the video conferencing system. First, we extract the facial animation parameters (FAP) and texture difference, which is the difference between expressive facial image and neutral facial image synthesized by 3-D model-based coding, from the input sequence. Then, we define the region of interest (ROI) by using facial animation tables (FAT) to separate the texture differences of the whole face into several partial texture differences in each ROI. Besides, each ROI are affected by specific FAP groups, that is, the partial texture difference in ROI are also affected. We cluster the FAPs and its partial texture differences of each ROI according to the minimum quantization error. With the codevectors of each cluster, we can then predict the partial texture difference by the nearest one to the received FAPs, alleviate the defect caused by the loss of the enhancement layer, and reconstruct a vivid 3D virtual face model.

The remaining sections of this paper are organized as follows. In section 2, our modification of the decoder in the virtual conferencing system is introduced. A brief

overview of our error concealment scheme is also described in this section. Section 3 presents how we separate the difference image in the head region into four parts, such as forehead, mouth, eyes, and cheeks, according to the FAT of the IST model. Then, convert each part of the difference image to Partial Texture Difference on the texture map space. We also show how we use these Partial Texture Differences to generate the estimated real image. In Section 4, the relationship between FAP and Partial Texture Difference is trained by using a VQ-like algorithm, in order to minimize the re-constructing error between the real image and the estimated one. Some experimental results are shown in Section 5. Finally, the conclusion and future work are discussed in Section 6.

2: SYSTEM OVERVIEW

An "FAP to Partial Texture Difference Conversion" process is added in the decoder as depicted in Fig. 1, in order to generate the estimated real image at the decoder when the enhancement layer isn't received.

In our previous system architecture [1], the encoder contains both image analysis part and image synthesis part, while only image synthesis part is contained in the de-coder. With such architecture, it is possible to check the quality of the synthesized image. Note that the synthesized image generated by the encoder must be identical to that one generated by the decoder, or errors will occur. It might happen while inefficient bandwidth is available for the transmitted data, either base layer or enhancement layer, lost or not received. But the FAPs as the base layer contain the most important parameters; its accuracy must be guaranteed. So we make an assumption that always accurate FAPs are received at the decoder. Therefore, we can develop an error concealment scheme added to the decoder by using the relationship between FAPs and difference images. According to this scheme, we can alleviate the defect caused by the loss of the enhancement layer, and the video realistic avatar still can be reconstructed with a vivid facial expression.

The "FAP-to-Partial Texture Difference Conversion" process is to predict the partial texture difference on the texture map domain with the received FAPs, and then texture map in neutral expression and partial texture difference are combined to form the texture map in such expression that FAPs extracted. Thereby, a new texture map is generated with the weighted sum of the predicted texture map using our process and the previous texture map which is updated without transmission errors. It involves two major steps:

1) FAP and Partial Texture Difference Extraction. (Section 3)

2) FAP-to-Partial Texture Difference Conversion. (Section 4)

3: FAP-TO-PARTIAL TEXTURE DIFFERENCE EXTRACTION

In order to obtain FAP and Partial Texture Difference, a user-customized 3D facial model is initialized first with the frontal facial image in neutral expression by extracting salient facial feature points and head profile. According to head profile and tracked feature points extracted from the following sequence, facial model is adapted, and the facial expressions are represented by the deformed facial model, which is driven by FAPs.

Besides, in order to represent realistic vivid expression of a face, it is necessary to provide the texture information of the deformed facial model. We use texture difference to represent the difference image between facial texture in different expression and that in neutral one. Instead of extracting texture difference of whole face, the texture differences of the region of interest, which is defined by facial animation tables (FAT), are extracted. Further, we convert it from image domain to texture map domain for more convenient procedure for generating updated texture map.

3.1: FAP Extraction

The MPEG-4 Facial Animation is standardized to provide a common form of description and transmission for talking head related applications. However, there is no standardization provided by MPEG-4 for the estimation of facial motion and extraction of these parameters. There are more and more methods proposed in the literature for the FAP extraction [9,10], and an efficient method proposed by Chang et al.[11] is adopted.

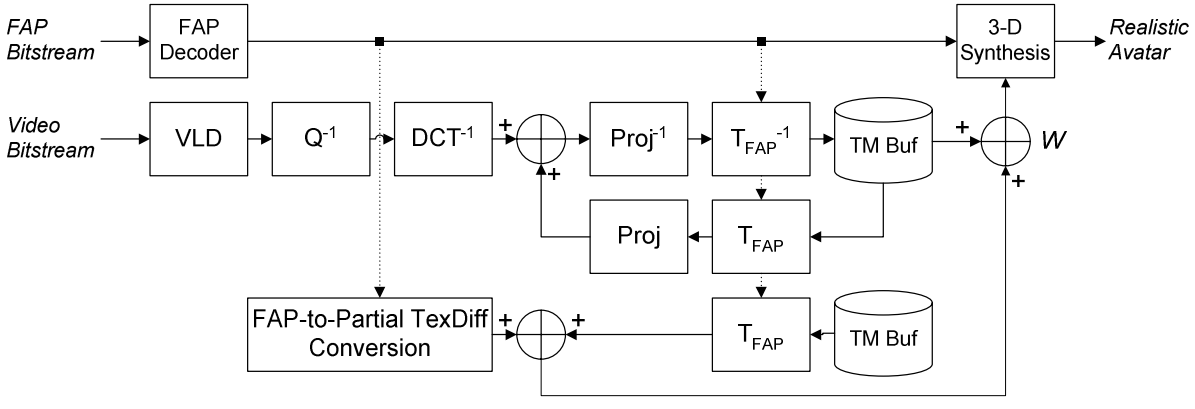


Fig. 1. The modified decoder

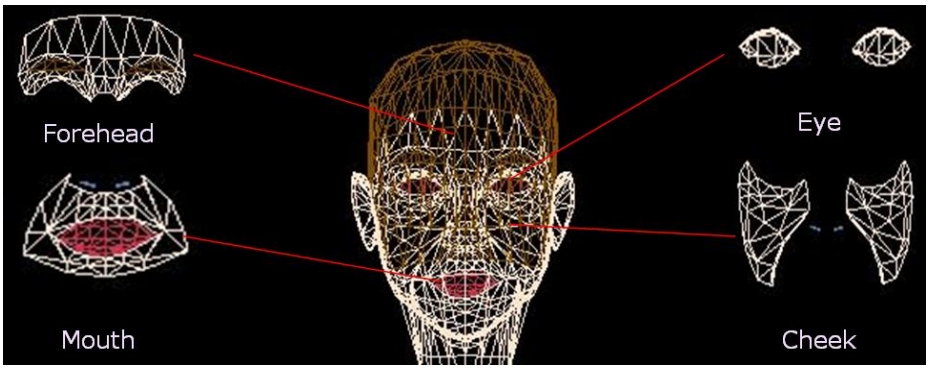


Fig. 2. The region of interest (ROI) defined by the FAPs of specific group.

3.2: Partial Texture Difference Extraction

The MPEG-4 Facial Animation is standardized to provide a common form of description and transmission for talking head related applications. However, there is no standardization provided by MPEG-4 for the estimation of facial motion and extraction of these parameters. There are more and more methods proposed in the literature for the FAP extraction [9,10], and an efficient method proposed by Chang et al.[11] is adopted.

4: FAP-TO-PARTIAL DIFFERENCE CONVERSION

The main title should be 14-point boldface Times or Times Roman, centered over both columns, followed by two blank 12-point lines. In the main title, initially capitalize nouns, pronouns, verbs, adjectives, and adverbs; do not capitalize articles, coordinate conjunctions, and prepositions (unless the title begins with such a word).

4.1: Proposed Method

K-Means [13], a classical algorithm of vector quantizer for clustering, is first adopted. With the training date set $D_m = \{x_{im}\}_{i=1}^{L}$ of the ROI m, we call codebook the set $W_m = \{w_{km}\}_{k=1}^{K}$ with modified codevector $w_{km} \in D_m$. The Voronoi Set V_m of the codevector w_{km} can be modified as $V_{km} = \{x_i \in D \mid k = \arg\min_{j=1,\ldots,k} d(x_{im}, w_{jm})\}$, where $d(x_{im}, w_{km})$ is the distortion function, which is defined to determine to the quality of synthetic image with partial update by w_{km}.

Our algorithm then consists of the following steps:

1. Initialize the codebook W_m of ROI m with K codevectors w_{km}, k=1,…,K.
2. Compute the initial Voronoi Set V_{km} and the initial quantization error $E_{D_m}^{(0)}(W_m)$.

$$E_{D_m}(W_m) = \frac{1}{L}\sum_{k=1}^{K}\sum_{x_{mi}\in V_{km}} d(x_{im}, w_{km})$$

3. Compute the new codevector w_{km}, the new Voronoi Set V_{km} of each new codevector w_{km} and new quantization error $E_{D_m}^{(1)}(W_m)$.

$$w_{km} = \arg\min_{x_{jm}} E\{d(x_{jm}, w_{km}) \mid x_{jm} \in V_{km}\}, j = 1,\ldots,K$$

4. Return the codebook if $\frac{|E_{D_m}^{(1)}(W_m) - E_{D_m}^{(0)}(W_m)|}{E_{D_m}^{(0)}(W_m)} < T$, otherwise go to Step 3

4.2: Error Concealment Using FAP-to-Partial Texture Difference Conversion

Our error concealment scheme contains two stages for generating the predicted texture map. First, when a FAP data is inputted, we search for the codevector nearest to it, predict its appropriate partial texture difference, and use this partial texture difference to update original texture map buffered. (Fig. 3)

Second, the updated texture map for the FAP data, is generated by the weighted sum of the predicted texture map and previous updated texture map without transmission errors.

The weighting factor is

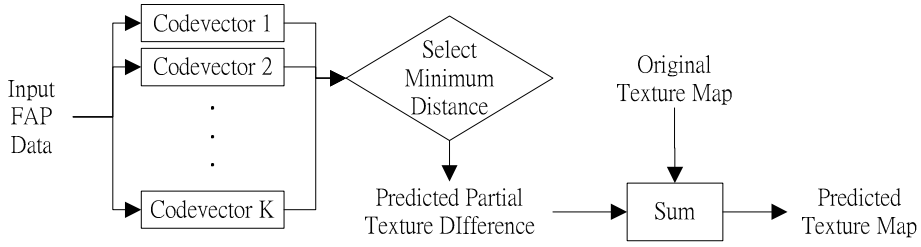


Fig. 3. The predicting procedure

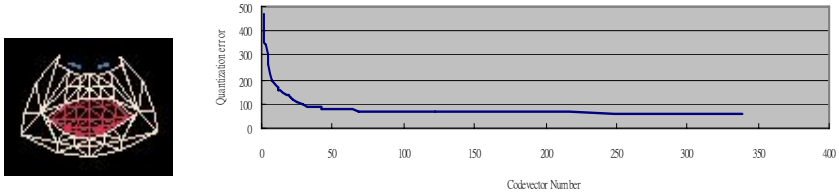


Fig. 4. The relation between the number of codevectors and corresponding quantization error in mouth ROI.

$$W^{(t)} = \begin{cases} 1, & if\,(x,y) \notin ROIs. \\ c, & if\,(x,y) \in ROIs, and \quad \hat{w}_{km}^{(t-1)} = \hat{w}_{km}^{(t)} \\ 1-c, & if\,(x,y) \in ROIs, and \quad \hat{w}_{km}^{(t-1)} \neq \hat{w}_{km}^{(t)} \end{cases}$$

where is the previous predicted codevector, is the predicted one now with the received FAPs, and c is a constant ($c \approx 1$). This makes the pixel value p of the predicted updated texture map equals to $p = W^{(t)} \cdot p^{(t-1)} + (1 - W^{(t)}) \cdot \hat{p}^{(t)}$, where $p^{(t-1)}$ is the pixel value of the previous texture map, and $\hat{p}^{(t)}$ is the pixel value of the predicted texture map.

5: SIMULATION RESULTS

For FAP-to-Partial Texture Difference Conversion, our data is a person counting numbers in English and expressing difference expressions. This sequence contains 900 frames and recorded at 30 frames per second. The frame size is 320×240 pixels and 4:4:4 RGB color space. All training data are extracted using the method described in Section 3. There are 68 FAPs defined by MPEG-4 Facial Animation, however, not all FAPs can be extracted easily. Thus, only 20 FAPs related to the lip deformation, 8 FAPs related to the eyebrow deformation, and 2 FAPs related to the eyelid deformation are used for FAP-to-Partial Texture Difference Conversion. Without extracting FAPs of cheeks region, we use FAPs of lip deformation for the training of cheeks region.

5.1: Number of Codevectors

In order to evaluate the performance of this conversion, one issue to be addressed is the choice of the number of codevectors. Fig. 4 shows as the relation between the number of codevectors and the corresponding quantization error in mouth ROI. In this experiment, we change the number of codevectors and each codevector is selected randomly in the initialization step. It shows obviously that the quantization error decreases with increasing the number of codevectors. The choice of the number of codevectors is decided depending on the storage capacity.

5.2: Performances of FAP-to-Partial Texture Difference Conversion

Following the VQ-like training procedure we proposed and the relation between the number of codevectors and corresponding quantization error shown in above section, we chose the number of codevectors of each ROI and which are depicted in Table 1. Typically, less then 10 iterations are needed for training of each ROI.

Examples of the codevectors of mouth ROI are shown in Fig. 5, and the codevectors are combined with the deformed original texture map for display purpose. For Example, Mouths with difference shapes and textures are selected by using our scheme. We use these few number of texture difference in mouth region to update the original texture map, and then the synthetic

Table 1. The number of chosen codevectors in each ROI

Region of Interest	Number of Codevectors	Storage Requirement (KB)
Mouth	29	1300
Eye	16	290
Eyebrow and Forehead	11	987
Cheek	11	897

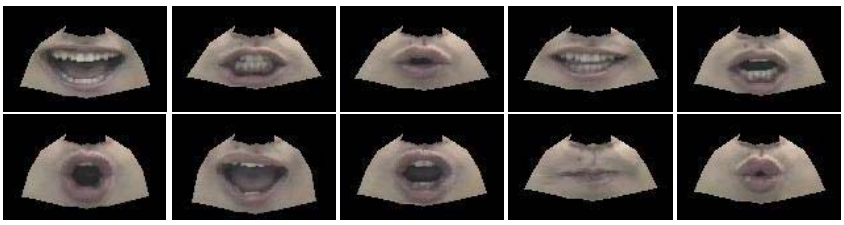


Fig. 5. An example of codevectors of mouth region

(a) (b) (c)

Fig. 6. Examples of the FAP-to-Partial Texture Difference (a) the synthetic image generated by FAP-to-Partial Texture Difference Conversion, (b) the blended image, (c) the synthetic image textured with original texture map.

images, which are shown in Fig. 6, are generated by these updated texture map. The performances of FAP-to-Partial Texture Difference are evaluated by calculating the PSNR between the blended image and synthetic image in the head region. The average PSNR is 36.67 dB, and the average PSNR of the synthetic image without texture update is 33.56 dB, which is from FAP bitstream as base layer only is received.

5.3: Error Concealment Using FAP-to-Partial Texture Difference Conversion

For evaluating the performance of our error concealment scheme, we give up one frame of transmitted residues to simulating the loss of enhancement layer. Thus, the texture map in the decoder isn't updated without receiving enhancement layer, and the following difference image is used to update this texture map. Then, error propagation occurs. So, we use our error concealment scheme to generate updated texture map by using the received FAPs to predict the partial texture difference, and the results is described in above section. After predicted texture map is generated, the previous updated texture map is combined with it to form the partial updated texture map. It is depicted in Fig. 7. In Fig. 7, we skip to difference image of frame 24, and then the texture map of frame 23 is updated by the difference image of frame 25. The error concealment results are shown in Fig. 8.

6: CONCLUSION AND FUTURE WORK

In this paper, we have proposed a method for error concealment of video realistic avatar for virtual conferencing system using the FAP-to-Partial Texture Difference Conversion. The system architecture of virtual conferencing system is based on 3-D model-based coding and traditional 2-D video coding for facial texture update to achieve very low bit-rate video coding. With inefficient bandwidth available, the enhancement layer of video bitstream might be lost. In order to avoid serious error propagation, our proposed method provides the predicted texture map according to the information of received FAPs and weighted sum with previous texture map to generate the predicted texture map more accurately. Using both spatial correlation and temporal correlation, our error concealment scheme can conceal the lost frame, and makes the video realistic avatars to have more realistic expression as users' representatives.

As the experimental result shows, the FAP-to-Partial Texture Difference Conversion can easily accomplish the error concealment scheme. By using the information of FAPs, an automatic engine of our method can provide a predicted texture map to make the 3-D facial model show the realistic expressions. The relationship between FAPs and partial texture differences can be considered as the spatial correlation; that is, the FAPs are the parameters to control the geometry of the model and partial texture differences are the information of facial texture corresponding to the geometry. Thus, with a known geometry, the texture can be predicted. Furthermore, for the error concealment purpose, we use the temporal correlation by using the weighted sum of

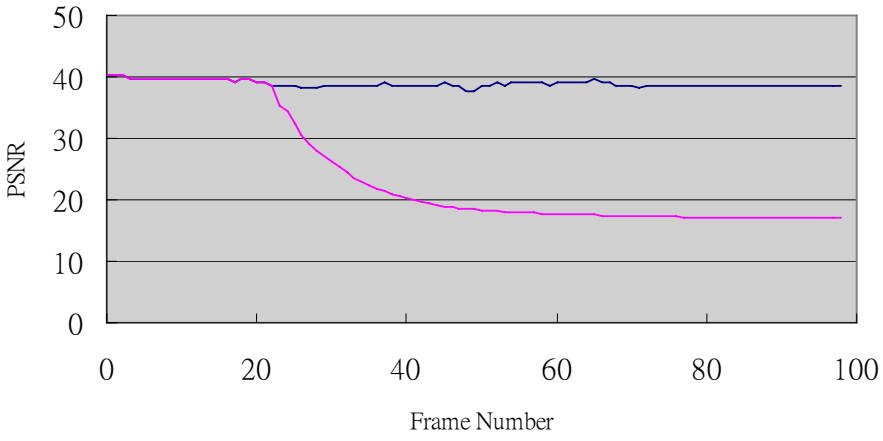
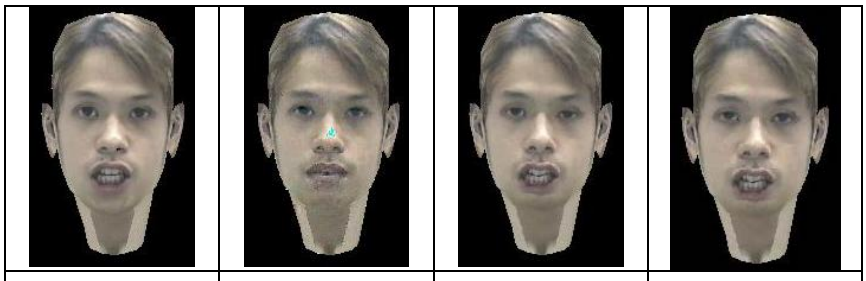


Fig. 7. PSNR plot of the sequence without transmission error and plot of the sequence with enhancement layer lost at frame 24.



(a) 38.58 dB (b) 32.22 dB (c) 34.51 dB (d) 36.33 dB

Fig. 8. Frame 25 (a) synthetic image without transmission error, (b) synthetic image with transmission error at Frame 24, (c) predicted image by using FAP-to-Partial Texture Difference Conversion, (d) predicted image generated by weighted sum of previous texture map at Frame 24 and predicted texture map at Frame 25

predicted texture map and previous texture map. It makes the predicted one more similar to the real one.

When using the partial texture difference instead of whole texture difference, the training procedure using a VQ-like algorithm can be simplified such that each partial texture difference in ROI relates to low dimensional FAP vectors, and the storage requirement of the texture information can also be reduced. The codevectors contain difference texture information, for example the codevectors of mouth region include opened mouth and closed mouth with difference size. Besides, using the partial texture difference can be considered as image-based method, for example, we generate different image of mouth region and use it with appropriate FAPs. But there is only small amount of image to be stored; the intermediate image is produced by the model-based method which is deformed by FAP distances of the codevectors and the input FAPs. This method can avoid the effect that unreal facial texture caused by large geometry warping.

However, the trained codevectors contains only the expressions of frontal face. When a human is talking, the head movements of 3-D rigid motion cause different residues. The problem is needed to be solved that is to extend the codevectors including difference poses for predicting the texture difference. Besides, the codevectors is person dependent. Thus, an efficient way to adapt the codevectors to other people also needs to be found.

REFERENCES

1. Yao-Jen Chang, Chien-Chia Chien, and Yung-Chang Chen, "Video Realistic Avatar for Virtual Face-to-Face Conferencing," Proceedings of ICME 2002, Lausanne, Switzerland, Aug.26-29, 2002.
2. Deepak S. Turaga and Tsuhan Chen, "Model-Based Error Concealment for Wireless Video," Circuits and Systems for Video Technology, IEEE Transactions on Volume 12, Issue 6, June 2002.
3. S.Aign and K. Fazel, "Temporal and Spatial error concealment techniques for hierarchical MPEG-2 video codec," in Proc. Globecom, 1995
4. W. Kwok and H.Sun, "Multi-directional interpolation for spatial error concealment," IEEE Trans. Consumer Electron., vol. 39, pp. 455-460, Aug. 1993.
5. M. J. Chen, L. G. Chen, and R. M. Weng, "Error concealment of lost motion vectors with overlapped motion compensation," Circuits and Systems for Video Technology, IEEE Transactions, on Volume 7, June 1997.
6. S. Tsekeridou and I. Pitas, "MPEG-2 error concealment based on block matching principles," IEEE Trans. Circuits Sust. Video Technol. Vol. 10, pp. 630-645, June 2000.
7. D. E. Pearson, "Developments in model-based video coding," Proc. IEEE, vol 83, pp. 892-906, June 1995
8. W. J. Welsh, S. Searsby, and J. B. Waite, "Model-based image coding," British Telecom Technol. J., vol. 8, no. 3, pp. 94-106, July 1990
9. N. Sarris, N. Grammalidis, and M. G. Strintzis, "FAP Extraction Using Three-Dimensional Motion Estimation," IEEE Trans. Circuits and Systems for Video Technology, Vol. 12, No. 10, Oct. 2002.
10. F. Dornaika and J. Ahlberg, "Fast and reliable active appearance model search for 3-D face tracking," IEEE Trans. Systems, Man and Cybernetices, Vol. 34, Issue 4, Aug 2004.
11. Jen-Chung Chou, Yao-Jen Chang, Yung-Chang Chen, "Facial feature point tracking and expressions analysis for virtual conferencing systems," Multimedia and Expo, 2001, ICME 2001, IEEE International Conference on 22-25 Aug. 2001.
12. L. Yin and A. Basu, "Partial update of active textures for efficient expression synthesis in model-based coding," Proceedings of IEEE-ICME 2000, Vol. 3, pp. 1763-1766, July 2000.
13. S. P. Lloyd, "An Algorithm for Vector Quantizer Design," IEEE Trans. Comm., vol. 28, no. 1, pp. 84-95, 1982.