UPDATING MIXTURE OF PRINCIPAL COMPONENTS FOR ERROR CONCEALMENT*

Trista Pei-chun Chen and Tsuhan Chen

Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.
{peichun, tsuhan}@andrew.cmu.edu

ABSTRACT

In this paper, we present a new statistical modeling technique called "updating mixture of principal components" (UMPC). UMPC specifically captures the non-stationary as well as the multi-modal characteristics of the data. Real-world data such as video data typically have these two characteristics. The video content changes over time and has a multi-modal probability distribution. We apply UMPC to perform error concealment for video data transmitted over networks with losses, and show that UMPC outperforms conventional error concealment methods.

1. INTRODUCTION

When transmitting video data over networks, the video data could suffer from losses. Error concealment is a way to recover or conceal the loss information due to the transmission errors. Through error concealment, the reconstructed video quality can be improved at the decoder. Projection onto convex sets (POCS) [1] is one of the most well known methods to perform the error concealment.

Error concealment based on POCS formulates each constraint about the unknowns as a convex set. The optimal solution is obtained by iteratively projecting a previous solution onto each convex set. For error concealment application, the projections of data refer to (1) projecting the data with some losses to a model that is built on error-free data, and (2) replacing the reconstructed data from the first projection with the correctly received data in the corresponding region. The success of a POCS algorithm relies on the model onto which the data is projected. We propose in this paper "updating mixture of principal components" (UMPC) to model the non-stationary as well as the multi-modal natures of the data.

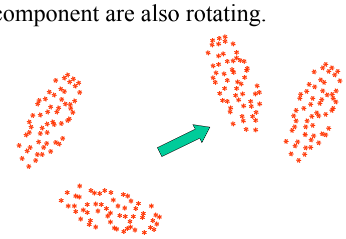
It has been proposed that the "mixture of principal components" (MPC)" [2] can represent the video data with a multi-modal probability distribution. For example, face images in a video sequence can have different poses, expressions, or even changes in the characters. It is thus natural to use a multi-modal probability distribution to describe the video data. In addition, the statistics of the data may change over time as proposed by "updating principal components" (UPC) [3]. By combining the strengths of both MPC and UPC, we propose UMPC that captures both the non-stationary and the multi-modal characteristics of the data precisely.

* Work supported in part by Industrial Technology Research Institute.

This paper is organized as follows. In Section 2, we formulate the proposed UMPC method for modeling non-stationary and multi-modal data. Both MPC and UPC are shown to be special cases of the proposed UMPC model. We apply UMPC to error concealment in Section 3. The experiment results in Section 4 show the performance of UMPC for error concealment over conventional methods. We then conclude in Section 5.

2. UPDATING MIXTURE OF PRINCIPAL COMPONENTS (UMPC)

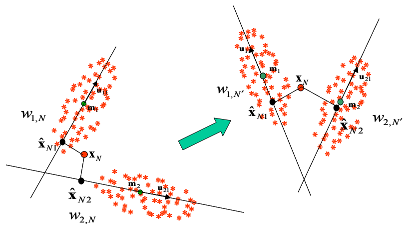
Given a set of data, we try to model the data with minimum representation error. We specifically consider multi-modal data as illustrated in Figure 1 (a). The data are clustered to multiple components (two components in this example) in a multi-dimensional space. As mentioned, the data can be non-stationary, i.e., the stochastic properties of the data are time-varying. At time instant n, the data are clustered as Figure 1 (a) and at time instant n', the data are clustered as Figure 1 (b). The mean of each component is shifting and the most representative axes of each component are also rotating.



(a) (b)

Figure 1. Multi-modal data at (a) time n (b) time n'

At any time instant, we attempt to represent the data as a weighted sum of the mean and principal axes of each component. As time proceeds, the model changes its mean and principal axes of each component as shown from Figure 2 (a) to Figure 2 (b), so that it always models the current data efficiently. To accomplish this, the representation/reconstruction error of the model evaluated at time instant n should have less contribution from data that are further away in time from the current time instant n.



(a) (b)

Figure 2. UMPC at (a) time n (b) time n'

The optimization formula for minimizing the weighted reconstruction error at time instant n can be written as:

$$\min_{\substack{\mathbf{w}_i,\mathbf{m}_j^{(n)},\mathbf{U}_j^{(n)}\\ \forall i,j}} \sum_{i=0}^{\infty}\alpha^i \left\| \mathbf{x}_{n-i} - \sum_{j=1}^{M} w_{n-i,j}\left[\underbrace{\mathbf{m}_j^{(n)} + \sum_{k=1}^{P}\left[\left(\mathbf{x}_{n-i}-\mathbf{m}_j^{(n)}\right)^T\mathbf{u}_{jk}^{(n)}\right]\mathbf{u}_{jk}^{(n)}}_{\hat{\mathbf{x}}_{n-i,j}}\right]\right\|^2 \quad (1)$$

The notations are organized as follows:

n	:	Current time index
D	:	Dimension of the data vector
M	:	Number of mixture components
P	:	Number of eigenvectors in each mixture component
$\mathbf{x}_i$	:	Data vector at time i
$\mathbf{m}_j^{(n)}$	:	Mean of the j^{th} mixture component estimated at time n
$\mathbf{u}_{jk}^{(n)}$	:	k^{th} eigenvector of the j^{th} mixture component estimated at time n
$\mathbf{U}_j^{(n)}$	:	Matrix with P columns of $\mathbf{u}_{jk}^{(n)}$, $k=1\sim P$
$\hat{\mathbf{x}}_{ij}$	:	Reconstruction of $\mathbf{x}_i$ with mixture component j
$\hat{\mathbf{X}}_i$	:	Matrix with M columns of $\hat{\mathbf{x}}_{ij}$, $j=1\sim M$
w_{ij}	:	Weight of $\hat{\mathbf{x}}_{ij}$ to reconstruct $\mathbf{x}_i$
$\mathbf{w}_i$	:	Vector with M entries of w_{ij}
α	:	Decay factor, $0<\alpha<1$
q,r	:	Index for the mixture component

The reconstruction errors contributed by previous data are weighted by powers of the decay factor α. At any time instant n, the solution to this minimization problem is obtained by iteratively determining the means, sets of eigenvectors, and weights, respectively while fixing the other parameters. That is, we optimize the means using the previous sets of eigenvectors and weights. After updating the means, we optimize the sets of eigenvectors and weights accordingly. The next iteration starts again in updating the means and so on. The iterations are repeated until the parameters converge. Note that the initial condition of each parameter at any time instant n is calculated with the weighted combination of the optimized parameter of the previous time instant $n-1$ and the new data $\mathbf{x}_n$. The detail is as follows.

The mean $\mathbf{m}_q^{(n)}$ of mixture component q at time n is:

$$\mathbf{m}_q^{(n)} = \left(1-\frac{w_{nq}^2}{\sum_{i=0}^{\infty}\alpha^i w_{n-i,q}^2}\right)\mathbf{m}_q^{(n-1)} + \left(\frac{w_{nq}}{\sum_{i=0}^{\infty}\alpha^i w_{n-i,q}^2}\right)\left(\mathbf{x}_n - \sum_{j=1,j\neq q}^{M} w_{nj}\hat{\mathbf{x}}_{nj}\right) \quad (2)$$

We can see that $\mathbf{m}_q^{(n)}$ is obtained from the previous estimator $\mathbf{m}_q^{(n-1)}$ and the current input $\mathbf{x}_n$. The second term of (2) is constructed by how much the other component $\hat{\mathbf{x}}_{ij}$, $j\neq q$ cannot represent the data $\mathbf{x}_n$ with some scalar multiplication in front. The covariance matrix $\mathbf{C}_r^{(n)}$ of mixture component r at time n is:

$$\mathbf{C}_r^{(n)} = \alpha\mathbf{C}_r^{(n-1)} + (1-\alpha)\begin{bmatrix} w_{nr}\left[(\mathbf{x}_n-\mathbf{m}_r)\mathbf{x}_n^T + \mathbf{x}_n(\mathbf{x}_n-\mathbf{m}_r)^T\right] - \\ \sum_{j=1}^{M} w_{nj}w_{nr}\left[(\mathbf{x}_n-\mathbf{m}_r)\mathbf{m}_j^T + \mathbf{m}_j(\mathbf{x}_n-\mathbf{m}_r)^T\right] - \\ \sum_{j=1,j\neq r}^{M} w_{nj}w_{nr}\sum_{k=1}^{P}\left[\mathbf{u}_{jk}^T(\mathbf{x}_n-\mathbf{m}_j)\right]\left[(\mathbf{x}_n-\mathbf{m}_r)\mathbf{u}_{jk}^T + \mathbf{u}_{jk}(\mathbf{x}_n-\mathbf{m}_r)^T\right] - \\ w_{nr}^2(\mathbf{x}_n-\mathbf{m}_r)(\mathbf{x}_n-\mathbf{m}_r)^T \end{bmatrix} \quad (3)$$

Again, $\mathbf{C}_r^{(n)}$ is obtained from the previous estimator $\mathbf{C}_r^{(n-1)}$ and the current input $\mathbf{x}_n$. To complete one iteration of updating for means, covariance matrices and weights, the solution for weights is:

$$\begin{bmatrix} 2\hat{\mathbf{X}}_i^T\hat{\mathbf{X}}_i & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix}\begin{bmatrix} \mathbf{w}_i \\ \lambda \end{bmatrix} = \begin{bmatrix} 2\hat{\mathbf{X}}_i^T\mathbf{x}_i \\ 1 \end{bmatrix} \quad (4)$$

where $\mathbf{1}=[1 \ \cdots \ 1]^T$ is an $M\times 1$ vector. We see that both MPC and UPC are special cases of UMPC with $\alpha\rightarrow 1$ and $M=1$ respectively.

3. ERROR CONCEALMENT WITH UMPC

With object-based video coding standards such as MPEG-4 [4], the region of interest (ROI) information is available. A model based error concealment approach can use such ROI information and build a better error concealment mechanism. Figure 3 shows two video frames with ROI specified. In this case, ROI can also be obtained by face trackers such as [5].

(a) (b)

Figure 3 Two video frames with object specified

When the video decoder receives a video frame with error free ROI, it can use the data in ROI to update the existing UMPC with the processes described in Section 2. In this paper, all available error free ROI are used to update the UMPC. Less frequent update to reduce the computational complexity is possible at the expense of less adaptivity. In the experiment, the time consumed on an Intel Pentium III 650 PC to update the UMPC model, with three mixture components and two eigenvectors each, is about 10 seconds per ROI. Practical system design can consider updating the UMPC model with the incoming error free ROI when the error to represent this ROI with the current UMPC model is larger than a threshold. As to reconstructing the corrupted ROI with the UMPC model, the time consumed is almost negligible with 20 ms per ROI.

When the video decoder receives a frame of video with corrupted macroblocks (MB) in the ROI, it uses UMPC to reconstruct the corrupted ROI. In Figure 4, we use three mixture components: 1st, 2nd, and 3rd, to illustrate the idea of UMPC for error concealment. The choice of number of mixture components is based on empirical results. For example, if we choose face region as the ROI, three mixture components approximately represent left, center, and right modalities of a human face. Optimal choice of number of mixture components can be found by examining the representation performance of the UMPC model versus the number of mixture components.

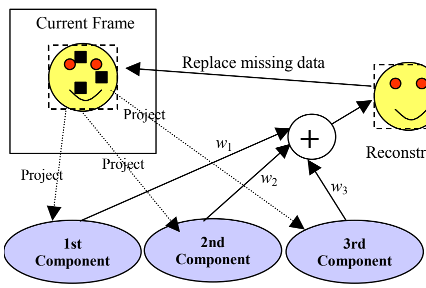


Figure 4. UMPC for error concealment

The corrupted ROI is first reconstructed by each individual mixture component. The resulting reconstructed ROI is formed by linearly combining the three individually reconstructed ROI. The weights for linear combination are inverse proportional to the reconstruction error of each individually reconstructed ROI. After the reconstructed ROI with UMPC is done, the corrupted MB is replaced with the corresponding data in the reconstructed ROI just obtained. The process of reconstruction with UMPC and replacement of corrupted MB is repeated iteratively until the final reconstruction result is satisfactory.

4. EXPERIMENT

The test video sequence is recorded from a TV program [6]. The video codec used in this paper is the ITU-T H. 263 standard [7]. Some frames of this video sequence are shown in Figure 3.

We use a two state Markov chain [8] to simulate the bursty error to corrupt the MB as shown in Figure 5. "Good" and "Bad" correspond to error free and erroneous states respectively. The overall error rate ε is related to the transition probabilities p and q by $\varepsilon = p/(p+q)$. We use $\varepsilon = 0.05$ and $p = 0.01$ in the experiment.

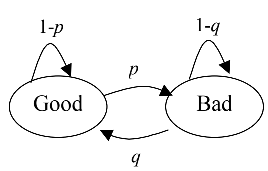


Figure 5. Two state Markov chain for MB error simulation

There are two sets of experiments: Intra and Inter. In the Intra coded scenario, we compare three cases: (1) *none*: no error concealment takes place. When the MB is corrupted, the MB content is lost; (2) *MPC*: error concealment with MPC as the model. The number of mixture components M is three and the number of eigenvectors P for each mixture components is two; (3) *UMPC*: error concealment with UMPC as the model with $M = 3$ and $P = 2$. The decay factor α is 0.9. In the Inter coded scenario, we also compare three cases: (1) *MC*: error concealment using motion compensation; (2) *MPC*: error concealment with MPC as the model operated on motion compensated data; (3) *UMPC*: error concealment with UMPC as the model operated on motion compensated data.

In updating the model at each time instant, we iterate the model for five times. A smaller number of iteration can be used for a faster speed. In the error concealment stage when an erroneous MB is received, five iterations of POCS, with either MPC or UMPC model, are performed.

Figure 6 and Figure 7 show the means and eigenvectors of UMPC at two different time instances. They show that the model captures three main poses of the face images. The eigenvectors associated with each mean face refine the faces at each component with expressions in mouths, eyebrows, etc.

While Figure 6 shows UMPC after training upon receiving 20 frames, Figure 7 shows UMPC after training upon receiving 60 frames. Since there is a change of characters, UMPC model captures such change and we can see that the means and eigenvectors in Figure 7 describe more about the second character.

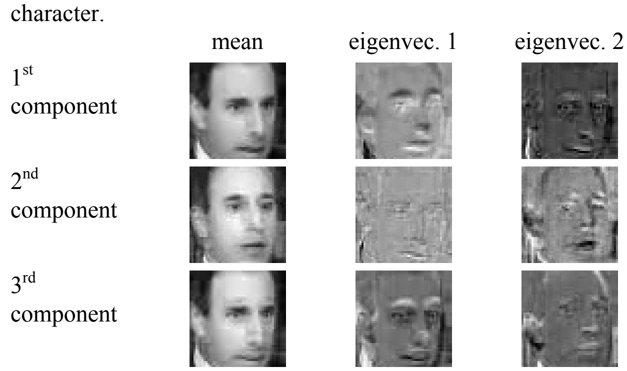


Figure 6. Means and eigenvectors for UMPC at Frame 20

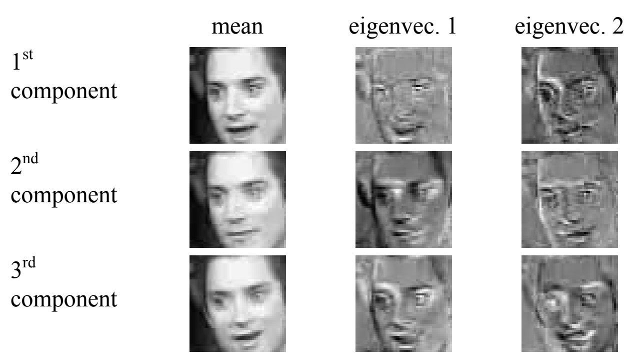
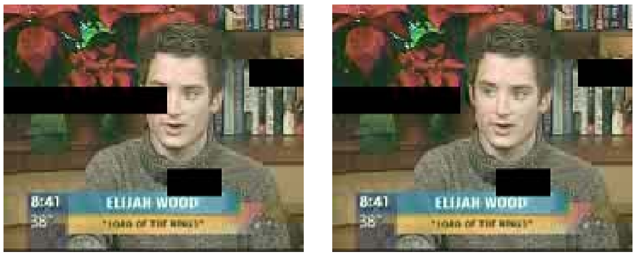


Figure 7. Means and eigenvectors for UMPC at Frame 60

Figure 8 and Figure 9 show the decoded video frames without and with the error concealment. Figure 8 (a) shows a complete loss of MB content when the MB data is lost. Figure 8 (b) shows that the decoder successfully recovers the MB content with the corrupted ROI projected onto the UMPC model. Figure

9 (a) shows the MB content being recovered by motion compensation when the MB data is lost. The face is blocky because of the error in motion compensation. Figure 9 (b) shows that the decoder successfully recovers the MB content inside the ROI with the motion compensated ROI projected onto the UMPC model. Note that in this paper, we focus on performing error concealment to the ROI, which is the face region, only. The peak signal to noise ratio (PSNR) is evaluated in the ROI.



(a) (b)

Figure 8 Error concealment for the Intra coding scenario with: (a) no concealment; (b) concealment with UMPC

(a) (b)

Figure 9. Error concealment for the Inter coding scenario with: (a) motion compensation; (b) motion compensation and UMPC

The overall PSNR performance of the decoded video frames is summarized in Table 1. In both the Intra and Inter scenarios, error concealment with UMPC performs the best.

Table 1 Error concealment performance of four models at INTRA and INTER coded scenarios

	None (Intra) /MC (Inter)	MPC	UMPC
Intra	15.5519 dB	29.3563 dB	30.6657 dB
Inter	21.4007 dB	21.7276 dB	22.3484 dB

The frame-by-frame PSNR performances from Frame 20 to 60 are shown in Figure 10 and Figure 11, with Intra and Inter scenarios respectively. We can see from Figure 10 that the performance curve of UMPC sits on top of all the others. Error concealment with UMPC on the average performs the best in the Inter scenario as shown in Figure 11. While playing the reconstructed video sequences of all three methods in the Inter scenario, we see that the video quality suffers from error propagation.

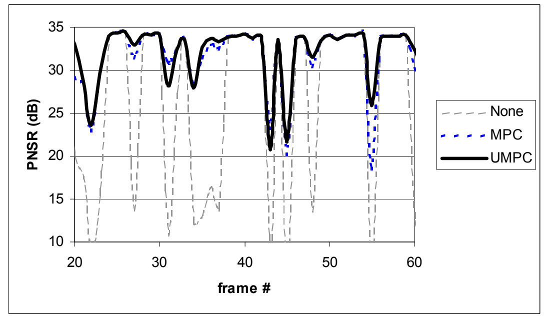


Figure 10. Error concealment results: INTRA

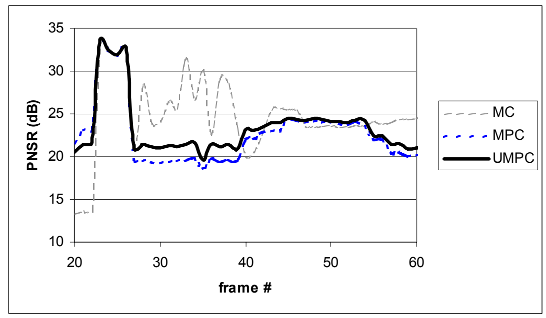


Figure 11. Error concealment results: INTER

5. CONCLUSION

We proposed a new statistical model called UMPC to represent the non-stationary and multi-modal video data. The error concealment result with UMPC had better performance than conventional approaches with MPC and motion compensation only. In the future, we will consider including error statistics to improve the performance. In addition, we want to improve the error concealment in the Inter scenario.

6. REFERENCES

[1] H. Sub and W. Kwok, “Concealment of Damaged Block Transform Coded Images using Projections Onto Convex Sets”, *IEEE Trans. Image Processing*, Vol. 4, pp. 470-477, April 1995.

[2] D. S. Turaga, Ph.D. Thesis “Statistical Modeling for Networked Video: Coding Optimization, Error Concealment, and Traffic Analysis”, Carnegie Mellon University, July 2001.

[3] X. Liu and T. Chen, "Shot Boundary Detection Using Temporal Statistics Modeling", to appear in *ICASSP 2002*.

[4] Motion Pictures Experts Group, "Overview of the MPEG-4 Standard", ISO/IEC JTC1/SC29/WG11 N2459, 1998.

[5] J. Huang and T. Chen, "Tracking of Multiple Faces for Human-Computer Interfaces and Virtual Environments", *ICME 2000*, New York.

[6] NBC Today Show on Dec. 12th, 2001.

[7] ITU-T Recommendation H.263, January 27, 1998.

[8] M. Yajnik, S. Moon, J. Kurose, D. Towsley, “Measurement and modeling of the temporal dependence in packet loss,” *INFOCOM 1999*, pp. 345-52, March 1999.