Noname manuscript No.
(will be inserted by the editor)

Error Concealment via Particle Filter by Gaussian Mixture Modeling of Motion Vectors for H.264/AVC

Ali Radmehr · Abdorasoul Ghasemi

Received: / Accepted:

Abstract We use particle filtering for correcting the erroneous Motion Vectors (MVs) which are derived from the Boundary Matching Algorithm (BMA) in packet video communications. Assuming a two-state Markov channel model for transmission, the error of the extracted MVs by BMA is shown to be modeled by the Gaussian Mixture (GM) distribution. Formulating the problem in the state-space, we deploy particle filtering for denoising the erroneous MVs. The main challenge of using particle filters is high computational complexity that is directly related to the number of particles. The proposed particle filtering scheme is efficient even if the number of particles is decreased. Experimental results are provided to show the efficiency of this filtering approach compared to a recent scheme based on Kalman filtering. The experiments show meaningful increase in the quality of the recovered video sequences in terms of PSNR up to 3 dB compared with the other error concealment (EC) techniques. Also, the computational complexity of the proposed scheme is discussed.

Keywords video error concealment · particle filtering · non-linear filtering · Mont Carlo (MC) filtering · non-Gaussian filtering

1 Introduction

In recent years, the growth of wireless technologies has led to the popularity of video services on mobile devices. Unfortunately, the wireless channels are

Ali Radmehr
Faculty of Electrical and Computer Engineering, K. N. Toosi University of Technology, Tehran, Iran

Abdorasoul Ghasemi
Faculty of Electrical and Computer Engineering, K. N. Toosi University of Technology, Tehran, Iran

unreliable for video signal transmission, especially for compressed videos. Advanced video coding standards such as H.263, and H.264 are highly compressed and applicable in interactive communications which is mostly due to the recent motion compensation techniques. For example A. Almuhit *et al.* proposed a novel elastic motion model based on elastic image registration with 2-D cosine basis function [1]. Due to the high compression rate, a few errors in the video bit stream may lead to severe problems in decoding process. On the other hand, the motion compensated prediction which is used for increasing the compression rate of video data causes the errors to be propagated through the subsequent frames at the decoder side.

To alleviate this problem, error resilience and error concealment (EC) techniques have been proposed. The former makes the coded videos more indestructible against transmission errors while the latter attempts to conceal and recover the erroneous parts of the frames deploying the correctly received information [2, 3].

Several strategies have been proposed for error resilient techniques in [2, 4]. One approach is using feedback mechanism. This can be achieved for example by automatic retransmission request (ARQ) which uses the rate distortion of the transmission line function to select the inter or intra-coding mode [5]. Although these methods are efficient ways of protecting the video data from channel errors, they add more delay to the data decoding time which makes it unusable for many interactive applications.

Another approach is using the forward error correction (FEC) method. In this method, the decoder can recover the errors by using the redundant information, for example the error correction or other types of codes, which are inserted among the compressed video signals at the encoder side [6, 7, 8, 9]. Although error resilience techniques make video codecs more robust, they decrease the efficiency of the video coding and increase the bit rate of the transmitted video which eventually limit its applications.

EC techniques attempt to recover erroneous parts of a video frame using the correctly received data at the decoder side. These techniques have made use of the spatially and/or temporally adjacent Macroblocks' (MBs) pixels information at the decoder side regardless of making any changes in the channel or source codes [10, 11]. In spatial methods, the neighboring pixels are used to interpolate the missed coefficients in the same erroneous frame. On the other hand, temporal methods exploit the correlation between consecutive frames. The challenging part in temporal approaches is to recover motion vectors (MVs) of the erroneous MBs. Therefore, various schemes are proposed to recover the MVs. For instance they can be estimated by zero MV technique, easily, *i.e.,* by replacing the erroneous MB with the one at the same position in previous frame. Moreover, the MV of the erroneous MBs can be recovered by more sophisticated techniques which are discussed in [12, 13, 14]. For example, Hrusovsky *et al.* proposed an algorithm using statistical distribution of the corrupted motion information. The distribution is derived from the damaged video sequence and then the MVs are refined deploying a Particle Filter based approach [15]. Also the erroneous MB can be recovered using Boundary Match-

ing Algorithm (BMA). To enhance this technique several methods have been proposed. Authors in [16] proposed a novel correlation based enhancement algorithm to improve the accuracy of the MVs which are derived using BMA. Another highly considered approach to this problem is developed by Chen *et al.* in [17]. The authors in the mentioned paper have proposed a BMA-based algorithm using spatio-temporal correlation (STBMA) and found the best replacing MB in the reference frame. Then they have used a PDE-based method to refine the lost pixels. In order to apply this stage they have minimized the weighted difference between the gradient of the lost MB and the recovered one which is derived from the STBMA method in the first stage. Their method reduce the blocking artifact and recover the missing MB effectively. However due to the the high dependency to spatio-temporal correlation their proposed method will be less effective in sequences with high non-uniform MVs. In this paper we propose a novel particle filter based method which exploits a non-linear non-Gaussian model that can deal with non-uniform systems effectively.

Also several filtering techniques are proposed to enhance the recovered MVs. For example in [18] authors try to use Kalman filter after MV restoration. They use edge strength and intensity reliability to restore the pixel values. In addition, there are also other sophisticated algorithms which can estimate the erroneous MV in a better way. For instance, Lie and Gao [19] proposed a dynamic-programming-based technique to evaluate the performance of boundary matching and side smoothness of recovered MB which is enhanced with adaptive Kalman filtering algorithm. More recently other algorithms are introduced using both temporal and spatial methods together [20, 21]. For instance, an autoregressive model is developed by Yongbing Zhang *et al.* for EC in block-based packet video coding which uses both spatial and temporal information [20].

In this paper we develop a novel Particle Filter (PF) based algorithm to conceal the missing MB. First, the estimation of the MVs of the corrupted MB is obtained using BMA. Next, a non-Gaussian/non-linear model is developed which is used for deploying the filtering algorithm. However, the high complexity of PF limits its application. Therefore, we consider several constraints in the proposed filtering method to balance the accuracy of the recovered MVs and the required computational complexity. The rest of this paper is organized as follows. In section II, filtering scheme for EC is presented. Section III, is devoted to the problem statement and notations. In section IV, we investigate usage of particle filtering for MV recovery. In section V, the proposed algorithm is discussed and several practical considerations are investigated. Section VI is devoted to the simulation results that are compared with similar schemes in terms of peak signal-to-noise ratio (PSNR) and processing time. Finally in section VII, we make the concluding remarks.

2 Filtering Schemes for Error Concealment

The video data packets are transmitted through slices in MPEG-x or GOBs in H.26x standard. Sometimes, packets are lost or corrupted which causes the decoder to decode MBs erroneously. One of the major problems in concealment of these erroneous MBs is finding the most suitable MVs for them by using the correctly received data at the decoder side. If one estimates these MVs correctly, the erroneous MBs can be replaced by the closest ones from the previous frame. A well-known technique which is used to estimate these MVs by using the information of the previous frame, is BMA. In this method, the MV of the corrupted MB is restored based on the MV information at the nearest neighbors of the erroneous MB in the current frame [22]. This information can help us to find the best displacement of the erroneous MB in the previous frame. It leads to finding an estimated MV for the corrupted MB. This scheme is investigated further in section III.

In some cases, the channel error leads to severe degradation in several consecutive MBs. Hence the erroneous parts of the frames are concealed using one of the EC techniques. To mitigate this problem, filtering techniques are introduced to modify these MVs and make them more appropriate for the recovery of the erroneous MBs. The objective is to iteratively exploit the MVs of the adjacent MBs in the current frame and their corresponding information in the previous frames. Then, by applying an estimation technique, the filtering scheme finds the best matching MV for the erroneous MB. According to our knowledge, video EC techniques based on Kalman Filter (KF) are widely used in literature and different approaches are proposed to obtain the best EC method. A brief overview of the EC based on KF and its severe problems are investigated in section II, further.

2.1 Kalman Filtering for MV Recovery

In [23] KF is applied on the extracted MVs of the adjacent MBs in the current frame and previous ones which are derived from BMA for fine tuning. Note that abbreviation and variables used throughout this paper are written in table I and table II, respectively.

Let $\mathbf{X}_t$ and $\mathbf{X}_{t-1}$ denote the MVs of the erroneous MB in the t^{th} and $(t-1)^{th}$ estimation states, respectively. Assume $\mathbf{Y}_t$ denotes the observed MV of the erroneous MB in the t_{th} estimation state. Also, $\mathbf{V}_t$ and $\mathbf{U}_t$ denote the observation noise and process noise, respectively. The state and observation models are given by (1) and (2), respectively.

$$\mathbf{X}_t = \mathbf{A}\mathbf{X}_{t-1} + \mathbf{U}_t \tag{1}$$

$$\mathbf{Y}_t = \mathbf{B}\mathbf{X}_t + \mathbf{V}_t \tag{2}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the state transition and measurement matrices, respectively.

These matrices are determined by using the correctly received MVs and the Least Square Error (LSE) method in each frame [23]. Then, the observed noisy MV of the erroneous MB is applied in (1) and (2) to find the best representative MV as the state.

Experimental results show that the quality improvement by about $0.5 \sim 0.72$ dB is achieved in PSNR using this filtering technique [23].

2.2 Motivation for Particle Filtering

The two major assumptions of Kalman filtering are linearity and Gaussianity. That is the dynamic state and observation model in (1) and (2) are linear. $\mathbf{A}$ and $\mathbf{B}$ are estimated for each frame specifically. When a frame consists of the objects with different motion speeds and directions, there is inaccuracy in MV recovery since these matrices are estimated using LSE which is inaccurate.

To illustrate the problem visually, we consider two different frames for the 14^{th} and 54^{th} frames of the "Suzie" video sequence. Assuming a given channel error model in Fig. 1(a) to (d) the erroneous and the corresponding reconstructed frames by deploying Kalman filtering are depicted. The results show good performance for MV recovery on the 14^{th} frame than the 54^{th} one since it has more uniform MVs. Also Fig. 2(a) and (b) shows the recovered MVs of the 14^{th} and 54^{th} frames, respectively. While the MVs of the 14^{th} frame are more uniform and almost zero in most parts of the frame, the MVs of the 54^{th} frame have different directions and amplitudes. Therefore, $\mathbf{A}$ in (1) is better estimated for the 14^{th} frame since the error of LSE in estimation is smaller. In fact, the frame that has more uniform MVs will have smaller error in estimating $\mathbf{A}$ and leads to better MV approximation.

This discussion shows that assuming linearity for this system is not always accurate and cause severe degradation in video quality especially when the MVs of the erroneous MBs are not uniform in a frame. On the other hand, Gaussianity is another limitation of Kalman filtering, *i.e.,* it cannot handle non-Gaussian observation and process noises which can not be modeled by a Gaussian PDF.

In Kalman filtering, the observation noise of the extracted MVs from the BMA method is assumed to have Gaussian distribution. However, in following we argue that the observation noise of the BMA is better modeled by a GM distribution.

We set up an experiment in which the MVs for two different sequences are extracted by the BMA method as noisy measurements. Let δ_x and δ_y be the error values of the recovered MVs using BMA in x and y directions, respectively. The estimation of errors Probability Density Functions (PDF) is derived based on the normal kernel function and the density is measured at 100 equally spaced points that cover the range of δ_x and δ_y.

In Figs. 3 and 4, the PDF for "Carphone" and "Suzie" sequences are depicted, respectively. In this experiment, for "Suzie" sequence, the Packet Loss

Rate (PLR) is 15% and the Quantization Parameter (QP) is set to 20 and for "Carphone" sequence the PLR and QP is 5 and 25, respectively.

The PDFs which are shown in both of these two figures are obviously non-Gaussian. We consider a GM model as an alternative. Therefore, these PDFs are better approximated by a mixture of Gaussians.

Figs. 5 and 6 show the errors between the d-component GM approximated PDFs and the original one in the sense of Sum of Square Error (SSE) where d is the number of Gaussian components in GM model. These figures show that the SSE is decreased when the number of GM components are increased. That is, the system observation noise is more exactly modeled by the GM distribution. In other words, increasing d leads to better modeling of the system at the cost of adding more complexity to it which makes it less efficient in interactive communications.

Since particle filtering scheme is not restricted to linear and Gaussian systems, we are motivated to use this technique for video error concealment and MV recovery.

3 Problem Statement and Notations

A MV is considered as a pair of two independent elements in x and y directions. The MV for $(m,n)^{th}$ MB in x and y directions is denoted by $x_{k,x}(m,n)$ and $x_{k,y}(m,n)$ in k^{th} frame, respectively. Let $I_{i,j}(m,n,k)$ be the $(i,j)^{th}$ pixel at $(m,n)^{th}$ MB in k^{th} frame where $i \in \{0,1,2,\ldots,15\}$ and $j \in \{0,1,2,\ldots,15\}$.

The packet network channel can be modeled by a Markovian two-state Elliot-Gilbert model [24]. As it is depicted in Fig. 7, when the channel is in bad state the packet is received erroneously and the decoder considers it as lost. This model is run for each packet and the average time the model stays at the bad state is considered as PLR. The PLR for this model is given by

$$PLR = \frac{a}{a+b} \tag{3}$$

where a and b are the channel state transition probabilities. When PLR increases, the number of erroneous slices increases and the video quality degrades drastically.

A new dynamic system model is proposed to find better MVs for concealing the errors. We use this model to enhance the MVs by filtering. In the state-space the system model can be formulated by

$$\mathbf{X}_t^{new} = h_t(\mathbf{X}_t^{old}, \mathbf{U}_t) \tag{4}$$

$$\mathbf{Y}_t = g_t(\mathbf{X}_t^{new}, \mathbf{V}_t) \tag{5}$$

where $\mathbf{X}_t^{old}$ is the prior state vector of the MV of the considered MB and $\mathbf{U}_t$ is the process noise vector which can be non-Gaussian. h_t is a function which can be non-linear and varies for each frame. Also g_t is the observation function and it can be non-linear, too. $\mathbf{V}_t$ is the observation noise vector. For

the sake of simplicity $\mathbf{X}_t^{new}$ is denoted as $\mathbf{X}_t$ and $\mathbf{X}_t^{old}$ is denoted as $\mathbf{X}_{1:t-1}$. Therefore, (4) and (5) can be simplified to (6) and (7).

$$\mathbf{X}_t = h(\mathbf{X}_{1:t-1}, \mathbf{U}_t) \tag{6}$$
$$\mathbf{Y}_t = g(\mathbf{X}_t, \mathbf{V}_t) \tag{7}$$

The block diagram of the proposed filtering scheme for error concealment is depicted in Fig. 8. As it is shown in this figure, $\mathbf{X}_{1:t-1}$ is the prior states of $\mathbf{X}_t$, that is, it can be derived through $t-1$ previous states. In order to develop a probabilistic model, two ideas are deployed. First, we propose the set $\mathcal{S}$ such that its components are chosen from the MVs of the nearest adjacent neighbors of corrupted MB. It is added as an additional step before applying the filtering and is given by

$$\mathcal{S} = \{x^{i,j}(m,n) | i \in [-1,1], j \in [-1,1], (i,j) \neq (0,0)\} \tag{8}$$

where $x^{i,j}(m,n)$ is the MV that is located in $(m+i, n+j)^{th}$ position and $x^{0,0}$ is the MV of the corrupted MB in the middle of these 8 neighboring MBs. The components of $\mathcal{S}$ are considered as the initial states of $\mathbf{X}_t$. To decrease the complexity the set $\mathcal{S}$ should be made before deploying the particle filter. It is depicted in Fig. 8. Note that the number of components of $\mathcal{S}$ depends on the region of the frame where the erroneous MB is located and the measure of simplicity that one will achieve. If the neighboring MVs are lost in the current frame, one can exploit the adjacent MVs at the same place in the previous frame. Second, we assume a GM PDF for the system in order to develop an efficient system model and particle filters can deal with GM models effectively. It is beneficial especially when the system is non-linear and non-Gaussian. In order to benefit from this scheme, a d-component GM posterior probability for the current state $\mathbf{X}_t$ is considered in (9).

$$p(\mathbf{X}_t \mid \mathbf{X}_{1:t-1}) = \sum_{l \in d} \lambda_l N(\mu_l, \sigma_l) \tag{9}$$

where σ_l and μ_l are mean and variance of the l^{th} GM distribution, respectively, and λ_l is its mixing density. We assume that mixing densities are equal and h is a function that selects a component from the set $\mathcal{S}$, randomly, hence, (6) can be simplified to (10).

$$\mathbf{X}_t = h(\mathbf{X}_{1:t-1}) + \mathbf{U}_t \tag{10}$$

Note that the particle filters have the ability to correct both prediction and measurement noises, efficiently. In other words, the selected component by h will be predicted and corrected iteratively using the filtering technique and the error will be minimized in several iterations [25, 26].

The observation in the proposed scheme is obtained through the BMA method. Its function is denoted by $g(.)$ in (7). As a matter of fact, it locates the best displacement of the $(m,n)^{th}$ MB in the target frame through the reference

frame. The best displacement is achieved when the sum of squared difference between the surrounding top, bottom, and left lines of the neighboring MBs is minimized. Let MV_x and MV_y denote the components of the recovered MV by the BMA method which is given in (11).

$$(MV_x, MV_y) = (i' - i, j' - j) \tag{11}$$

where i and j are the pixel indexes of the considered MB and i' and j' are displaced i and j. As it is discussed in section II, these MVs are considered as noisy observations since the error values for MV_x and MV_y are unpredictable. Considering $\mathbf{V}_t$ as the GM distributed observation noise vector (7) can be simplified to (12).

$$\mathbf{Y}_t = g(I_{i',j'}(m, n, k-1), I_{i,j}(m, n, k), \mathbf{X}_t) + \mathbf{V}_t \tag{12}$$

where $I_{i',j'}(m, n, k)$ and $I_{i,j}(m, n, k)$ are pixel information for $(m, n)^{th}$ MB.

4 Particle Filtering for MV Recovery

4.1 Basics of Particle Filtering

Particle filters can estimate the states of non-linear/non-Gaussian systems effectively. It is a Bayesian sequential importance sampling method which recursively approximates the posterior distribution [27, 28]. Suppose the system is modeled by a Markov process which its initial distribution probability is denoted by $p(x_0)$ and has a transition equation which is given by $p(x_t \mid x_{1:t-1})$. Also, the hidden states of the system are denoted by x_t where $t \in \{1, 2, \ldots, T\}$ and $x_t \in \mathcal{X}$ where $\mathcal{X}$ is the set of all system states and T is the number of the states. The observations are denoted by y_t where $t \in \{1, 2, \ldots, T^*\}$ and T^* is the number of observations and $y_t \in \mathcal{Y}$ where $\mathcal{Y}$ is the set of all the observations which are assumed to be conditionally independent. Given observations $y_{1:t-1} = \{y_1, \ldots, y_{t-1}\}$ up to time $t-1$, the posterior distribution at time t can be derived using (13)

$$p(x_t \mid y_{1:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid y_{1:t-1}) dx_{t-1} \tag{13}$$

Since observation y_t is available at time t and considering Markov model, the state can be updated using Bayes' rule which is written in (14)

$$p(x_t \mid y_{1:t}) = \frac{p(y_t \mid x_t) p(x_t \mid y_{1:t-1})}{\int p(y_t \mid x_t) p(x_t \mid y_{1:t-1}) dx_t} \tag{14}$$

Equations (13) and (14) are prediction and updating steps for particle filtering, respectively.

Since it is usually impossible to sample efficiently from the posterior distribution $p(x_t \mid y_{1:t})$ at any time t, alternative methods have been proposed in order to relieve this problem [28]. On helpful technique to solve this problem

is called Importance Sampling (IS). However, this method is not applicable for recursive estimation which is explained specifically in [27,28,29,30]. Therefore, we use another technique which is called Sequential Importance Sampling (SIS) [27]. This technique is a recursive version of IS and can approximate the posterior distribution $p(x_t \mid y_{0:t})$ by a weighted set of particles which are denoted by $w_t^{(i)}$. In other words, the weight of the i^{th} particle in t^{th} state is denoted by $\{w_t^{(i)}, x_t^{(i)}\}_{i=1}^{N}$ for N particles which is calculated by (15) recursively.

$$\tilde{w}_t^{(i)} \propto \tilde{w}_{t-1}^{(i)} \frac{p(y_t \mid x_t^{(i)}) p(x_t^{(i)} \mid x_{t-1}^{(i)})}{\pi(x_t^{(i)} \mid x_{0:t-1}^{(i)}, y_{1:t})} \tag{15}$$

where $\tilde{w}_t^{(i)}$ is the normalized importance weight for the i^{th} particle and $\pi(x_t^{(i)} \mid x_{0:t-1}^{(i)}, y_{1:t})$ is the importance density at time t. By assuming(16), the equation (15) can be simplified to (17).

$$\pi(x_t^{(i)} \mid x_{0:t-1}^{(i)}, y_{1:t}) = p(x_t^{(i)} \mid x_{t-1}^{(i)}) \tag{16}$$

$$\tilde{w}_t^{(i)} \propto \tilde{w}_{t-1}^{(i)} p(y_t \mid x_t^{(i)}) \tag{17}$$

Obviously, one can evaluate importance weights for each particle by (17), recursively. These weights require the particles to be propagated through time before they are being calculated. This choice is easily implemented and updated using the measurement likelihood $p(y_t \mid x_t^{(i)})$ for the sampled particles. In other words if one initializes the particles' weight, *i.e.,* $w_0^{(i)}$, then $w_t^{(i)}$ can be updated using $p(y_t \mid x_t^{(i)})$, accordingly.

SIS is an attractive method. Unfortunately, it does not guarantee to converge as time proceeds and the distribution of the normalized weights $\tilde{w}_t^{(i)}$ would be more distorted as t is increasing. Therefore, only one particle has a non-zero weight when time proceeds and the algorithm fails to represent the posterior distributions, sufficiently [28]. In other words, it is not practical for recursive estimation of the hidden states because the complexity of this method is increasing with time due to estimating the posterior distribution, *i.e.,* $p(x_{0:t} \mid y_{1:t})$, that have nearly zero weights. An appropriate measurement to solve the degeneracy problem is using the effective sample size N_{eff} which is given in (18).

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N} \tilde{w}_t^{(i)}} \tag{18}$$

when $\hat{N}_{eff}$ is below a fixed threshold N_{thres}, the particles are resampled [27]. To overcome this problem two ideas are applied:

1)Eliminating particles that have small normalized importance weights.

2)Multiplying particles which have large weights.

By adding these ideas, a resampling step will be added to the SIS method which is used for obtaining a new set of particles and weights [31]. Also this

method is modular, very quick and easy to implement. One important property of these filters is that they can be implemented on the hardware and parallel computers which can make them useful for more interactive applications [27, 28, 32].

4.2 Application of Particle Filtering for MV Recovery

Let $\mathbf{X}_t^{(i)}(m,n)$ be the MV of the i^{th} particle at the current state, *i.e.,* t, for $(m,n)^{th}$ MB at the decoder side where (m,n) is the coordinate in terms of (row, column). The state vector model is given by (19). It consists of x and y components of the considered MV which are assumed to be independent.

$$\mathbf{X}_t = (x_{t,x}^{(i)}, x_{t,y}^{(i)}) \tag{19}$$

The diagram in Fig. 9 illustrates the proposed method. Using (10) and (19), the update equation is formulated in (20).

$$\begin{aligned} x_{t,x}^{(i)}(m,n) &= x_{t-1,x}^{(i)}(m,n) + u_{t,x} \\ x_{t,y}^{(i)}(m,n) &= x_{t-1,y}^{(i)}(m,n) + u_{t,x} \end{aligned} \tag{20}$$

where $u_{t,x}$ and $u_{t,y}$ are the process noise terms in x and y directions, respectively. They are supposed to have GM distribution with zero mean where their covariance values are set to be equal.

For modeling the observations, we consider (12) as the observation model. Therefore, $\mathbf{Y}_t$, *i.e.,* the observed value, can be derived for t^{th} state. Horizontal and vertical components of the observed MV are denoted by $[y_{t,x}^{(i)}(m,n), y_{t,y}^{(i)}(m,n)]$ in x and y directions for i^{th} particle at the t^{th} state.

The particles' weights are calculated based on the similarity between $\mathbf{X}_t$ and $\mathbf{Y}_t$ using the Euclidian distance between them which are formulated in (21) and (22)

$$f(x_{t,x}^{(i)}, y_{t,x}) = [\sum_{i=1}^{N}(x_{t,x}^{(i)} - y_{t,x})^2]^{\frac{1}{2}} \tag{21}$$

$$f(x_{t,y}^{(i)}, y_{t,y}) = [\sum_{i=1}^{N}(x_{t,y}^{(i)} - y_{t,y})^2]^{\frac{1}{2}} \tag{22}$$

The measurement likelihood of the particles are achieved by

$$p(y_{t,x} \mid x_{t,x}^{(i)}) = N(f(x_{t,x}^{(i)}, y_{t,x})^2, 0, 1) \tag{23}$$

$$p(y_{t,y} \mid x_{t,y}^{(i)}) = N(f(x_{t,y}^{(i)}, y_{t,y})^2, 0, 1) \tag{24}$$

Therefore the particles' weight are calculated by

$$w_{t,x}^{(i)} = w_{t-1,x}^{(i)} \times p(y_{t,x} \mid x_t^{(i)}) \tag{25}$$

$$w_{t,y}^{(i)} = w_{t-1,y}^{(i)} \times p(y_{t,y} \mid x_t^{(i)}) \tag{26}$$

After the particles' weight derived using (25) and (26) one can easily estimate the parameters of the state by taking expectation of the state with respect to their weights by

$$\hat{x}_{t,x} = E\{x_{t,x}\} \tag{27}$$

$$\hat{x}_{t,y} = E\{x_{t,y}\} \tag{28}$$

Also in order to avoid the degeneracy problem, which is discussed in section 4.1, N_{eff} is calculated and if it is below N_{thres}, new states and corresponding weights are generated using Algorithm 2.

5 The Proposed Scheme and the Algorithm

5.1 The Implementation of the Proposed Scheme

The particle filtering procedure is described briefly in this section. For the sake of simplicity, we denote $x_{t,x}^{(i)}(m,n)$ or $x_{t,y}^{(i)}(m,n)$ as $x_t^{(i)}$ and $w_{t,x}^{(i)}(m,n)$ or $w_{t,y}^{(i)}(m,n)$ as $w_t^{(i)}$. Also it is assumed that the p^{th} MB is missed in the current erroneous slice. The algorithm has to be performed on each corrupted MB within an erroneous slice. We explain the proposed particle-filter-based error concealment method in the following steps which are depicted in Fig. 9.

Step 1) Error Detection: The corrupted slices/GOBs at the decoder side are identified.

Step 2) Build the set $\mathcal{S}$ by using (8)

Step 3) Initialize with N random particles that are drawn from the set $\mathcal{S}$ where its elements are uniformly distributed and set the corresponding weights equal, *i.e*, $w_0^{(i)} = \frac{1}{N}$

Step 4) Obtain the observation using BMA

Step 5) Particle filtering is applied by Algorithm 1 for p^{th} MB in the erroneous slice and the candidate MV, *i.e*, $\hat{x}_t$, is estimated for t^{th} state.

Step 6) Resample the particles of the t^{th} state if $N_{eff} < N_{thres}$ by using Algorithm 2.

The pseudo-code of these steps are illustrated in Algorithm 1 where M is the number of erroneous MBs in the erroneous slice and N is the number of the particles. e is the index of the detected erroneous MB and i is the index of the particle. The resampling step is added to the Algorithm 1 which is illustrated in the Algorithm 2. In this algorithm $W_t^{(i)}$ is Cumulative Distribution Function (CDF) and $j(i)$ is the resampled particles.

In the pseudo-code for the resampling step, $\tilde{x}_t^i$ is the particles' state before resampling. Also $j(i)$ is the resampled index [28, 32].

Algorithm 1 Particle-filter-based Error concealment algorithm

for $e = 1$ to M **do**
  build the set $\mathcal{S}$
  for $i = 1$ to N **do**
    draw x_0^i from $\mathcal{S}$
    calculate $w_0^i = 1/N$
  end for
  for $i = 1$ to N **do**
    predict next state of the particles by $x_{t+1}^{(i)} = x_t^{(i)} + u_t$
    evaluate the particles likelihood by $p(y_t \mid x_t^{(i)})$ using (23) and (24)
    calculate the weights by $w_t^{(i)} = w_{t-1}^{(i)} \times p(y_t \mid x_t^{(i)})$
    calculate the summation of all weights by $S = \sum_{i=0}^{N} w_t^{(i)}$
    normalize all the weights by $\tilde{w}_t^{(i)} = \frac{w_t^{(i)}}{S}$
    estimate the MV's state $\hat{x} = E\{x_t\}$
    apply resampling step by using Algorithm 2
  end for
end for

Algorithm 2 Resampling

calculate $\hat{N}_{eff}$ by $\hat{N}_{eff} = \frac{1}{\sum_{i=0}^{N} \tilde{w}_t^i}$
if $\hat{N}_{eff} \geq N_{thres}$ **then**
  set $x_t^i = \tilde{x}_t^i$
else
  calculate the cumulative distribution function (cdf) $W_t^{(i)}$ of the weight $w_t^{(i)}$
  for $i = 1$ to N **do**
    draw n independent uniforms $\{U^i\}_{1 \leq i \leq n}$ on the interval (0,1).
    look for the minimum $j(i)$ that satisfies $W_t^{j(i)} \geq n$.
  end for
end if
for $i = 1$ to N **do**
  Set $x_t^i = \tilde{x}_t^{j(i)}$
  Set $w_t^i = \frac{1}{N}$
end for

5.2 Practical Considerations

Computational complexity is an important issue in using particle filters. This problem arises drastically when the number of particles increases. To address this issue, let the number of particles be denoted by N and the number of the erroneous MBs in a frame by M. Also, we consider n elements in the set $\mathcal{S}$. Since there are two simultaneous calculations for the considered MVs in x and y directions, we calculate the computations for one direction, then it is doubled to achieve all of the calculations.

In the initialization phase, N samples are drawn from a uniform distribution from the set $\mathcal{S}$. In particle filtering, there are four parts. Prediction part needs 1 additions which is repeated N times. In evaluation part there are 2 additions and one Gaussian PDF calculation for finding the likelihood which is repeated N times. There is N divisions in normalization part. If resampling is used in filtering, there are N multiplications and N additions and one division which is for finding N_{eff}. Also resampling part needs N table lookups to draw from the uniform distribution and $N \times N$ Comparisons. There is a possibility that resampling algorithm can be implemented in $O(N)$ operations by using the techniques which are mentioned in [27]. The main reason for low number of particles is the novel idea of using the set $\mathcal{S}$. In other words, choosing the initial state for PF from the set $\mathcal{S}$ leads to a faster convergence and a significant reduction in complexity [28].

6 Experimental Results

The proposed method is evaluated using the standard H.264/AVC codec and examined on CIF(352×288 pixels) and QCIF (176×144 pixels) video sequences which contained 150 frames. The simulations were run on a Pentium-4, 2 GHz core 2 duo CPU personal computer with Microsoft Windows 7 operating system. The experiments are performed using MATLAB software. It is coded at 30 frame/sec and the coding structure is "IPPP". Note that the Quantization Parameter (QP) is 25 and kept consistent along the whole frames. We have tested the method on different kinds of standard frames such as "Stefan", "Tennis", "Suzie", "Container", "Foreman" and "Carphone". Based on extensive simulations we found that 3-component mixture model leads to the best subjective and objective results and hence d sets 3 in the whole experiments. The PSNR is calculated between two video sequences by (29)

$$PSNR = 10\log_{10}(\frac{B^2}{MSE}) \tag{29}$$

where B is the largest possible value of the signal which is 255 in our experiment and MSE is the Mean Square Error difference between two frames. We calculate it for Y component of each frame and denote it as YPSNR in decibel unit.

For comparison several typical EC methods are implemented. The first one is BMA. In this method the erroneous MB is replace with the most similar one in the previous frame [22]. Another EC algorithms are the DMVE [33] and the proposed particle-filter-based algorithm which is evaluated with one of the most recent Kalman filter based EC methods [34]. This method is denoted as "KF". Also the proposed algorithm is denoted as "PF".

Table III shows the average YPSNR for 150 frames of the "Stefan", "Foreman", "Tennis" and "Carphone" sequences. In this test YPSNR and the needed time to process the considered EC method are evaluated. The particle number is 100 in this test. The needed time using the PF and the KF is more than BMA since they are post processing applied to the BMA. The proposed algorithm shows better performance than the other methods. Detailed results are given in this table. The KF method degrades the quality of the "Stefan" sequence for some PLR values since KF suffers from linearity and Gaussianity constraints which are discussed in section II. These results show the superiority of the proposed algorithm than other considered methods.

In Fig. 10, YPSNR is evaluated for each frame of the "Foreman" sequence. This sequence has QCIF resolution (176×144) and the PLR is 20% in this experiment. The proposed algorithm seems to be superior to the other methods in most of the frames which is reflected in YPSNR values.

The results in Fig. 11 show better subjective quality for the proposed techniques than other methods. It is tested on 4^{th} frame of the "Stefan" sequence with QCIF resolution and the background text, fans, texture and the edges are restored better in (f).

Fig. 12 compares other techniques versus the proposed one. It is tested on 2^{nd} frame of the "Foreman" sequence with CIF resolution. The displaying concealment is superior in most of the erroneous parts include the hat, lips and most parts of the face.

Figs. 13 and 14 illustrate the comparison among BMA, DMVE, KF and the proposed algorithm for two other types of error patterns. The PLR is assumed to be 25%. Fig. 13 is 10^{th} frame of the "Stefan" sequence with QCIF resolution. The proposed concealment technique recovers the missing MBs with less blocking artifacts and the straight lines are restored better. Fig. 14 is 7^{th} frame of the "Foreman" sequence with QCIF resolution. The recovered frame obtain better visual quality specifically around the nose and ear parts of the face.

To summarized this paper, we have highlighted the main points of the proposed technique. We are motivated to use particle filtering as a fine tuning process to overcome the inefficiency of the BMA algorithm which is due to the lack of information for MV recovery. Also one of the Kalman filtering technique is investigated and its inefficiency due to linearity and Gaussianity constraints is discussed in detail. In the proposed scheme, a new system model is developed which is non-linear and non-Gaussian. In addition error PDF of the BMA is investigated and a novel GM model is developed. This model is used for particle filtering and the problem of high complexity is solved using the set $\mathcal{S}$ which is useful for the algorithm to converge faster. Unfortunately

MVs in H.264/AVC do not always recovered correctly and the post processing techniques seem to be necessary.

7 Conclusion and Discussion

In this paper, we have deployed particle filtering algorithm for concealing the erroneous MBs which are received in the decoder side. An efficient system model and importance weight function were developed to maintain an effective particle-filter-based framework for erroneous MV recovery. Also the update formulation is proposed based on the GM and several new parameters are used to develop this novel particle-filter-based EC technique. Experiments have shown that using particle filters for MV recovery in the way which is discussed in this paper can increase the quality of videos up to about 3 dB in YPSNR. Besides, this EC scheme can open up a new path for using filtering techniques in video EC. For further investigation we recommend using other optimization algorithms to enhance the accuracy of the particle filters. It has a great potential in obtaining a high performance for video EC. The results have shown that the proposed particle filtering method is a useful tool for recovering the erroneous MBs effectively.

References

1. A. AlMuhit, M. R. Pickering, M. R. Frater, J. F. Arnold, Video Coding using Elastic Motion Model and Larger Blocks, IEEE Trans. Circ. And Syst. for Video Technology 20 661-672 (2010).
2. Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, Error resilient video coding techniques, IEEE Signal Process. Mag. 17 61-82 (2000).
3. Y. Wang, Q. F. Zhu, Error control and concealment for video communication: A review, proc. IEEE 86 974-997 (1998).
4. V. DeBrunner, L. DeBrunner, L. Wang, S. Radhakrishnan, Error control and concealment for image transmission, IEEE Commun. Surveys 3 9991010 (2000).
5. R. Zhang, S. L. Regunathan, K. Rose, Video coding with optimal inter/intra-mode switching for packet loss resilience, IEEE J. Sel. Areas Commun. 18 966-976 (2000).
6. S. B. Wicker, Error Control Systems for Digital Communication and Storage, Prentice Hall, 1995.
7. J. Hagenauer, T. Stockhammer, Channel coding and transmission Aspects for wireless multimedia, Proc. IEEE 87 1764-1777 (1999).
8. C. S. Kim, S. U. Lee, Multiple description coding of motion fields for robust video transmission, IEEE Trans. Circuits Syst. Video Technol. 11 999-1010 (2001).
9. K. Goyal, Multiple description coding: compression meets the network, IEEE Signal Process. Mag. 18 74-93 (2001).
10. B. W. Wah, X. Su, D. Lin, A survey of error-concealment schemes for real-time audio and video transmissions over the internet, Proc.Int.Symp. Multimedia Software Engineering, 17-24 (2000).
11. P. Cuenca, L. Orozco-Barbosa, A. Garrido, F. Quiles, T. Olivares, A survey of error concealment schemes for MPEG-2 video communications over ATM networks, Proc. IEEE Conf. Elect. and Comput. Eng. 1 25-28 (1997).
12. S. Tsekeridou, F. A. Cheikh, M.Gabbouj, I. Pitas, Motion field estimation by vector rational interpolation for error concealment purposes, Proc. IEEE Int.Conf. Acoust. Speech, Signal Process. 6 3397-3400 (1999).

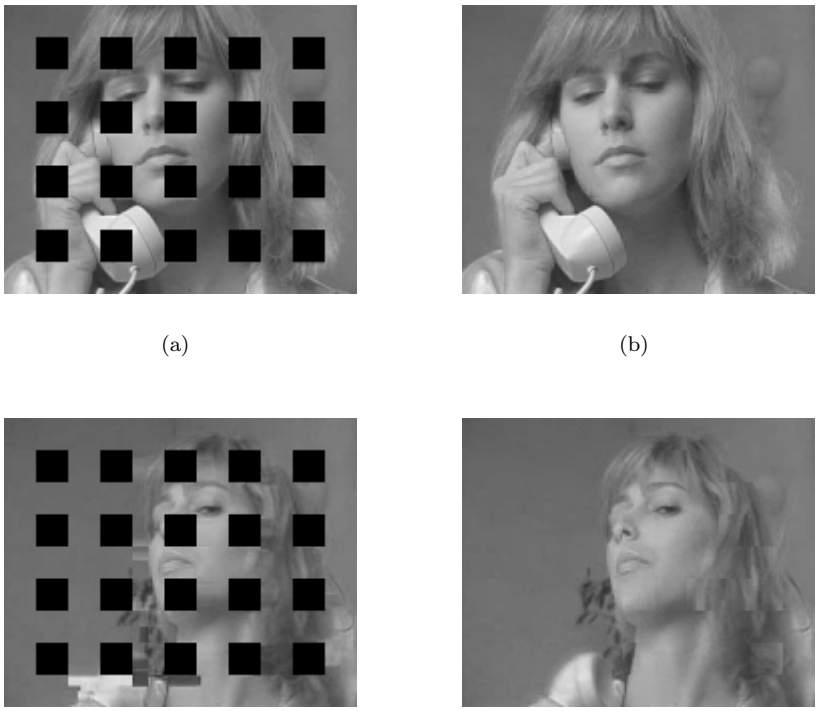
13. M. Al-Mualla, N. Canagarajahm, D. R. Bull, Error concealment using motion field interpolation, Proc. IEEE Int. Conf. Image Process. 3 512-516 (1998).
14. J. Zhang, J. F. Arnold, M. R. Frater, M. R. Pickering, Video error concealment using decoder motion vector estimation, Proc. IEEE Int. Conf. Speech Image Technol. Comput. Telecommun. 2 777-780 (1999).
15. B. Hrusovsky, J. Mochnac, S. Marchevsky, Error Concealment Method Based on Motion Vector Prediction Using Particle Filters, RADIOENGINEERING, 20(3) 692-702 (2011)
16. A. Radmehr, A. Ghasemi, A novel video error concealment technique using modified boundary matching algorithm with correlation function, 21st Iranian Conference on Electrical Engineering (ICEE) 2013.
17. Yan Chen, Yang Hu, Oscar C. Au, Houqiang Li, Chang Wen Chen, Video error concealment using spatio-temporal boundary matching and partial differential equation, IEEE trans. on multimedia 10 2-15 (2008).
18. S. Takahashi, T. Ogawa, H. Tanaka, M.Haseyama, Kalman filter-based error concealment for video transmission, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences 28 779787 (2009).
19. W. N. Lie, Z. W. Gao, Video error concealment by integrating greedy suboptimization and Kalman filtering techniques, IEEE Trans.Circuits Syst. Video Technol. 16 982-992 (2006).
20. J. Seiler, A. Kaup, Reusing the H.264/AVC Deblocking Filter for Efficient Spatio-Temporal Prediction in Video Coding, IEEE Int.Conf. on Acoustics, Speech, and Signal Processing (ICASSP) 1049-1052 (2011).
21. Yongbing Zhang, Xinguang Xiang, Debin Zhao, Siwe Ma, Wen Gao, Packet Video Error Concealment With Auto Regressive Model, IEEE trans. on circuits and systems for video technology 22 12-27 (2012).
22. W. M. Lam, A. R. Reibman, and B. Liu, Recovery of lost or erroneously received motion vectors, Proc.IEEE Int. Conf. Acoustics, Speech, Signal Process. 3 417-420 (1993).
23. Z. W. Gao, W. N. Lie, Video error concealment by using Kalman filtering technique, Proc. IEEE Int. Symp. Circuits Syst. 2 26-29 (2004).
24. M. Ghanbari, Video coding: An introduction to standard codecs, IEE Press, 1999.
25. G. Kitagawa, Monte Carlo filter and smoother for non-Gaussian non-linear state space models, J. Comp. Graph. Stat. 5 1-25 (1996).
26. Jarad Niemi, Mike West, Adaptive Mixture Modeling Metropolis Methods for Bayesian Analysis of Nonlinear State-Space Models, Journal of Computational and Graphical Statistics, 19 260-280 (2010).
27. Arnaud Doucet, Simon Godsill, Christofe Andrieu, On Sequential Mont Carlo Sampling methods for Bayesian filtering, Cluwer Academic Publisher: Statistics and computing, 197-208 (2000).
28. Arnaud Doucet, Nando de Freitas, Neil Gordon, Sequential Monte Carlo Methods in Practice (Statistics for Engineering and Information Science), Springer, 2001.
29. J.V. Candy, Bootstrap particle filtering, IEEE Signal Processing Magazine 24 73-85 (2007).
30. M. S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Trans. on Signal Processing 2 1-15 (2002).
31. O. Cappe, S. J. Godsill, E. Moulines, An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo, Proc. IEEE 95 899-924 (2007).
32. Miodrag Bolic, Petar M. Djuric, Sangjin Hong, Resampling Algorithms for Particle Filters: A Computational Complexity Perspective, EURASIP Journal on Applied Signal Processing 1 2267-2277 (2004).
33. J. Zhang, J. F. Arnold, M. R. Frater, A cell-loss concealment technique for MPEG-2 coded video, IEEE trans. on circuits and systems for video technology 10 659-665 (2000).
34. Shihua Cui, Huijuan Cui, Kun Tang, Error concealment via Kalman filter for heavily corrupted videos in H.264/AVC, Signal Processing:Image Communication 28 430-440 (2013).

Table 1 ABBREVIATIONS USED THROUGHOUT THIS PAPER

abbreviation	definition
EC	Error Concealment
ARQ	Automating Retransmission Request
FEC	Forward Error Correction
MB	Macroblock
MV	Motion Vector
KF	Kalman Filter
DMVE	Decoder Motion Vector Estimation
PSNR	Peak Signal to Noise Ratio
GOB	Group Of Blocks
LSE	Least Square Error
PDF	Probability Density Function
SSE	Sum of Square Error
QP	Quantization Parameter
PLR	Packet Loss Rate
IS	Importance Sampling
SIS	Sequential Importance Sampling
MC	Monte Carlo
MSE	Mean Square Error
CDF	Cumulative Distribution Function
BMA	Boundary Matching Algorithm
GM	Gaussian Mixture

Table 2 VARIABLES USED THROUGHOUT THIS PAPER

variables	definition
X_t	t^{th} motion vector state
t	current state
A	state transition matrix
B	measurement matrix
U_t	process noise for t^{th} state
V_t	observation noise for t^{th} state
Y_t	observed motion vector for t^{th} state
δ_x	error value of the x component of the recovered motion vector using BMA in k^{th} frame
δ_y	error value of the y component of the recovered motion vector using BMA in k^{th} frame
$x_{k,x}(m,n)$	The x component of the motion vector for $(m,n)^{th}$ macroblock
$x_{k,y}(m,n)$	The y component of the motion vector for $(m,n)^{th}$ macroblock
$I_{i,j}(m,n,k)$	$(i,j)^{th}$ pixel value at $(m,n)^{th}$ macroblock in k^{th} frame
X_t^{new}	current state vector of the motion vector
X_t^{old}	prior state vector of the motion vector
$\hat{X}_t$	estimated motion vector for t^{th} state
σ_l	variance of the l^{th} Gaussian component of the Gaussian mixture
μ_l	mean of the l^{th} Gaussian component of the Gaussian mixture
λ_l	mixing density
MV_x	x component of the recovered motion vector using BMA
MV_y	y component of the recovered motion vector using BMA
m	x index of the considered macroblock
n	y index of the considered macroblock
$i(i')$	displacement value of the macroblock in x direction
$j(j')$	displacement value of the macroblock in y direction
k	frame number
T	number of states
T^*	number of observations
N	number of particles
$\pi(.)$	Importance density
$w_t^{(i)}$	weight of i^{th} particle in the t^{th} state
$\tilde{w}_t^{(i)}$	normalized weight of the i^{th} particle in the t^{th} state
$\hat{N}_{eff}$	effective sample size
$f(.)$	normal PDF function
p	current MB in the considered erroneous slice



(a) (b)

(c) (d)

Fig. 1 (a) The 14th erroneous frame from "Suzie" sequence. (b) The 14th frame is reconstructed by Kalman filtering (c) The 54th erroneous frame from "Suzie" sequence. (d) The 54th frame is reconstructed by Kalman filtering. (Notice that it is shown in (d), the 54th frame is reconstructed poorly since it has more nonuniform motion speed in different areas of the frame.)

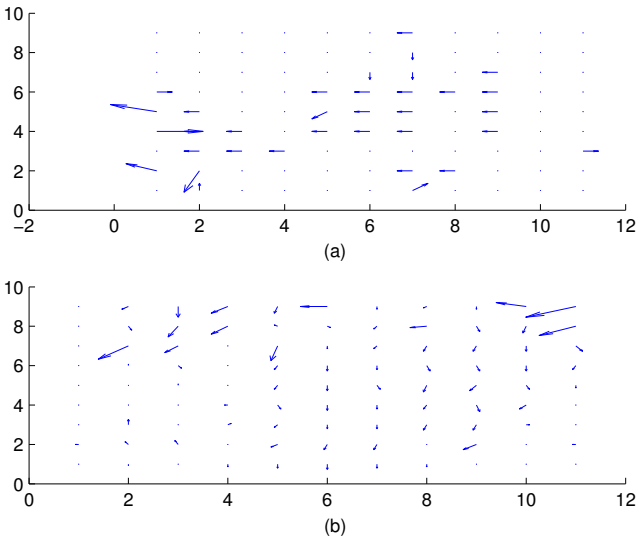


Fig. 2 (a) MV diagram for the 14th frame of "Suzie" video sequence. The MVs are more uniform than 54th frame, averagely. The objects' motion speed are almost the same. (b) MV diagram for 54th frame of "Suzie" video sequence. The MVs are very different from each other. The speed and direction of MVs are alternated in different areas of this frame extremely which leads to inaccuracy in MV recovery by Kalman filtering.

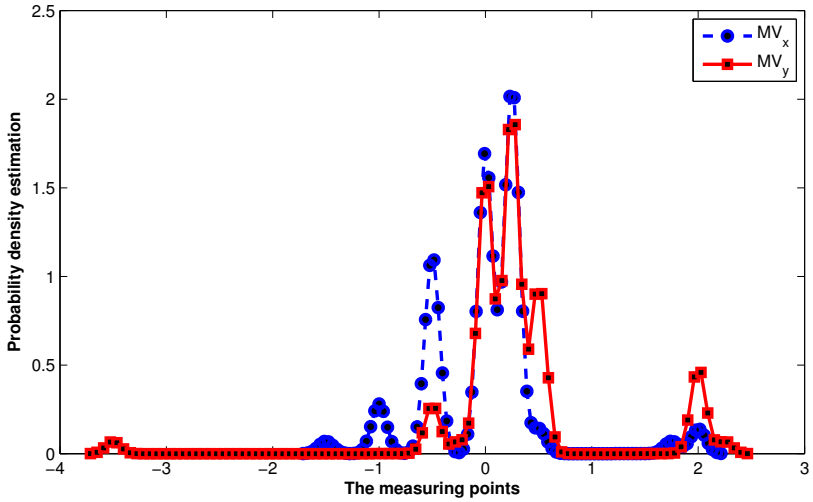


Fig. 3 The PDFs of the "Carphone" sequence for both x and y elements of the corrupted MVs are depicted when PLR=5% and QP=25. They show the non-Gaussian behavior, clearly.

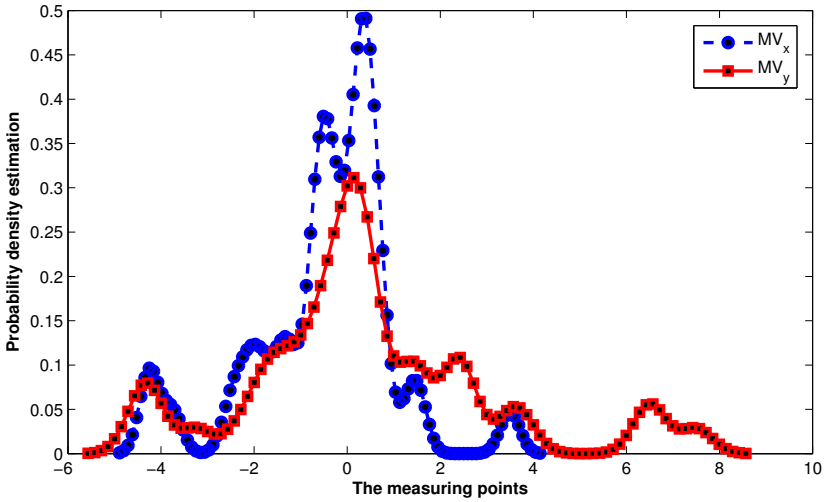


Fig. 4 The PDFs of the "Suzie" sequence for both x and y elements of the corrupted MVs are depicted when PLR=15% and QP=20. They show the non-Gaussian behavior, clearly.

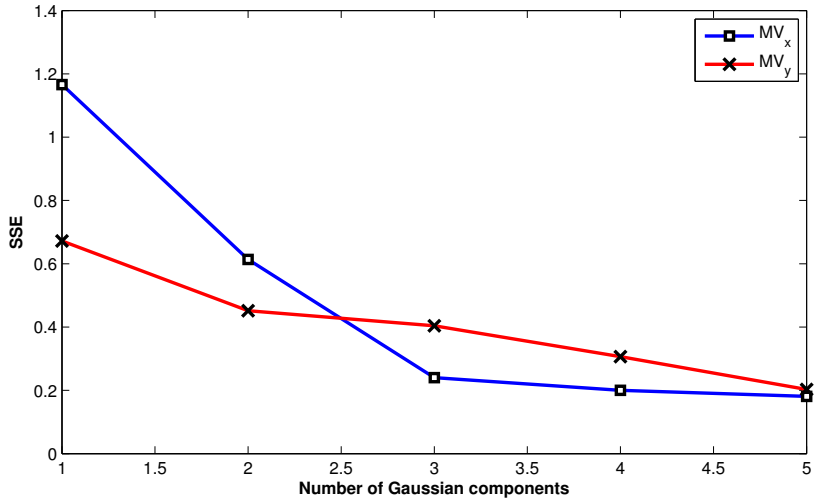


Fig. 5 This diagram shows the error values when the PDF is approximated by GMs with various number of components. The "Carphone" sequence is tested with PLR=5% and QP=25. It is obvious that when we increase the number of components, the SSE is decreased dramatically.

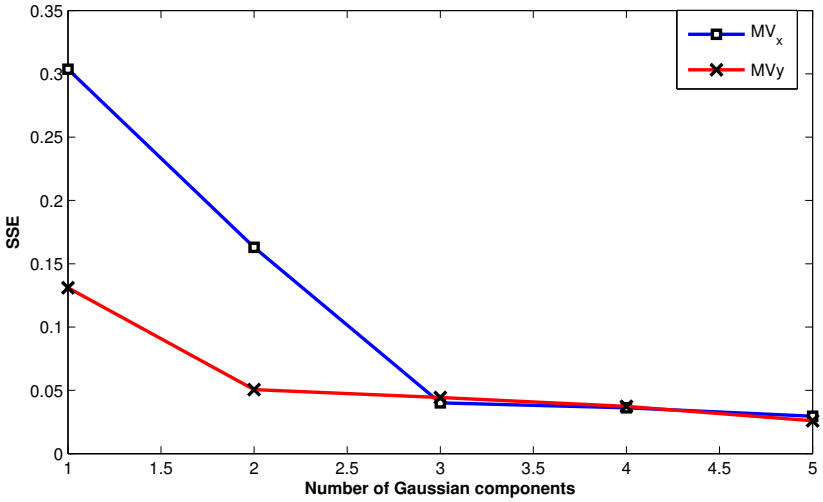


Fig. 6 This diagram shows the error values when the PDF is approximated by GMs with various number of components. The "Suzie" sequence is tested with PLR=15% and QP=20. It is obvious that when we increase the number of components, the SSE is decreased dramatically.

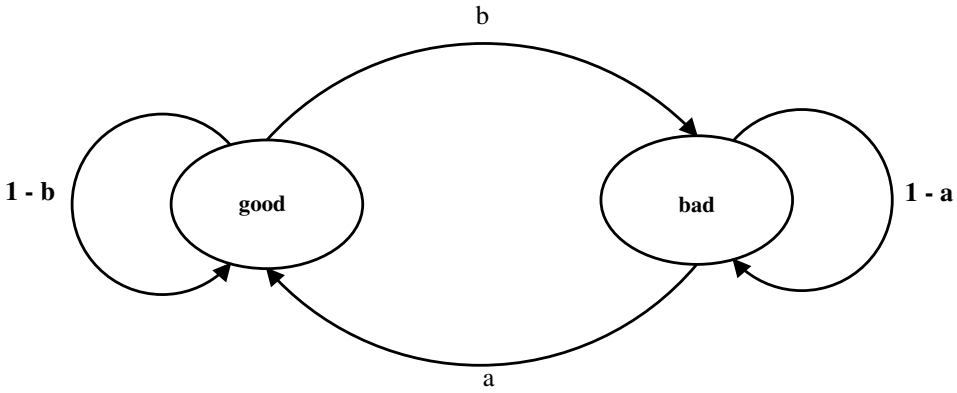


Fig. 7 The discrete two-state Elliot-Gilbert channel model. If the transmission channel were in the bad state, the data received in the decoder would be decoded incorrectly.

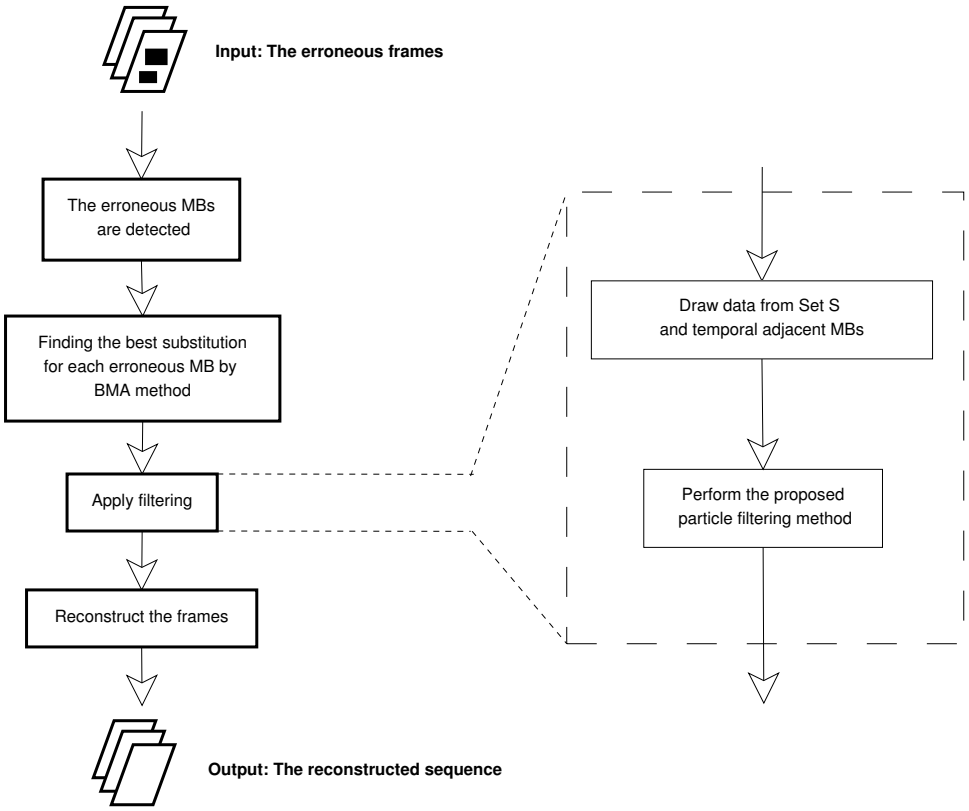


Fig. 8 The summery of the proposed method in brief. The filtering methods are applied for fine tuning after MV estimation which is recovered using the BMA method. The set $\mathcal{S}$ is derived from (8).

Table 3 Performance evaluation of the proposed algorithm versus other EC methods

Sequence	Measurement	DMVE			BMA			KF			PF		
PLR(%)		5	10	20	5	10	20	5	10	20	5	10	20
Stefan	YPSNR(dB)	29.34	25.64	25.49	29.10	25.62	24.77	29.55	25.97	25.61	29.64	26.63	26.13
	TIME(s)	3.09	6.22	12.48	0.08	0.17	0.34	0.09	0.17	0.34	0.12	0.20	0.46
Foreman	YPSNR(dB)	33.71	33.41	30.98	33.20	32.5	30.43	35.69	33.57	31.25	36.40	35.38	32.49
	TIME(s)	2.87	5.70	11.59	0.08	0.15	0.31	0.09	0.17	0.34	0.11	0.23	0.46
Tennis	YPSNR(dB)	36.14	35.03	31.64	35.21	34.84	31.11	36.55	35.22	32.45	37.90	37.20	33.50
	TIME(s)	2.8	5.69	11.44	0.08	0.15	0.31	0.09	0.20	0.33	0.12	0.23	0.46
Carphone	YPSNR(dB)	36.10	34.32	31.94	34.79	33.20	31.12	36.52	35.18	34.07	37.99	36.36	34.98
	TIME(s)	2.02	5.83	11.50	0.07	0.17	0.36	0.09	0.15	0.31	0.11	0.22	0.45

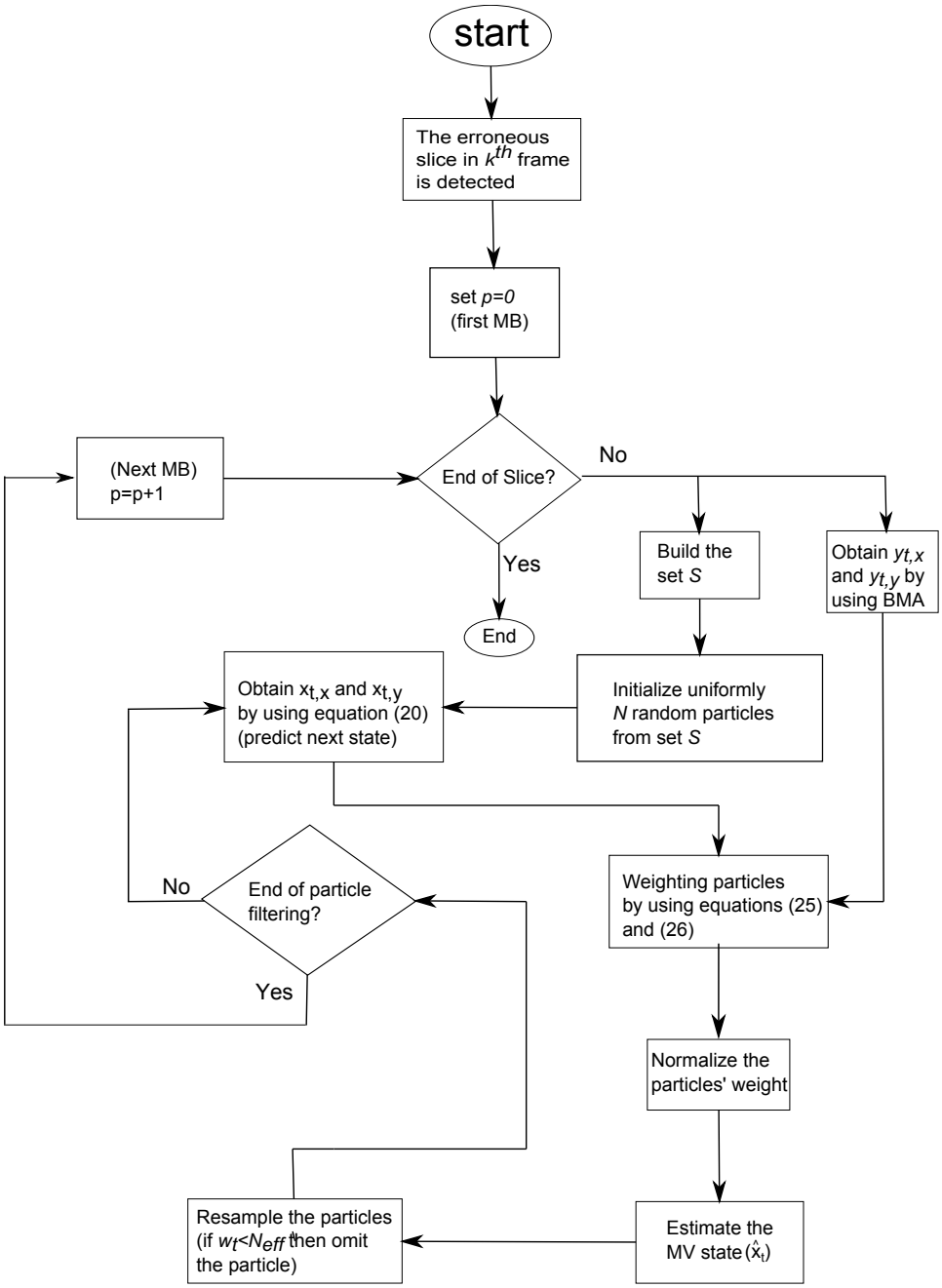


Fig. 9 It illustrates the proposed method briefly. p^{th} MB in the erroneous slice is chosen. The observation is obtained for the erroneous MB using BMA. After initializing the particles, the next state is predicted and particles are weighted according to the observation. Then particles are resampled and the MV's state is predicted. This method is applied for each erroneous slice in each frame.

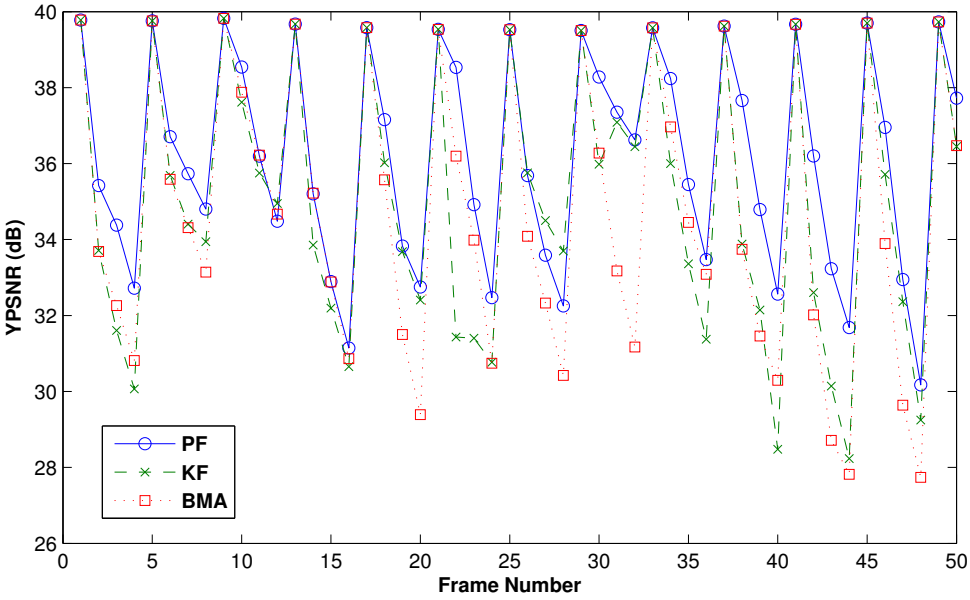
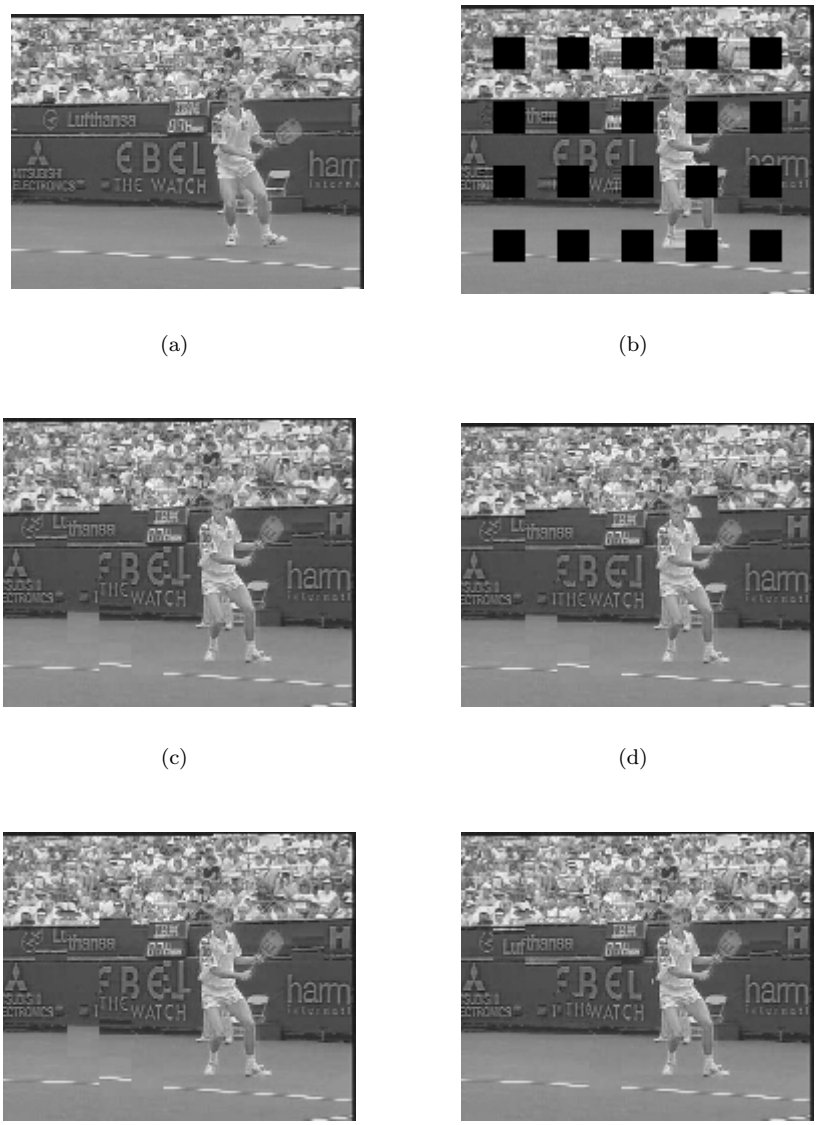


Fig. 10 YPSNR versus frame number for "Foreman" sequence. This sequence has QCIF resolution (176 × 144). QP is set to 22 and PLR is 15%.



(a) (b)

(c) (d)

(e) (f)

Fig. 11 The 4th frame of the "Stefan" sequence in QCIF resolution:(a) error-free frame (b) corrupted frame (c) concealed with DMVE (d) concealed with BMA (e) concealed with KF (f) concealed with the proposed method (PF)

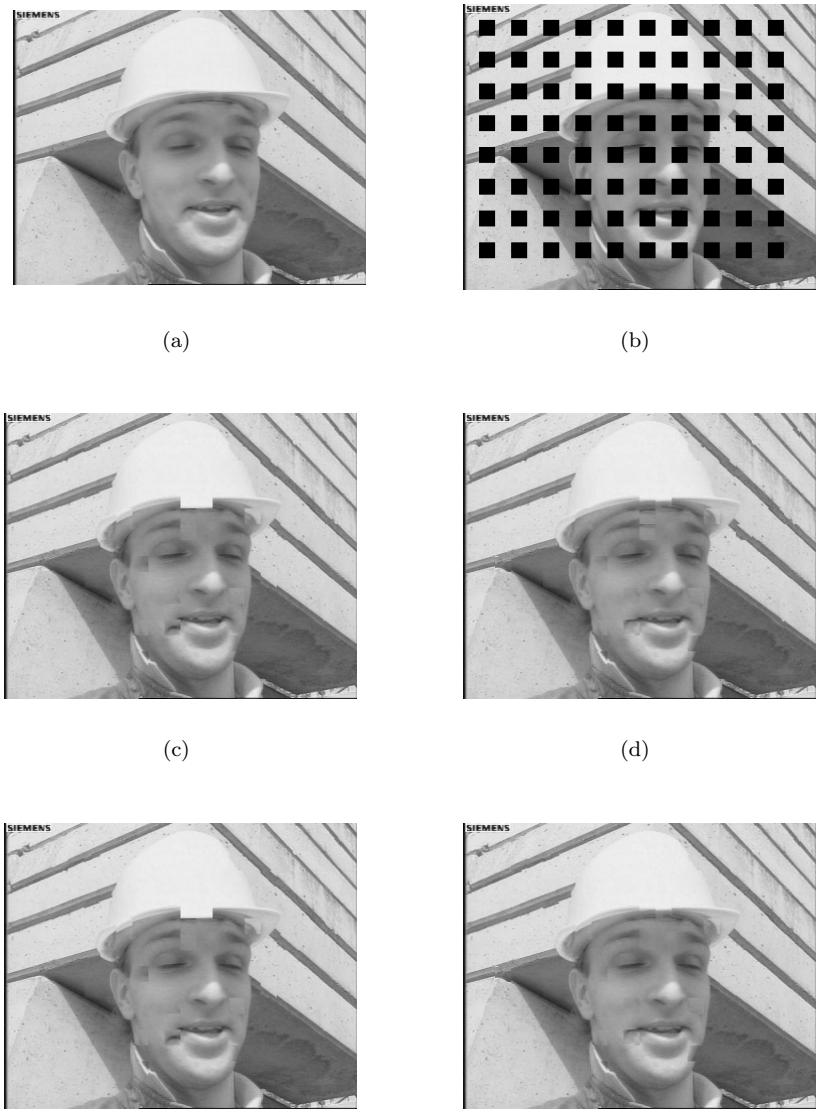
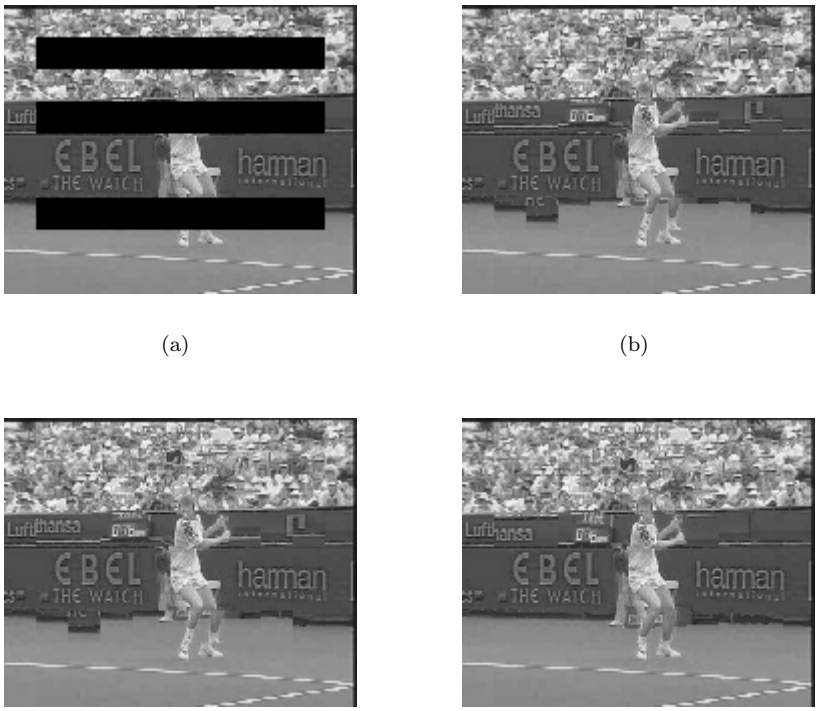


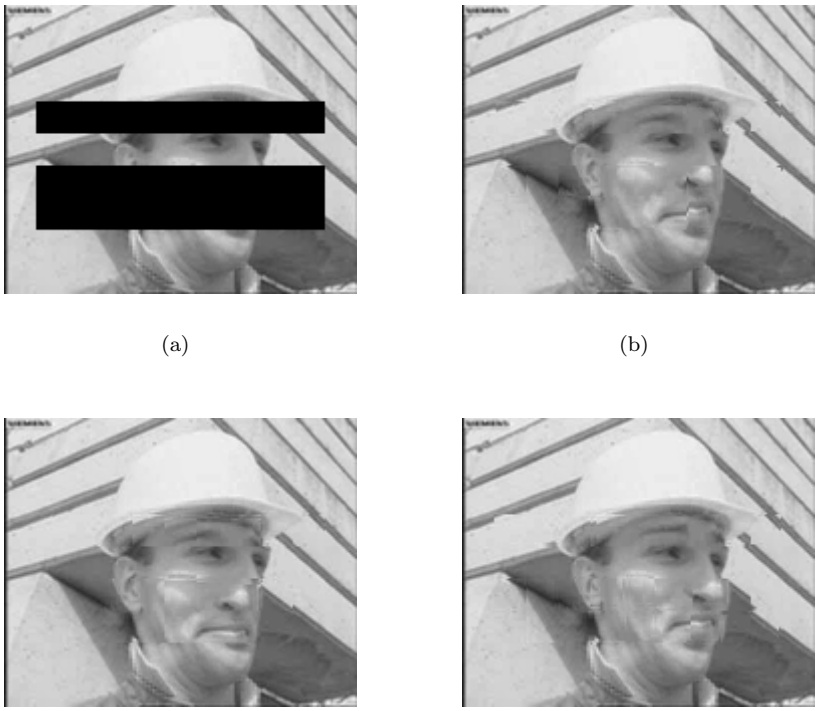
Fig. 12 The 2nd frame of the "Foreman" sequence in CIF resolution :(a) error-free frame (b) corrupted frame (c) concealed with DMVE (d) concealed with BMA (e) concealed with KF (f) concealed with the proposed method (PF)



(a) (b)

(c) (d)

Fig. 13 The 10th frame of the "Stefan" sequence in QCIF resolution :(a) corrupted frame (b) concealed with BMA (c) concealed with KF (d) concealed with the proposed method(PF)



(a) (b)

(c) (d)

Fig. 14 The 7th frame of the "Foreman" sequence in QCIF resolution :(a) corrupted frame (b) concealed with DMVE (c) concealed with KF (d) concealed with the proposed method(PF)