Multiple Hypotheses Bayesian Frame Rate Up-Conversion by Adaptive Fusion of Motion-Compensated Interpolations

Hongbin Liu, Ruiqin Xiong, *Member, IEEE,* Debin Zhao, *Member, IEEE,* Siwei Ma, *Member, IEEE,* and Wen Gao, *Fellow, IEEE*

***Abstract*—Frame rate up-conversion (FRUC) improves the viewing experience of a video because the motion in a FRUC-constructed high frame-rate video looks more smooth and continuous. This paper proposes a multiple hypotheses Bayesian FRUC scheme for estimating the intermediate frame with maximum *a posteriori* probability, in which both temporal motion model and spatial image model are incorporated into the optimization criterion. The image model describes the spatial structure of neighboring pixels while the motion model describes the temporal correlation of pixels along motion trajectories. Instead of employing a single uniquely optimal motion, multiple "optimal" motion trajectories are utilized to form a group of motion hypotheses. To obtain accurate estimation for the pixels in missing intermediate frames, the motion-compensated interpolations generated by all these motion hypotheses are adaptively fused according to the reliability of each hypothesis. We revealed by numerical analysis that this reliability (i.e., the variance of interpolation errors along the hypothesized motion trajectory) can be measured by the variation of reference pixels along the motion trajectory. To obtain the multiple motion fields, a set of block-matching sizes is used and the motion fields are estimated by progressively reducing the size of matching block. Experimental results show that the proposed method can significantly improve both the objective and the subjective quality of the constructed high frame rate video.**

***Index Terms*—Bayesian estimation, frame rate up-conversion, Huber–Markov random field, motion estimation, motion-compensated interpolation.**

Manuscript received June 16, 2011; revised November 2, 2011; accepted December 19, 2011. Date of publication May 2, 2012; date of current version July 31, 2012. This work was supported in part by the National Natural Science Foundation of China, under Grants 61073083, 60736043, and 61121002, in part by the National Basic Research Program of China, 973 Program, under Grant 2009CB320904, in part by the Beijing Natural Science Foundation, under Grant 4112026, and in part by the Specialized Research Fund for the Doctoral Program of Higher Education, under Grants 20100001120027 and 20120001110090. This paper was recommended by Associate Editor P. Salembier.

H. Liu was with the Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China, and with the Institute of Digital Media, School of Electronic Engineering and Computer Science, Peking University, Beijing 100871, China. He is now with the Research and Development Center, LG Electronics China, Beijing 100022, China (e-mail: hongbin.liu@lge.com).

R. Xiong is with the School of Electronic Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: rqxiong@pku.edu.cn).

D. Zhao is with the Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China (e-mail: dbzhao@hit.edu.cn).

S. Ma and W. Gao are with the School of Electronic Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: swma@pku.edu.cn; wgao@pku.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2012.2197081

I. Introduction

With the rapid development of computing and communication technologies in the past decades, digital videos are becoming pervasive in our daily lives. People watch broadcast video programs on television and browse internet videos using desktop computers, laptops, and even mobile devices. At the same time, the display technology also advances rapidly. One important feature is that the screen refresh rate of displays is becoming higher (up to 120 Hz or even higher) so that a dynamic scene can be reproduced on screens with vivid viewing experience. However, this benefit may not be fully realized when the video available for viewing has a much lower frame rate than the display devices can support. In some cases, the video has a high original frame rate but is transmitted at a reduced frame rate, by periodically skipping some frames, in order to meet the transmission rate constraint of a network connection. In some other cases, the frame rate of a captured video is restricted by the processing capability of recording devices. In these scenarios, the user's viewing experience may be severely limited by the video data so that the potential of high refresh rate display cannot be fully exploited.

Frame rate up-conversion (FRUC) [1]–[10] refers to the process to construct a high frame rate (HFR) video by periodically inserting new frames into an input lower frame rate (LFR) video. This improves the viewing experience because the motion in the constructed HFR video usually looks more smooth and continuous. Simple approaches to FRUC include frame repetition and frame averaging. Although the former method cannot improve the viewing experience at all, the latter one is likely to produce ghosting artifacts, because the collocated pixels in adjacent frames do not correspond to the same part of an object if it is moving. Taken this into consideration, a more appropriate approach is to perform frame interpolation along the motion trajectories. This is commonly referred to as motion-compensated frame rate up-conversion (MC-FRUC).

For the success of the MC-FRUC schemes, two issues need to be addressed. The first one is how to figure out the motion trajectories between missing HFR frames and their adjacent frames available in the LFR video, given the fact that the HFR frames are unknown at the time of motion trajectory estimation. The second one is how to estimate the pixels of missing HFR frames from the pixels of LFR frames. These two

issues are handled by motion estimation (ME) and motion-compensated interpolation (MCI), respectively. Many works have been done in these two aspects, which are summarized as follows.

For motion trajectory estimation, Hann *et al.* [2] proposed a 3-D recursive search (3DRS) algorithm, which recursively optimizes the motion vector (MV) obtained from spatially or temporally neighboring blocks. Tai *et al.* [11] proposed a multipass ME scheme to employ variable block sizes to represent the motion field at regions with different characteristics, i.e., using larger block for smooth motion and smaller block for complex motion. This is similar with the idea of variable block-size ME in H.264/AVC [12]. Huang *et al.* [13], [14] proposed to first perform ME for each block using a small block size, and then merge the neighboring blocks with similar MVs into larger block and reestimate the motion for the merged block. Kang *et al.* [15] proposed to use both the bidirectional and unidirectional matching ratios of blocks in the previous and the following references frames to enhance the ME accuracy. The above algorithms use sum absolute difference (SAD) or sum squared difference (SSD) as the criterion to choose MV in block-matching process. Different from the above algorithms, Wang *et al.* [16], [17] explicitly incorporated the temporal and spatial smoothness of the motion field into the ME criterion. In [16], motion fields of adjacent frames are assumed to follow the Markov and Gibbs random field distribution jointly, and the inconsistency of motion in neighboring block is penalized. Besides the algorithms that try to obtain smooth motion field in the ME stage, postprocessing techniques are also proposed to correct inconsistent MVs after the ME stage is completed [18], [19].

MCI derives the intermediate frame according to the estimated motion trajectories. This is usually done in two steps. First, each block in the intermediate frame is assigned a pair of MVs, which point to the previous reference frame and the following reference frame, respectively. Second, the intermediate blocks are interpolated by averaging the reference blocks pointed by the two MVs. However, this usually leads to blocking artifacts at block boundaries. To reduce such artifacts, overlapped block motion compensation (OBMC) is introduced in [20]–[22], in which the application region of each MV is a window larger than the block and can overlap with each other. This makes the transition across block boundaries smooth in the interpolated frame. Recently, a motion-aligned autoregressive model (MAAR) is proposed [23], in which each pixel in the intermediate frame is approximated by a linear combination of the pixels in a square neighborhood in the reference frames. By adaptively estimating the MAAR model parameters, the approach [23] achieves the current state-of-the-art FRUC performance.

The limitations of the above MC-FRUC schemes are as follows. First, most of them do not consider the spatial consistency of neighboring pixels. Second, to the best of our knowledge, none of them consider and analyze the reliability of estimated motion trajectories. Furthermore, none of them consider the possibility of employing multiple motion trajectory hypotheses to obtain a better estimation for the intermediate frame.

In this paper, we propose a multiple hypothesis Bayesian FRUC scheme, in which both temporal motion model and spatial image model are incorporated into the optimization criterion for estimating an intermediate frame with maximum *a posteriori* probability. The image model describes the spatial structure of neighboring pixels while the motion model describes the temporal correlation of pixels along motion trajectories. Instead of employing a single uniquely optimal motion, multiple "optimal" motion fields are utilized to form a group of motion trajectory hypotheses. To obtain accurate estimation for the pixels in missing intermediate frames, the interpolated frames generated with these motion trajectory hypotheses are fused together according to the reliability of each hypothesis. By numerical analysis from some experiments, we revealed that the variance of MCI errors can be accurately modeled by the variation of reference pixels along the motion trajectory.

To obtain the multiple motion fields, we propose a new ME scheme, in which a set of block sizes is used and the motion fields are estimated in several ME steps, by progressively reducing the block size in block matching. To reduce the motion ambiguity in the motion search process, the motion estimated with large matching block in an earlier step is used to constrain the motion search in the next step. The proposed method is different with the variable block-size ME in [12]. The latter one generates only a single motion field, allowing different block sizes for block matching at different regions. On the other hand, the proposed method generates multiple estimations of the motion field via multiple ME stages. Each estimation stage is performed using a fixed block size, but the block size is progressively reduced as the ME stages go on.

The remainder of this paper is organized as follows. Section II reviews the basic FRUC model. Section III proposes the multiple hypotheses Bayesian FRUC model. In its sections, the concept of multiple hypotheses Bayesian estimation is introduced and the numerical solution for the proposed optimization problem is discussed. The reliability of MCI is also analyzed. Section IV proposes the progressively reduced block-size ME scheme. Experimental results are reported in Section V to evaluate the proposed FRUC scheme. Finally, Section VI concludes this paper.

II. Basic FRUC Model

A. Problem Statement

In this paper, we focus on the problem of how to double the frame rate of an input video. The more general FRUC problem of increasing the frame rate by other factors can be similarly solved. Suppose f_t is the intermediate frame to be estimated, and f_{t-1} and f_{t+1} are the previous and the following neighboring frames of f_t, respectively. The goal of FRUC problem is to find a pixel value with the maximum probability for each pixel of f_t, based on f_{t-1} and f_{t+1}. The mathematical formulation of this problem is

$$\hat{f}_t = \arg\max_{f_t} \Pr\left(f_t | f_{t-1}, f_{t+1}\right). \tag{1}$$

Here, $\Pr(\cdot)$ is the probability density function.

The intuition behind the formulation (1) is that the three consecutive frames f_{t-1}, f_t, and f_{t+1} should be consistent and form a continuous scene. In other words, stationary object in the three frames should be highly similar to each other, while moving object may move from one place to another in the three frames. Therefore, the motion that links the intermediate frame f_t with the pair of reference frames f_{t-1} and f_{t+1} is the key factor to consider in the FRUC problem. Explicitly incorporating the impact of motion, (1) can be reformulated as

$$\begin{aligned}\hat{f}_t &= \arg\max_{f_t} \int \Pr(f_t, m_t | f_{t-1}, f_{t+1})\, dm_t \\ &= \arg\max_{f_t} \int \Pr(f_t | f_{t-1}, f_{t+1}, m_t) \Pr(m_t | f_{t-1}, f_{t+1})\, dm_t.\end{aligned} \tag{2}$$

Here, m_t is the motion field that links f_t with f_{t-1} and f_{t+1}. The formulation (2) indicates that the FRUC problem can be solved by dividing it into estimation problems of two probabilities, i.e., the probability of motion field and the conditional probability of intermediate frame given motion. The conditional probability $\Pr(f_t | f_{t-1}, f_{t+1}, m_t)$ is usually well defined when m_t is accurate or very close to the true motion that links the frames f_{t-1}, f_t, and f_{t+1}. However, for motion m_t that is far from the true motion, this conditional probability is difficult to model. As a result, (2) cannot be solved directly. In practice, the FRUC problem (2) is commonly simplified to a two-step problem, as formulated by

$$\hat{m}_t = \arg\max_{m} \Pr(m | f_{t-1}, f_{t+1}) \tag{3}$$

$$\hat{f}_t = \arg\max_{f_t} \Pr(f_t | f_{t-1}, f_{t+1}, \hat{m}_t). \tag{4}$$

In this framework, a MV field $\hat{m}_t$ that is most compatible with the reference frames f_{t-1} and f_{t+1} is chosen as the optimal motion in the first step, and then the intermediate frame is estimated in the second step, based on the conditional probability $\Pr(f_t | f_{t-1}, f_{t+1}, \hat{m}_t)$ and the estimated motion $\hat{m}_t$.

B. Basic Solution

Generally speaking, for a complex dynamic scene, one cannot accurately figure out the motion at an arbitrary moment with a long time interval, only by the images recorded at the beginning and the end of this interval. To make the things easy, certain assumptions were made by most of the existing FRUC schemes. The first assumption is that the speed and direction of object movements do not change too much during this period, e.g., from the time $t-1$ to the time $t+1$, so that the motion looks smooth and continuous. This assumption is reasonable when the time interval between f_{t-1} and f_{t+1} is very short. Under such assumption, the motion that links f_t to f_{t-1} and the motion that links f_t to f_{t+1} are antisymmetric, as shown in Fig. 1. In this case, the motion linking f_t to f_{t+1} (or f_t to f_{t-1}) is usually estimated by first determining the motion linking f_{t-1} to f_{t+1} via block-matching algorithms and then scaling the obtained MVs by 1/2 (or $-1/2$).

Once an accurate estimation of motion field is obtained, we can interpolate the intermediate frame. For this purpose, another assumption is usually made that the intensity of image pixels remains approximately stable along motion trajectories,

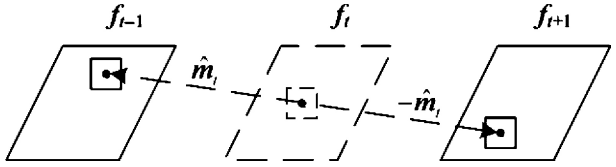


Fig. 1. Motion that links the intermediate frame with the reference frames.

except for a random disturbance, e.g., white Gaussian noise [24]. According to the above assumptions, f_t can be estimated from f_{t-1} and f_{t+1} by

$$f_t(x) = f_{t-1}(x + \hat{m}_t(x)) + n_1(x) \tag{5}$$

$$f_t(x) = f_{t+1}(x - \hat{m}_t(x)) + n_2(x). \tag{6}$$

Here, $\hat{m}_t$ denotes the backward motion field of f_t, as shown in Fig. 1. x is the coordinate vector, and n_1 and n_2 are the random disturbance noises that are assumed to be independent of each other and with equal variance. Combining (5) and (6), we get

$$f_t(x) = \frac{f_{t-1}(x + \hat{m}_t(x)) + f_{t+1}(x - \hat{m}_t(x))}{2} + \frac{n_1(x) + n_2(x)}{2}. \tag{7}$$

Obviously, the estimation of f_t at position x is mainly determined by the mean pixel value along the motion trajectory that passes through the pixel x of f_t. We introduce a MCI operator $\mathcal{M}(\cdot, \cdot, \cdot)$ that takes two frames and one motion field as input so that $f_c = \mathcal{M}(f_a, f_b, m)$ is defined by

$$f_c(x) = \frac{f_a(x + m(x)) + f_b(x - m(x))}{2}. \tag{8}$$

Then the optimal estimation of f_t in the basic FRUC model is

$$\hat{f}_t = \mathcal{M}(f_{t-1}, f_{t+1}, \hat{m}_t). \tag{9}$$

III. Multihypothesis Bayesian FRUC Model

A. Multihypothesis FRUC

Since motion is the key factor in FRUC problems, it is critical to obtain accurate estimation of the motion within the time interval between reference frames. Most existing FRUC schemes follow the idea described in the above section. One common feature of them is that only *one* motion field that is believed to be *uniquely* optimal is used to derive the intermediate frame. This motion field is usually obtained by performing block matching between f_{t-1} and f_{t+1}, using distance metrics such as SAD or SSD as the block-matching criteria.

However, since the data of intermediate frame f_t is missing, it is impossible to evaluate the true accuracy of a motion candidate with respect to the true motion of f_t. Indeed, the distance metric used by block matching only serves as an approximation of the appropriateness of each motion candidate. In this sense, the optimality of searched motion is not completely reliable. On the other hand, the motion field can be estimated with many different strategies, e.g., using different ME algorithms or parameters. The motion fields generated by these strategies are not necessarily the same, although they are all believed to

be "optimal" under their own individual optimization criterion. Since the optimality of these motion fields are obscure and it is difficult to pick out the real optimal one, we need a mechanism to make use of all these motion fields simultaneously.

For this purpose, we propose an extended FRUC framework. In this framework, instead of seeking for a single uniquely "optimal" motion field $\hat{m}_t$ in the ME procedure, a group of "optimal" motion fields, say $\hat{m}_{t,1}, \hat{m}_{t,2}, \cdots, \hat{m}_{t,K}$, are searched using a group of ME strategies $S_1, S_2, \cdots, S_K$. In this way, the conventional FRUC model formulated by (3) and (4) is extended as follows:

$$\begin{aligned}\hat{m}_{t,i} &= \mathrm{ME}\left(f_{t-1}, f_{t+1}, S_i\right) \\ &= \arg\max_{m} \Pr\left(m | f_{t-1}, f_{t+1}, S_i\right), \ i = 1, 2, \ldots, K\end{aligned} \tag{10}$$

$$\hat{f}_t = \arg\max_{f_t} \Pr\left(f_t | f_{t-1}, f_{t+1}, \left\{\hat{m}_{t,i}\right\}_{i=1,\ldots,K}\right). \tag{11}$$

Of course, the conditional probability $\Pr\left(f_t | f_{t-1}, f_{t+1}, \left\{\hat{m}_{t,i}\right\}_{i=1,\ldots,K}\right)$ is complicated. To keep the complexity under control, in practical implementation of the above model, (11) may be approximated by

$$\hat{f}_t \approx \arg\max_{f_t} \prod_{i=1}^{K} \Pr\left(f_t | f_{t-1}, f_{t+1}, \hat{m}_{t,i}\right). \tag{12}$$

In this model, each estimated motion field $\hat{m}_{t,i}$ provides a hypothesis for estimating the intermediate frame f_t. We call this model multihypotheses FRUC.

B. Multihypothesis Bayesian FRUC

In the FRUC models we discussed so far, the pixels in an intermediate frame f_t are estimated separately, assuming that these pixels are independent of each other. However, it is well known that spatially adjacent pixels of natural images are highly correlated. It means that when estimating a pixel in f_t, the optimal value should not only be close to the temporally neighboring pixels along motion trajectories but also be consistent with the spatial structure exhibited by the neighboring pixels. Taking this aspect into consideration, the FRUC problem (11) is reformulated by

$$\hat{f}_t = \arg\max_{f_t} \Pr\left(f_{t-1}, f_{t+1}, \left\{\hat{m}_{t,i}\right\}_{i=1,\ldots,K} | f_t\right) \cdot \Pr\left(f_t\right). \tag{13}$$

The first term $\Pr\left(f_{t-1}, f_{t+1}, \left\{\hat{m}_{t,i}\right\}_{i=1,\ldots,K} | f_t\right)$ is the likelihood function that links f_t with the data in reference frames according to the temporal model and estimated motion in the video sequence. The second term in $\Pr\left(f_t\right)$ is the image prior model describing the spatial structure of neighboring image pixels. This is usually modeled by Markov random field. We will discuss the likelihood function and the image prior model in detail in the subsequent section. Similar to the previous discussion, to control the complexity of implementation, this problem can be approximated by

$$\begin{aligned}\hat{f}_t &\approx \arg\max_{f_t} \prod_{i=1}^{K} \Pr\left(f_{t-1}, f_{t+1}, \hat{m}_{t,i} | f_t\right) \cdot \Pr\left(f_t\right) \\ &= \arg\max_{f_t} \sum_{i=1}^{K} \mathrm{Log}\Pr\left(f_{t-1}, f_{t+1}, \hat{m}_{t,i} | f_t\right) + \mathrm{Log}\Pr\left(f_t\right).\end{aligned} \tag{14}$$

C. Temporal Motion Model and Spatial Image Prior Model

Before we can numerically solve the FRUC problem (14), we need to adopt an appropriate motion model to describe the relationship of temporally adjacent frames and an appropriate image prior model to formulate the correlation of spatially neighboring pixels inside a frame. According to the discussion in Section II-B, the temporal relationship of adjacent frames can be formulated by

$$\begin{aligned}&\Pr\left(f_{t-1}, f_{t+1}, \hat{m}_{t,i} | f_t\right) \propto \\ &\prod\nolimits_{x\in\Lambda} \frac{1}{\sqrt{2\pi}\sigma_{t,i}\left(x\right)} \exp \\ &\left\{-\frac{\left(f_t\left(x\right) - \mathcal{M}\left(f_{t-1}, f_{t+1}, \hat{m}_{t,i}\right)\left(x\right)\right)^2}{2\sigma_{t,i}\left(x\right)^2}\right\}.\end{aligned} \tag{15}$$

Here, $\sigma_{t,i}\left(x\right)^2$ is the variance of the random disturbance noise on the motion trajectory that passes through the pixel x of f_t and is determined by $\hat{m}_{t,i}\left(x\right)$; Λ is the set of all pixel positions in a frame.

On the other hand, the spatial correlation within a video frame can be modeled by Huber–Markov random field (HMRF) [25]. We rewrite the video frame $f_t(x)$ in its vector form as $\mathbf{f}_t$, lexicographical ordered by x. Then the HMRF image prior model can be formulated as

$$\Pr\left(f_t\right) = \frac{1}{Z} \exp\left(-\frac{1}{\lambda} \sum_{c\in C} \rho_T\left(\mathbf{d}_c^T \mathbf{f}_t\right)\right). \tag{16}$$

Here, Z is normalization constant, λ is "temperature" parameter, c is a clique (i.e., a group of connected pixels), and C is the set of all cliques in a frame. $\mathbf{d}_\mathrm{c}$ is a column vector defined to extract the variation of pixel values in the clique c and $\mathbf{d}_\mathrm{c}^\mathrm{T}$ is its transpose. $\rho_T\left(\cdot\right)$ is the Huber function given by

$$\rho_T\left(z\right) = \begin{cases} z^2 & |z| \le T \\ T^2 + 2T\left(|z| - T\right) & |z| > T. \end{cases} \tag{17}$$

Here, T is a threshold to preserve significant edges. In this paper, each clique c is defined to be a pixel and one of its four nearest neighboring pixels.

Integrating the motion model (15) and prior image model (16) into (14), the core of our proposed FRUC model is to minimize the following cost function:

$$\begin{aligned}J(f_t) = &\sum_{i=1}^{K} \sum_{x\in\Lambda} \frac{\left(f_t\left(x\right) - \mathcal{M}\left(f_{t-1}, f_{t+1}, \hat{m}_{t,i}\right)\left(x\right)\right)^2}{2\sigma_{t,i}\left(x\right)^2} \\ &+ \frac{1}{\lambda} \sum_{c\in C} \rho_T\left(\mathbf{d}_c^T \mathbf{f}_t\right) \\ = &\sum_{i=1}^{K} \left\{\left(\mathbf{f}_t - \mathbf{p}_{t,i}\right)^T \Gamma_{t,i} \left(\mathbf{f}_t - \mathbf{p}_{t,i}\right)\right\} + \frac{1}{\lambda} \sum_{c\in C} \rho_T\left(\mathbf{d}_c^T \mathbf{f}_t\right).\end{aligned} \tag{18}$$

Here, we rewrite $p_{t,i}(x) = \mathcal{M}\left(f_{t-1}, f_{t+1}, \hat{m}_{t,i}\right)\left(x\right)$ in its vector form as $\mathbf{p}_{t,i}$ (i.e., prediction for $\mathbf{f}_t$) for the convenience of later discussions. $\Gamma_{t,i}$ is a diagonal matrix whose diagonal elements are $1/2\sigma_{t,i}\left(x\right)^2$, lexicographically ordered by x. The first term in (18) confines the solution to the ones close to the weighted mean pixel values along motion trajectories. The second term of (18) confines the solution to be spatially

smooth. We note that the parameter $\sigma_{t,i}(x), i = 1, 2, \ldots, K$ indicates the reliability of $p_{t,i}(x)$ as an approximation of $f_t(x)$. When $\sigma_{t,i}(x)$ is smaller, $p_{t,i}(x)$ is more reliable and the deviation of $f_t(x)$ from $p_{t,i}(x)$ is penalized with higher weight in (18), and vice versa. The estimation of $\sigma_{t,i}(x)$ will be discussed in the subsequent section.

D. Numerical Solution

The FRUC problem in (18) is a convex optimization problem. We use the steepest descent method [26] to find its solution. The optimization procedure consists of a number of iterations, and in each iteration, the estimation of $\mathbf{f}_t$ is adjusted toward the direction that minimizes the cost function most rapidly. Let $\mathbf{f}_t^{(p)}$ denote the estimate of $\mathbf{f}_t$ in the pth iteration, then (19) is used to generate the next estimation as follows:

$$\mathbf{f}_t^{(p+1)} = \mathbf{f}_t^{(p)} + \alpha^{(p)}\mathbf{r}^{(p)}. \tag{19}$$

Here, $\mathbf{r}^{(p)}$ and $\alpha^{(p)}$ are the steepest direction and the step size in the pth iteration, which are defined as follows:

$$\mathbf{r}^{(p)} = -\left(\sum_{c\in C}\rho_T'\left(\mathbf{d}_c^T\mathbf{f}_t^{(p)}\right)\mathbf{d}_c + 2\lambda\sum_{i=1}^{K}\Gamma_{t,i}(\mathbf{f}_t^{(p)} - \mathbf{p}_{t,i})\right) \tag{20}$$

$$\alpha^{(p)} = \frac{\mathbf{r}^{(p)T}\mathbf{r}^{(p)}}{\mathbf{r}^{(p)T}\left(\sum_{c\in C}\rho_T''\left(\mathbf{d}_c^T\mathbf{f}_t^{(p)}\right)\mathbf{d}_c\mathbf{d}_c^T + 2\lambda\sum_{i=1}^{k}\Gamma_{t,i}\right)\mathbf{r}^{(p)}}. \tag{21}$$

Here, $\rho_T'(\cdot)$ and $\rho_T''(\cdot)$ are the first and the second order derivative of $\rho_T(\cdot)$. To guarantee that the cost function does decrease in each iteration, we check the step size $\alpha^{(p)}$ and repeatedly halve the step size when necessary.

From (20) and (21), we can see that to solve the FRUC problem in (18), we need to estimate $\mathbf{p}_{t,i}$ and $\Gamma_{t,i}$ given that λ and $\mathbf{d}_c$ are constants. The following section discusses how to estimate $\Gamma_{t,i}$, and Section IV will describe how to estimate $\mathbf{p}_{t,i}$, i.e., prediction frame of the intermediate frame.

E. Reliability Estimation of Prediction Pixels

In this section, we discuss how to estimate the matrix Γ_i or the variance $\sigma_{t,i}(x)^2$ of random disturbance along the estimated motion trajectories at $f_t(x)$, for all x. We already mentioned that the variance $\sigma_{t,i}(x)^2$ of random disturbance indicates the reliability of $p_{t,i}(x)$ (the mean pixel value along the motion trajectory) as a prediction of the intermediate frame pixel $f_t(x)$. Intuitively, the value of $\sigma_{t,i}(x)^2$ depends on how reliable it is that the estimated MV form a true motion trajectory. Since the data of intermediate frame is missing and only reference frames are available, we measure the reliability of an estimated MV only based on variation of reference frame pixels along the motion trajectory. In the following, we only discuss how to measure the reliability of a MV, but leave the problem of how to generate MV fields in Section IV.

We denote that $d_{t,i}(x) = f_t(x) - p_{t,i}(x)$ and $s_{t,i}(x) = \left|f_{t-1}\left(x + \hat{m}_{t,i}(x)\right) - f_{t+1}\left(x - \hat{m}_{t,i}(x)\right)\right|$ ($\hat{m}_{t,i}$ is the ith estimated MV field of f_t) for a numerical analysis. The expectation of $d_{t,i}(x)^2$, i.e., $\mathrm{E}[d_{t,i}(x)^2]$, is supposed to be a good approximation of $\sigma_{t,i}(x)^2$. Furthermore, this value

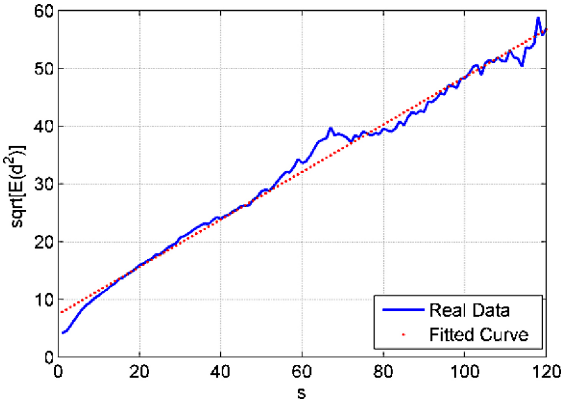


Fig. 2. Relationship between $\sqrt{\mathrm{E}[d^2]}$ and s.

is expected to be highly dependent on the value of $s_{t,i}(x)$. To establish the relationship between $\sigma_{t,i}(x)^2$ and $s_{t,i}(x)$, we collect the (d, s) pairs from a number of real videos, by first estimating the MVs $\hat{m}_{t,1}, \hat{m}_{t,2}, \cdots, \hat{m}_{t,K}$ using a group of ME strategies and then varying the parameters x, t, and i to get all the pairs ($d = d_{t,i}(x)$, $s = s_{t,i}(x)$). We divide the range of s into many small bins and evaluate the expectation $\mathrm{E}[d^2]$ for each bin. According to the numerical analysis results based on real video data, it turns out that the relationship between $\sqrt{\mathrm{E}[d^2]}$ and s can be model by a simple linear model $\sqrt{\mathrm{E}[d^2]} = as + b$, as shown in Fig. 2. That means we have the approximation $\sigma_{t,i}(x) \approx as_{t,i}(x) + b$.

IV. Progressively Reduced Block-Size ME

Now, we turn to the problem of generating multiple motion fields to form the multiple hypotheses (i.e., multiple prediction frames) for the intermediate frame. Since the intermediate frame is missing and its motion field is derived from that of its reference frames, this problem becomes generating multiple motion fields for the reference frames. A simple way is to perform conventional ME, i.e., through block matching with a fixed block size, and select the first K MVs that have the lowest block-matching cost (e.g., SAD) for each block in the reference frame. However, this method obviously introduces suboptimal MVs. It is most likely that these $K-1$ extra MVs will degrade the quality of the ultimate estimated intermediate frame.

Instead of using the above simple but naive approach, this paper proposes to generate multiple MVs by using a set of block sizes for block matching. The reason for employing multiple block sizes in the block-matching process is that each block size may be suitable for certain cases of video content based on the following discussions.

It is well known that a very small block size (e.g., only one pixel in the most extreme case or a 2-by-2 block in a less extreme case) is not suitable for block matching because there may exist many image patches in the reference frame that are similar to the current image block. Therefore, a very small block size for block matching is likely to introduce motion ambiguity. In this sense, a large block size helps to reduce the ambiguity in the motion optimization process and is therefore preferred. On the other hand, it is widely recognized that a

typical video may contain complex motion that cannot be described by global translation. In other words, the motion in different regions may vary and the boundary of each global motion region can be complex. When a very large block size is used for block matching, it is likely that only part of the content in the current image block can be exactly matched with the reference frame, no matter which translation vector is tested. In this case, the selected motion is the compromised result of different parts in the block. Therefore, a very large block size for block matching is likely to introduce motion inaccuracy. In this sense, a small block size helps to improve the accuracy of motion in the motion optimization process and is therefore preferred. To summarize the above discussions, it is not easy to determine the optimal block size that can produce the most accurate motion field via block matching.

Therefore, we propose a progressively reduced block-size motion estimation (PRBME) scheme to generate multiple motion fields. The whole process is completed in several ME passes, and later ME pass uses smaller block size than earlier ME pass. In the following, we take the f_{t+1}, for instance, to describe how to generate multiple motion fields for the reference frames. Let f_{t-1} and f_{t+1} be the forward and backward reference frame, and let f_t be the intermediate frame. Let $\left\{M \times M, \ldots, M/2^{N-1} \times M/2^{N-1}\right\}$ be the used block-size set, where N is the number of the ME passes. Let x be the coordinate vector of a block. Let $\hat{m}_{t+1,i}$ and $\hat{m}_{t,i}$ be the motion field of f_{t+1} and f_t generated in the ith ME pass, respectively. The PRBME procedure is performed as follows.

1) *Initialization*: Set $i = 0$ for the first ME pass.
2) *ME for* f_{t+1}: Split f_{t+1} into nonoverlapping blocks of the same size $M/2^i \times M/2^i$. For each block $x \in f_{t+1}$ (blocks are processed in raster-scan order), do the following substeps.
   a) *Generate the MV predictor*

$$\begin{aligned}\mathrm{PMV}_i(x) = \mathrm{median}\Bigg\{&\hat{m}_{t+1,i}\left(\frac{x}{b_i}+\begin{pmatrix}-1\\-1\end{pmatrix}\right), \hat{m}_{t+1,i}\left(\frac{x}{b_i}+\begin{pmatrix}0\\-1\end{pmatrix}\right)\\ &\hat{m}_{t+1,i}\left(\frac{x}{b_i}+\begin{pmatrix}1\\-1\end{pmatrix}\right), \hat{m}_{t+1,i}\left(\frac{x}{b_i}+\begin{pmatrix}-1\\0\end{pmatrix}\right), \hat{m}_{t+1,i-1}\left(\frac{x}{b_{i-1}}\right)\Bigg\}\\ &\text{for } i = 1, 2, \ldots, K.\end{aligned} \tag{22}$$

   Here, $b_i = M/2^i$ is the block size for block matching in the ith ME pass. The first four MVs in (22) are the MVs of the upper left, upper, upper right, and left blocks of block x, respectively (note that since the blocks are processed in raster-scan order, these four MVs are already available). The last MV in (22) is the MV estimated from an earlier ME pass, i.e., the $(i-1)$th ME pass. The idea in (22) is that the motion field estimated in a previous ME pass is used to constrain the motion in a later ME pass. This helps to reduce the motion ambiguity in later ME passes.
   b) *Block matching*: Search for the best MV by minimizing

$$J_{\mathrm{ME}}(x, mv, i) = \sum_{y \in \Omega_i(x)} |f_{t+1}(y) - f_{t-1}(y+mv)| + \lambda_1 \cdot |mv - \mathrm{PMV}_i(x)|^2. \tag{23}$$

TABLE I
TRAINED PARAMETER PAIRS (a, b) FOR THE LINEAR VARIANCE MODEL $\sqrt{\mathrm{E}[d^2]} = as + b$ IN DIFFERENT ME STEPS

(i, l)	(1, Fwd)	(2, Fwd)	(3, Fwd)	(4, Fwd)
a	0.71003	0.78738	0.62221	0.71361
b	3.32907	3.57604	3.69838	3.78688
(i, l)	(1, Bwd)	(2, Bwd)	(3, Bwd)	(4, Bwd)
a	0.74137	0.87074	0.67078	0.77185
b	3.33214	3.97756	3.60738	4.17610

Here, $\Omega_i(x)$ is the set of pixel coordinate in block x in the ith ME pass, and mv is the candidate MV. The first term in (23) is the block-matching distance. The second term in (23) controls the spatial consistency of motion field by the parameter λ_1. The optimal MV selected by (23) is stored in $\hat{m}_{t+1,i}\left(\frac{x}{b_i}\right)$.

3) *MV postprocessing for* f_{t+1}: For each block $x \in f_{t+1}$, nine MVs including MVs of the block x and x's eight neighboring blocks are checked, and the MV minimizing the block-matching difference (i.e., SAD) is selected as the final MV for x.
4) *MV derivation for* f_t: For each block $x \in f_t$, its MV is derived as half the MV of its collocated block in f_{t+1}, i.e., $\hat{m}_{t,i}(x) = \hat{m}_{t+1,i}(x)/2$.
5) *Predicting the intermediate frame*: For each block $x \in f_t$, its predicted block is generated as

$$p_{t,i}(y) = \frac{f_{t-1}\left(y + \hat{m}_{t,i}(x)\right) + f_{t+1}\left(y - \hat{m}_{t,i}(x)\right)}{2} \quad \text{for all } y \in \Omega_i(x). \tag{24}$$

Here, $p_{t,i}$ is the prediction frame of f_t generated in the ith ME pass. Note that there is also another way to derive prediction for pixels in f_t: place the estimated MV to the pixels of f_t where this MV crosses. However, it usually introduces overlapped areas and holes that degrade the image quality, and is therefore not used in this paper.

6) *Check stop*: Set $i = i + 1$. If $i < N$, go to step 2 for the next ME pass. Otherwise, the whole PRBME procedure is finished.

When generating motion fields for f_{t-1}, similar procedure is performed. We only need to switch the role of f_{t+1} and f_{t-1} in the above process. After PRBME is performed on both f_{t+1} and f_{t-1}, there are $K = 2N$ predictions produced for f_t.

With the estimated motion fields of $f_t(x)$, we can estimate the reliability of each motion trajectory, as discussed in Section III-E. Knowing $\sigma_{t,i}(x)$ and $p_{t,i}(x)$, as stated in (20) and (21) in Section III-D, we can run the Bayesian FRUC and generate the final estimation of $f_t(x)$.

V. EXPERIMENTAL RESULTS

In this section, various experiments are conducted to evaluate the performance of our proposed FRUC scheme.

A. Experiment Settings

In the experiment, two resolutions CIF (352×288) and 720P (1280×720) are tested, including CIF 30 Hz sequences *Bus*,

TABLE II
PSNR (dB) COMPARISON OF DIFFERENT FRUC METHODS FOR CIF AND 720P VIDEO SEQUENCES

CIF									
	Football	*Bus*	*Mobile*	*Stefan*	*Flower*	*Highway*	*Foreman*	*News*	Average
3DRS	22.28	25.99	27.03	27.75	31.18	32.82	33.51	36.38	*29.62*
OBMC	22.80	27.29	28.14	28.89	31.74	33.09	35.08	37.19	*30.53*
MAAR	22.81	27.00	28.18	28.93	32.02	33.25	35.28	37.40	*30.61*
DualME	20.90	22.96	22.45	24.01	26.50	32.27	31.65	34.79	*26.94*
Proposed	23.23	29.32	31.75	29.52	33.09	33.70	35.59	37.85	*31.76*
720P									
	Spin Calendar	*City*	*Harbor*	*Night*	*Sailormen*	*Optis*	*Shuttle Start*	*Big Ships*	Average
3DRS	27.58	29.39	31.84	32.09	34.52	39.69	41.54	40.50	*34.64*
OBMC	28.26	30.60	32.38	31.52	35.58	40.23	43.19	40.76	*35.32*
MAAR	31.04	30.53	32.35	31.52	35.98	40.40	43.51	41.05	*35.80*
DualME	23.30	27.66	28.98	27.15	30.85	36.24	40.50	36.66	*31.42*
Proposed	35.39	33.19	32.72	32.85	36.34	40.51	43.42	41.08	*36.94*

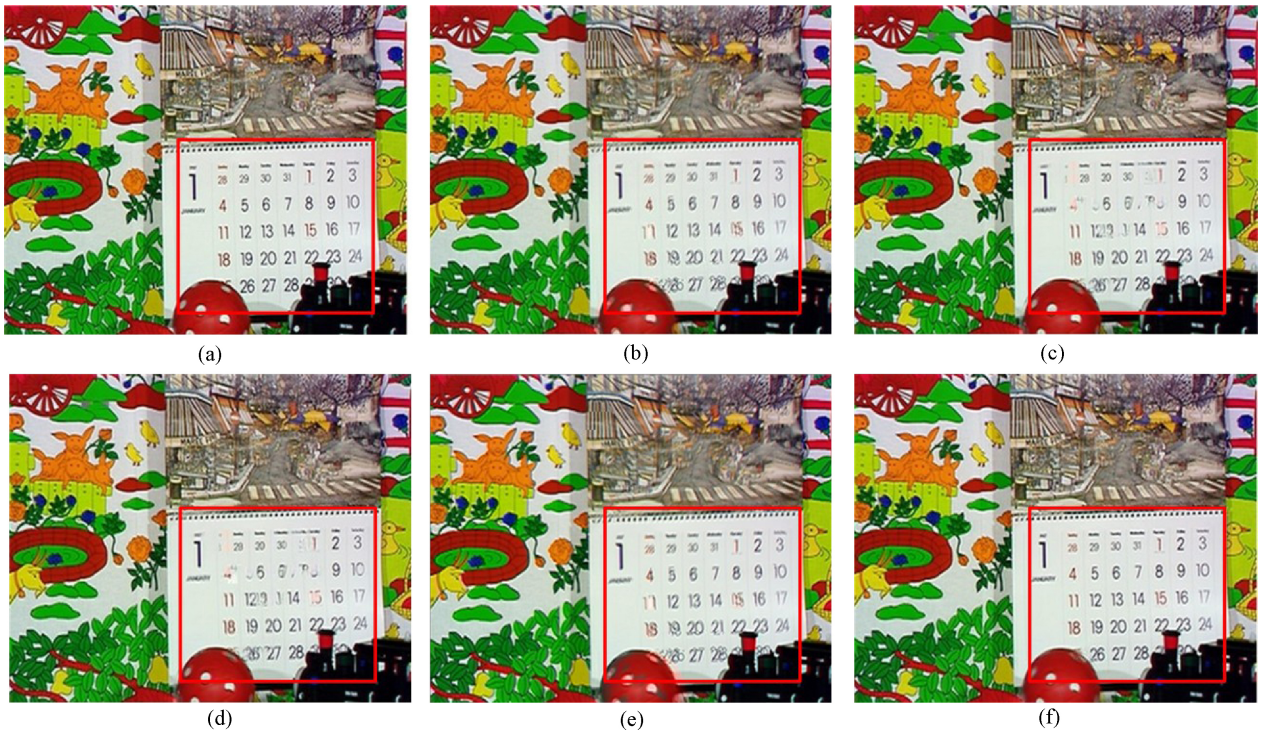


Fig. 3. FRUC results for *Mobile* (12th frame). (a) Original. (b) 3DRS. (c) OBMC. (d) MAAR. (e) DualME. (f) Proposed.

Mobile, *Flower*, *Football*, *Foreman*, *News*, *Stefan*, and *Highway*, and 720P 60 Hz sequences *Night*, *Spin Calendar*, *City*, *Harbor*, *Sailormen*, *Optis*, *Shuttle Start*, and *Big Ships*. To quantitatively measure the quality of the interpolated frames, we remove the first 50 even frames and reconstruct them from the first 51 odd frames for each sequence using FRUC techniques, and then compare the reconstructed frames to the original frames.

In our experiments, four FRUC algorithms are selected as benchmarks, including the well-known 3DRS [2], OBMC [21], MAAR (one of the state-of-the-art FRUC schemes) [23], and DualME [15]. In the experiments, the motion search range for MCI, OBMC, MAAR, and DualME is set to 17×17. The ME block size used in these four benchmarks is 8×8. In the proposed PRBME scheme, the block size varies from 32×32 to 4×4; therefore, size of the block size set is 4. λ_1 in the proposed ME criterion in (23) is set to 1. The parameter λ in (20) and (21) is set to 2000 and T in (17) is set to 5 empirically.

In the following, we show the offline trained parameters a and b for estimating $\sigma_{t,i}(x)$, as discussed in Section III-E. The trained parameters are listed in Table I, where i means block size $\left(32/2^i\right) \times \left(32/2^i\right)$ is used, l = Fwd means ME for f_{t+1}, and l = Bwd means ME for f_{t-1}. In the offline training, CIF sequences *Foreman*, *Tempete*, *Flower*, and *News* are used. We note that the parameters are quite stable, i.e., a varies slight around 0.7 and b varies in the range 3.3–4.1. It turns out that these coefficients can work well on other CIF and 720P sequences.

B. Performance Comparison

In this section, the proposed algorithm is compared with four benchmarks both objectively and subjectively.

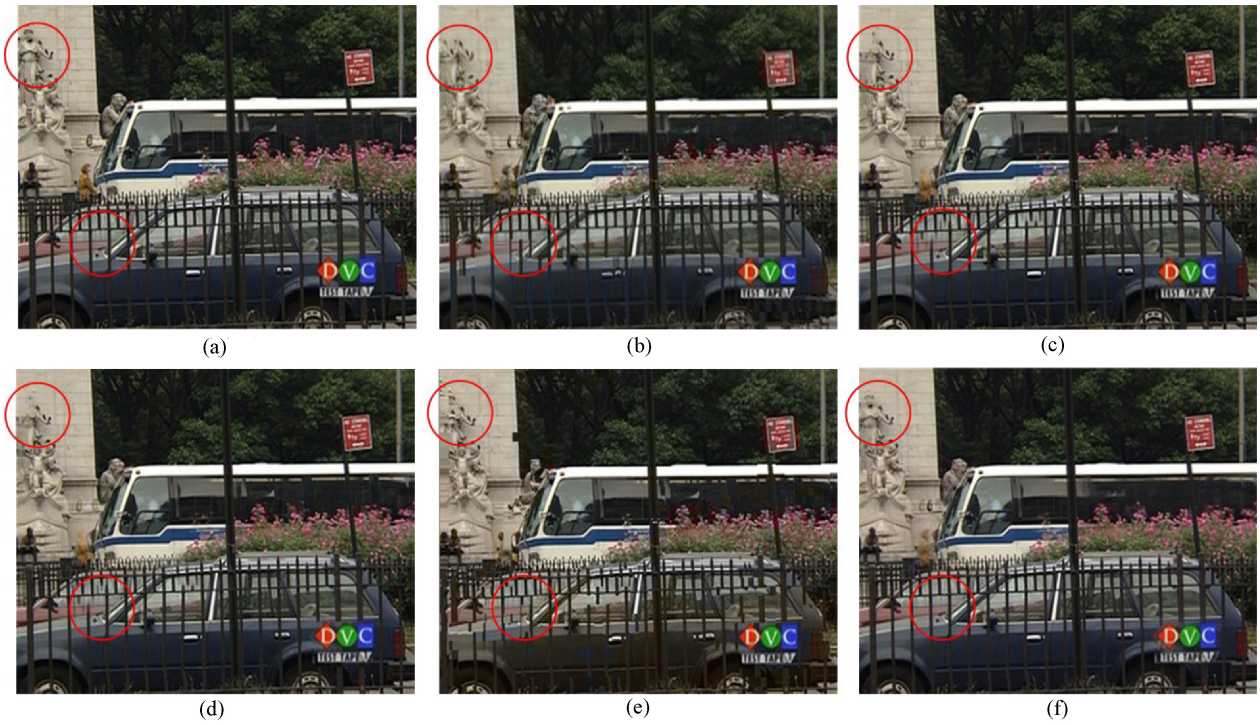


Fig. 4. FRUC results for *Bus* (18th frame). (a) Original. (b) 3DRS. (c) OBMC. (d) MAAR. (e) DualME. (f) Proposed.

TABLE III
COMPARISON OF AVERAGE PROCESSING TIME (S/FRAME)

	CIF								
	Football	*Bus*	*Mobile*	*Stefan*	*Flower*	*Highway*	*Foreman*	*News*	Average
3DRS	0.055	0.059	0.062	0.062	0.060	0.061	0.063	0.055	*0.060*
OBMC	0.262	0.252	0.268	0.241	0.227	0.211	0.222	0.196	*0.235*
MAAR	1.153	12.65	12.949	1.146	4.075	1.084	1.139	1.139	*4.417*
DualME	0.667	0.666	0.653	0.646	0.627	0.617	0.648	0.644	*0.646*
Proposed	1.263	1.141	1.089	1.109	1.097	1.122	1.103	0.981	*1.113*
	720P								
	Spin Calendar	*City*	*Harbor*	*Night*	*Sailormen*	*Optis*	*Shuttle Start*	*Big Ships*	Average
3DRS	0.659	0.549	0.548	0.542	0.552	0.547	0.546	0.554	*0.562*
OBMC	2.182	2.175	2.263	2.080	2.185	2.044	1.914	2.038	*2.110*
MAAR	41.133	40.982	38.718	38.59	38.90	37.991	38.222	40.753	*39.411*
DualME	5.987	6.245	6.076	5.878	6.208	6.237	6.023	6.323	*6.122*
Proposed	10.521	10.362	10.478	10.567	10.053	10.418	10.441	10.367	*10.401*

The average peak signal-to-noise ratios (PSNRs) of the 50 interpolated frames are shown for each test sequence in Table II. It can be observed that when compared with the best one of 3DRS, OBMC, MAAR, and DualME, our proposed FRUC scheme improves the results by at most 3.57 dB and 4.35 dB for the CIF and 720P sequences, respectively. The proposed scheme shows its superiority on sequences containing different scales of motion where the prediction frames of different MV fields can complement each other. For instance, it improves the results on the *Bus*, *Mobile*, *Flower*, *Night*, *Spin Calendar*, and *City* sequences substantially. However, when certain scale of motion dominates the sequence, the MVs in certain motion field are always more accurate than those in other motion fields. In that case, the proposed scheme achieves little improvement, e.g., on the *News*, *Optis*, *Shuttle Start*, and *Big Ships* sequences. The average PSNR of different algorithms on CIF and 720P sequences are also presented in Table II. It can be seen that the proposed algorithm gains up to 1.15 dB on CIF sequences and 1.14 dB on 720P sequences over the MAAR scheme (which performs the best of the four benchmarks).

The subjective quality comparison is shown in Figs. 3–6 for the CIF sequence *Mobile* and *Bus* and the 720P sequence *Spin Calendar* and *City*. It can be observed from Fig. 3 that the numbers on calendar recovered by 3DRS, MCI and OBMC, and MAAR contain many annoying artifacts. On the other hand, the proposed scheme recovers these numbers visually pleasantly. In Fig. 4, the "barrier" and the "statuary" around the left boundary (both highlighted in red circle) are interpolated with apparent errors by the four benchmarks, while the proposed scheme recovers them gracefully. Likewise, in Figs. 5 and 6, the textures are recovered with annoying

TABLE IV
EFFECT OF USING MULTIHYPOTHESES FOR FRUC (DB)

CIF									
Case	*Football*	*Bus*	*Mobile*	*Stefan*	*Flower*	*Highway*	*Foreman*	*News*	Average
a)	23.23	29.32	31.75	29.52	33.09	33.70	35.59	37.85	*31.76*
b)	22.44	26.61	30.59	27.85	30.86	33.46	33.92	37.09	*30.35*
c)	22.72	27.11	31.00	28.64	31.65	33.79	34.86	37.33	*30.89*
d)	22.46	27.02	30.33	28.25	31.89	33.64	35.25	37.54	*30.79*
e)	21.82	24.14	30.14	27.24	32.63	33.28	35.22	37.66	*30.27*
720P									
Case	*Spin Calendar*	*City*	*Harbor*	*Night*	*Sailormen*	*Optis*	*Shuttle Start*	*Big Ships*	Average
a)	35.42	33.21	32.73	32.86	36.37	40.55	43.60	41.11	*36.98*
b)	35.29	33.40	32.15	31.53	35.86	40.28	43.59	40.86	*36.62*
c)	34.09	33.10	32.42	31.24	36.14	40.35	43.60	40.83	*36.47*
d)	33.74	32.64	32.49	30.39	36.24	40.43	43.61	40.94	*36.31*
e)	33.22	32.54	32.56	29.52	35.85	40.02	43.59	40.88	*36.02*

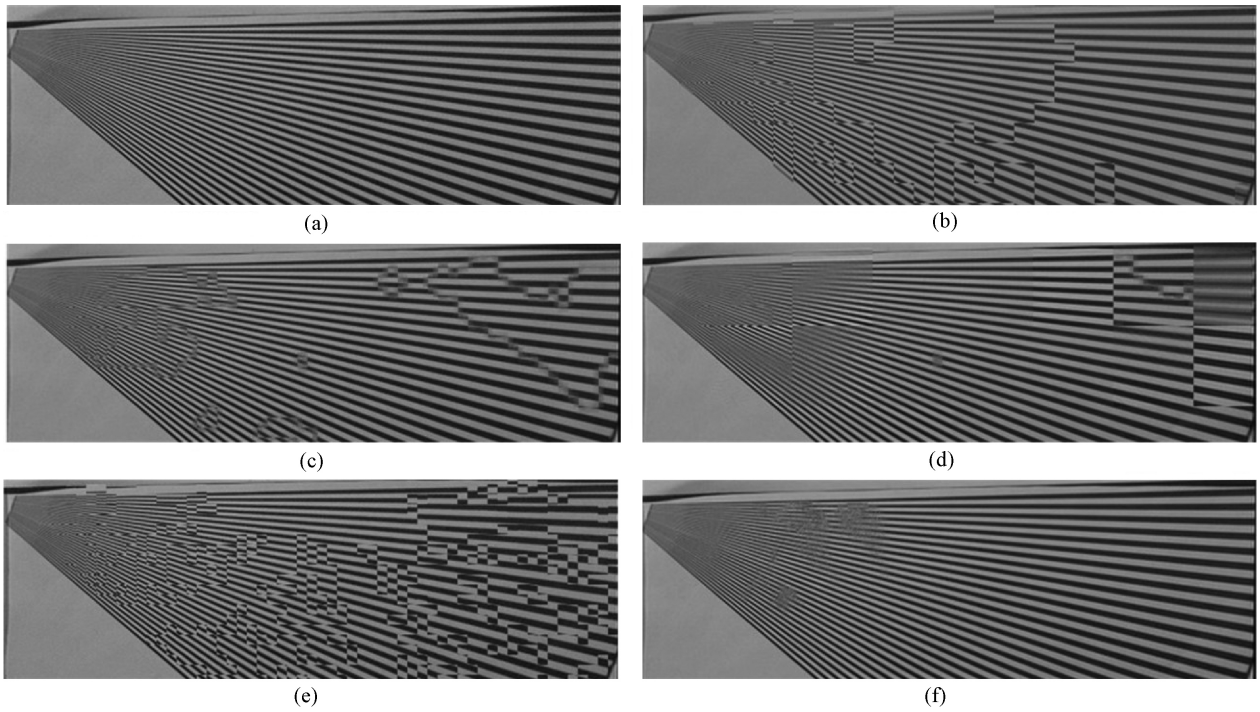


Fig. 5. FRUC results for *Spin Calendar* (second frame). (a) Original. (b) 3DRS. (c) OBMC. (d) MAAR. (e) DualME. (f) Proposed.

artifacts by the benchmarks schemes but are recovered very well by our scheme.

Performance improvement of the proposed scheme mainly attributes to the following factors. First, the multihypothesis Bayesian FRUC framework can discriminate reliable prediction pixels from unreliable prediction pixels, and use reliable prediction pixels but suppress unreliable pixels. Second, the proposed PRBME scheme can handle motion of variable sizes of objects.

Table III presents the average processing time (s/frame) of different FRUC methods on a typical personal computer (3.00 GHz Intel Core Duo CPU, 4 GB memory). It can be observed that the processing time of the proposed algorithm is much longer than that of 3DRS, OBMC, and DualME, on the other hand, the processing time of the proposed algorithm is 74.8% and 73.6% shorter than that of MAAR. Although the computational complexity of the proposed algorithm is higher than several existing MC-FRUC algorithms, it achieves much better quality than all four benchmark algorithms, thus it can be a good choice when computing capacity is powerful. On the other hand, the proposed algorithm can be speeded up by making tradeoff between the performance and the computational complexity. For instance, fast ME algorithm can be adopted, the number of ME steps in PRBME can be reduced, the forward and backward ME can be simplified to single directional ME, and so on.

We also study influence of multihypotheses on PSNR to understand its contribution more clearly. We present results of the following five cases in Table IV: a) block-size set $\{32 \times 32, 16 \times 16, 8 \times 8, 4 \times 4\}$ is used; b) only block size 32×32 is used; c) only block size 16×16 is used; d) only block size 8×8 is used; and e) only block size 4×4 is used. It can be observed from Table IV that the scheme with multiple hypotheses achieves the best results on both CIF

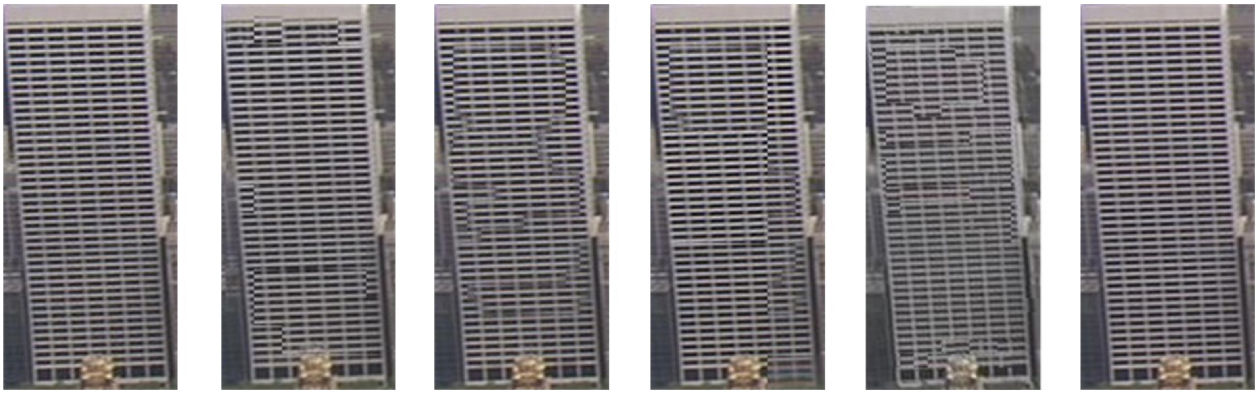


Fig. 6. FRUC results for *City* (tenth frame). (a) Original. (b) 3DRS. (c) OBMC. (d) MAAR. (e) DualME. (f) Proposed.

sequences and 720P sequences. For the scheme using single hypothesis, block sizes 16×16 and 8×8 work well for the CIF sequences while block sizes 32×32 and 16×16 work well for the 720P sequences. We can see that fixed matching block size cannot adapt to sequences of different resolutions. However, by fusing the estimations from multiple hypotheses properly, the proposed method works effectively on sequences with various resolutions and achieves the best results.

VI. Conclusion

In this paper, we proposed a multiple hypotheses Bayesian FRUC scheme. In this scheme, to estimate the intermediate frame with maximum *a posteriori* probability, both the temporal motion model and the spatial image prior model were incorporated into the optimization criterion. Instead of employing a single uniquely "optimal" motion field, multiple "optimal" motion fields were utilized. To obtain the multiple motion fields, a set of block-matching sizes was used and the motion fields were estimated by progressively reducing the size of matching block. The prediction frames generated by these multiple motion hypotheses were adaptively fused by modeling the relationship between the disturbance variance and the difference of reference pixels along the motion trajectories. Experimental results showed that although computational time of the proposed scheme is higher than several existing FRUC algorithms, the proposed method can significantly improve both the objective and the subjective quality of the constructed HFR video.

Acknowledgment

The authors would like to thank Dr. Y. Zhang for providing the implementation of the MAAR scheme [23]. They would also like to thank the anonymous reviewers for their very helpful suggestions that helped to improve the presentation of this paper.

References

[1] P. Haavisto, J. Juhola, and Y. Neuvo, "Fractional frame rate up conversion using weighted median filters," *IEEE Trans. Consumer Electron.*, vol. 35, no. 3, pp. 272–278, Aug. 1989.

[2] G. D. Haan, P. W. A. C. Biezen, H. Huijgen, and O. A. Ojo, "True motion estimation with 3-D recursive search block matching," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 5, pp. 368–379, Oct. 1993.

[3] O. A. Ojo and G. D. Haan, "Robust motion-compensated video upconversion," *IEEE Trans. Consumer Electron.*, vol. 43, no. 4, pp. 1045–1056, Nov. 1997.

[4] B.-W. Jeon, G.-I. Lee, S.-H. Lee, and R.-H. Park, "Coarse-to-fine frame interpolation for frame rate up-conversion using pyramid structure," *IEEE Trans. Consumer Electron.*, vol. 49, no. 3, pp. 499–508, Aug. 2003.

[5] C.-L. Huang and T.-T. Chai, "Motion-compensated interpolation for scan rate up-conversion," *Optic. Eng.*, vol. 35, no. 1, pp. 166–176, Jan. 1996.

[6] S.-H. Lee, O. Kwon, and R.-H. Park, "Weighted-adaptive motion-compensated frame rate up-conversion," *IEEE Trans. Consumer Electron.*, vol. 49, no. 3, pp. 485–492, Aug. 2003.

[7] G. Dane and T. Q. Nguyen, "Optimal temporal interpolation filter for motion-compensated frame rate up conversion," *IEEE Trans. Image Process.*, vol. 15, no. 4, pp. 978–991, Apr. 2006.

[8] Y. K. Chen, A. Vetro, H. Sun, and S. Y. Kung, "Frame-rate up-conversion using transmitted true motion vectors," in *Proc. IEEE Workshop Multimedia Signal Process.*, Dec. 1998, pp. 622–627.

[9] B.-D. Choi, J.-W. Han, C.-S. Kim, and S.-J. Ko, "Frame rate up-conversion using perspective transform," *IEEE Trans. Consumer Electron.*, vol. 52, no. 3, pp. 975–982, Aug. 2006.

[10] S.-J. Kang, K.-R. Cho, and Y. H. Kim, "Motion compensated frame rate up-conversion using extended bilateral motion estimation," *IEEE Trans. Consumer Electron.*, vol. 53, no. 4, pp. 1759–1767, Nov. 2007.

[11] S. C. Tai, Y. R. Chen, Z. B. Huang, and C. C. Wang, "A multi-pass true motion estimation scheme with motion vector propagation for frame rate up-conversion applications," *J. Display Tech.*, vol. 4, no. 2, pp. 188–197, Jun. 2008.

[12] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[13] A. H. Huang and T. Q. Nguyen, "A multistage motion vector processing method for motion-compensated frame interpolation," *IEEE Trans. Image Process.*, vol. 17, no. 5, pp. 694–708, May 2008.

[14] A. H. Huang and T. Q. Nguyen, "Correlation-based motion vector processing with adaptive interpolation scheme for motion-compensated frame interpolation," *IEEE Trans. Image Process.*, vol. 18, no. 4, pp. 740–752, Apr. 2009.

[15] S. J. Kang, S. J. Yoo, and Y. H. Kim, "Dual motion estimation for frame rate up-conversion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1909–1914, Dec. 2010.

[16] D. M. Wang, L. Zhang, and A. Vincent, "Motion-compensated frame rate up-conversion, part I: Fast multi-frame motion estimation," *IEEE Trans. Broadcast.*, vol. 56, no. 2, pp. 133–141, Jun. 2010.

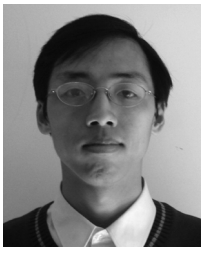
[17] D. M. Wang, A. Vincent, P. Blanchfield, and R. Klepko, "Motion-compensated frame rate up-conversion, part II: New algorithms for frame interpolation," *IEEE Trans. Broadcast.*, vol. 56, no. 2, pp. 142–149, Jun. 2010.

[18] R. Castagno, P. Haavisto, and G. Ramponi, "A method for motion adaptive frame rate up-conversion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 5, pp. 436–446, Oct. 1996.

[19] G. Dane and T. Nguyen, "Motion vector processing for frame rate up conversion," in *Proc. IEEE Int. Conf. Acou., Speech, Signal Process.*, May 2004, pp. 309–312.
[20] M. T. Orchard and G. J. Sullivan, "Overlapped block motion compensation: An estimation-theoretic approach," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 693–699, Sep. 1994.
[21] J. Zhai, K. Yu, J. Li, and S. Li, "A low complexity motion compensated frame interpolation method," in *Proc. ISCAS*, vol. 5. May 2005, pp. 4927–4930.
[22] B. D. Choi, J. W. Han, C. S. Kim, and S. J. Ko, "Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 7, pp. 407–416, Apr. 2007.
[23] Y. B. Zhang, D. B. Zhao, S. W. Ma, R. G. Wang, and W. Gao, "A motion-aligned auto-regressive model for frame rate up conversion," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1248–1258, May 2010.
[24] J. Konard and E. Dubois, "Bayesian estimation of motion vector fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 9, pp. 910–927, Sep. 1992.
[25] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, no. 6, pp. 721–741, Nov. 1984.
[26] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. Washington D.C.: V. H. Winston, 1977.

Hongbin Liu received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 2005, 2007, and 2011, respectively.

His current research interests include video processing, and image and video coding.

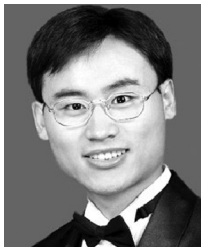


Ruiqin Xiong (M'08) received the B.S. degree from the University of Science and Technology of China, Hefei, China, in 2001, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2007.

He was a Research Intern with Microsoft Research Asia, Beijing, from August 2002 to July 2007, and a Senior Research Associate with the University of New South Wales, Sydney, Australia, from September 2007 to August 2009. Later, he joined the Institute of Digital Media, School of Electronic Engineering and Computer Science, Peking University, Beijing, where he is currently a Research Professor. His current research interests include image and video processing, compression, multimedia communication, channel coding, and distributed coding.

Debin Zhao (M'11) received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1985, 1988, and 1998, respectively.

He is currently a Professor with the Department of Computer Science, Harbin Institute of Technology. He has published over 200 technical articles in refereed journals and conference proceedings. His current research interests include image and video coding, video processing, video streaming and transmission, and pattern recognition.



Siwei Ma (S'03–M'12) received the B.S. degree from Shandong Normal University, Jinan, China, in 1999, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005.

From 2005 to 2007, he was a Post-Doctoral Researcher with the University of Southern California, Los Angeles. Later, he joined the Institute of Digital Media, Department of Electrical Engineering and Computer Science, Peking University, Beijing, where he is currently an Associate Professor. He has published over 70 technical articles in refereed journals and proceedings. His current research interests include image and video coding, video processing, video streaming, and transmission.

Wen Gao (M'92–SM'05–F'09) received the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991.

He is currently a Professor of computer science with Peking University, Beijing, China. He was a Professor with the Harbin Institute of Technology, Harbin, China, from 1991 to 1995, and a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, from 1996 to 2006. He has published extensively including five books and over 600 technical articles in refereed journals and conference proceedings. His current research interests include image processing, video coding and communication, pattern recognition, multimedia information retrieval, multimodal interface, and bioinformatics.

Prof. Gao was or is an Editorial Board Member of several journals, such as the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON AUTONOMOUS MENTAL DEVELOPMENT, the *EURASIP Journal of Image Communications*, and the *Journal of Visual Communication and Image Representation*. He was the chair of a number of prestigious international conferences on multimedia and video signal processing, such as IEEE ICME and ACM Multimedia. He was also an Advisory and Technical Committee Member of numerous professional organizations. He is a fellow of the Chinese Academy of Engineering.