Journal of Information Hiding and Multimedia Signal Processing ©2016 ISSN 2073-4212
Ubiquitous International Volume **7**, Number **6**, November **2016**

Novel Frame Rate Up-Conversion Method Through Weight Matching Criterion and Motion Vector Refinement

*Zongliang Gan, Changhong Chen, Ziguan Cui, and Feng liu

The College of Telecommunications and Information Engineering
Nanjing University of Posts and Telecommunications
No.66 Xin Mo Fan Road, 210003, NanJing, China
*Corresponding Author: ganzl@njupt.edu.cn

Received July, 2015; revised August, 2016

Abstract. *This paper presents a novel frame rate up-conversion (FRUC) framework using regulations matching criterion. Motion estimation is one of the key elements in FRUC, and the regularization matching criterion using the difference of Gaussians (DOG) is proposed to improve the motion estimation accuracy. The proposed FRUC framework has three steps. First, the initial motion vector field is calculated through the unidirectional motion estimation by two-pass neighbor recursive search. Second, motion vector refinement is used to get the more reliable motion vector through the bi-direction estimation and motion vector postprocessing. Finally, the intermediate frame is reconstructed by linear motion compensation interpolation or overlapped block motion compensation according to the matching energy. The experimental results show that the proposed method can achieve comparable performance to other competitive algorithms with low complexity.*

Keywords: Frame rate up-conversion, Motion estimation, Regularization, Frame interpolation

1. **Introduction.** Frame rate up-conversion (FRUC) technology can increase the temporal resolution of the video, and can be widely adopted in video communication with limited bandwidth, liquid crystal displays (LCDs) ghost phenomenon eliminating, and video edit. In general, there are two kinds of FRUC algorithm. One is very simple strategy, including methods such as frame repetition or linear interpolation. The other kind method, which is the motion-compensated frame rate up-conversion (MC-FRUC), have been more widely used for its good performance. Different from the video compression, motion estimation in MC-FRUC is aim to find the true trajectory rather than minimum prediction error. There are two types of motion estimation method in MC-FRUC: optical flow estimation and block matching estimation. Mahajan at al. [1] use the optical flow estimation to get the remarkable interpolated frame by moving gradients and using Poisson reconstruction. Although optical flow estimation can provide the reliable motion vector, the computational complex is too high. Hence, the most MC-FRUC methods use the block matching to find the motion vectors (MV) due to its lower complexity.

In general, there are three research aspects in MC-FRUC: motion trajectory estimation, matching energy criterion and intermediate frame interpolation. For motion trajectory estimation, there are two basic methods: unidirectional [2] [3] [4] [5] [6] [7] [8] and bi-directional estimation [9] [10] [11] [12] [13] [14] . The unidirectional trajectory estimation makes all motion vectors pass through the frame to intermediate frame in one direction, and this inevitably bring holes and overlaps in MV field. Hence, the median filter, or spatial interpolation can be used to handle above problem. For bi-directional estimation approach, the intermediate frame is divided into blocks before it is actually predicted, and each block has two symmetrical motion vectors, in which one pointing to the pervious frame and the other to the next frame. In this way, the bi-directional motion estimation is able to avoid the problem of the holes and overlaps.

In MC-FRUC, the matching energy criterion plays an important role to find the true motion vectors. The ordinary matching criterion use the sum absolute different (SAD) and sum squared different (SSD), which may not work well in MC-FRUC. Usually the temporal and spatial smoothness of MV field is considered as priori knowledge, and maximum a posteriori (MAP) estimation can be used to get more real motion vector by Markov random field assumption. Since MAP based MV estimation such as [3] need iterative algorithm to find optimal solution, the complexity of the motion estimation is high and currently it is hard to use this approach for the practical application. Furthermore, several relatively simple and effective estimation criterion models [2] [5] [7] [10] [11] [12] [13] [14] have been proposed. Haan at al. [2] proposed the classical 3-D recursive search (3DRS) method using the spatial and temporal neighbor blocks to get the smooth MV field with very low complexity, and Han at al. [7] improve the spatial and temporal candidates by MV retiming and localized global motion. Other authors put forward their own motion estimation match criteria algorithms using the neighbor blocks correlation or regularization weight of the MV [5] [10] [11] [12] [14]. Moreover, variable-size motion estimation (VS-ME) and abnormal motion vector rectify are also useful to get more reasonable MV, but it should be noted that the VS-ME computational cost is still too high for the practical applications.

After getting MV field, the intermediate frame interpolation will be done. The simplest method is linear interpolation using MV. When the MV field is no consistent, the overlapped block motion compensation (OBMC) can suppress the blocking artifacts. The more complex approach, such as trilateral filter [5], spatio-temporal auto-regressive (STAR) mode [8], and multiple hypotheses adaptive fusion [15], can obtain better quality interpolation frame with the cost of complexity.

In this paper, we mainly focus on the low complex MC-FRUC method. Motivated by the above analysis, match criterion is very important to find the reliable MV, and the smooth property of MV field is also critical to decrease the artifacts, in which the most of block artifacts is generated by the MV field discontinuity. The motion estimation of the proposed MC-FRUC framework is based on weighted matching criterion , which use the difference of Gaussians (DOG) of the block as regularization term. Overall, the proposed FRUC method have three steps. First, the unidirectional motion estimation is used to find the initial MV field through two-pass neighbor recursive search. Second, motion vector refinement is performed to get the more reliable motion vector, in which the bi-direction estimation and motion vector postprocessing are adopted sequentially. Finally, the intermediate frame is gotten by linear motion compensation interpolation (MCI) or OMBC according to the matching energy. Although the our approach has low complex, the experimental results are shown that the proposed method can get comparable subjective and objective quality, compared with the other leading methods.

2. **Regularization Motion Estimation Matching Criterion.** The accuracy of the motion vectors is very important to MC-FRUC. Therefore a series of literature propose different motion estimation criterion. In [5], edge information with high frequency data is used to the regularization matching. However, the high frequency of the image sometime

is affected by noise, specifically quantization noise caused by video coding. Inspired by [5], we use the middle frequency of the frame as regularization information for motion estimation to avoid high-frequency noise affect. The DOG feature is very suitable as regularization item in the MC-FRUC, since the human retina extracts details from images using DOG of the various sizes and encodes such differences with action potentials from human visual perspective [16]. In our tests, the DOG feature can more robust describe texture and boundary information of the frame, and it can be used as a regularization term to find the more accurate motion vectors. Essentially, the DOG feature is a band-pass filter, which works by performing two different Gaussian blurs on the image, subtracting them to yield the result. The DOG of the given frame $f(l)$ can be defined as:

$$bp(l) = \mathrm{GF}_{\sigma 1}[f(l)] - \mathrm{GF}_{\sigma 2}[f(l)] \tag{1}$$

where the GF_σ is Gaussian convolution with σ. After extensive testing on about 60000 frames, σ_1 is set to 0.6, σ_2 is set to 1.5, and the filter window is set to 7×7 window.

Then, with the DOG feature, the matching error criterion of the current block between the two frames can be defined as:

$$\begin{aligned} E(f_{t-1}, f_{t+1}; \overrightarrow{\mathbf{mv}}) = \tfrac{1}{num}[\sum_{l \in B} \mathrm{K} \cdot |f_{t-1}(\mathbf{l}) - f_{t+1}(\mathbf{l} + \overrightarrow{\mathbf{mv}})| \\ +\lambda \sum_{l \in B} |bf_{t-1}(\mathbf{l}) - bf_{t+1}(\mathbf{l} + \overrightarrow{\mathbf{mv}})|] \end{aligned} \tag{2}$$

where $\mathbf{l}$ is the pixel coordinate (x, y), num is the pixel number in the block, and $bp(l)$ is DOG filter defined in (2). There are two regularization parameters K and λ.

Firstly, K can be described as follows:

$$K = \begin{cases} 0; if \left| f_{t-1}(\mathbf{l}) - f_{t-1}(\mathbf{l} + \overrightarrow{\mathbf{mv}}) \right| < 3; \\ 1; if \left| f_{t-1}(\mathbf{l}) - f_{t-1}(\mathbf{l} + \overrightarrow{\mathbf{mv}}) \right| < 25; \\ 2; else. \end{cases} \tag{3}$$

where K is first regularization parameter, and it is used to correcting domain pixel matching error. If the two pixels in two frames are with similar values, it seems to there two pixels belong to the same object and the difference caused by the noise can be eliminated. If the two pixels in two frames are with very different values, these two pixels seem to belong two different object and the match error tend to be given with bigger weight. With the lots of test, the empirical weight is set as follows: when the difference of two pixels are within 3, K is set to 0 and these small differences are not considered; if the difference of two pixels are more then 25, K is set to 2.

The λ in (2) is second regularization parameter and it tunes the influences of the two distance terms in (2). The two distance terms measure the original pixel domain matching error and the middle frequency matching error respectively. If λ is zeros, the equation (2) became conventional motion estimation match. Although the optimized λ is changed with the frame content, the precise prediction of λ need a large number of calculations. In this paper, we empirical set $\lambda = 1.2$ with the lots of test.

Finally, the MV of the block B in f_t can be gotten by minimize the matching error criterion as:

$$\overrightarrow{\mathbf{mv}}(B(i)) = \underset{f_{t+1}(l+\overrightarrow{\mathbf{mv}}) \in S}{\arg\min} \left(E(f_{t-1}, f_{t+1}; \overrightarrow{\mathbf{mv}})\right) \tag{4}$$

where S is search window or candidate set in motion estimation.

3. **The Proposed MC-FRUC Method.** The framework of the proposed MC-FRUC method is presented in Fig. 1, and the intermediate frame f_t is predict by the successive frame f_{t-1} and f_{t+1} in the same video scene. The first step is to determine the initial motion vector field through unidirectional motion estimation, which has two-pass process

include the positive scan and inverse scan search. Then, based on above results, bi-directional motion estimation and motion vector postprocessing are performed to get the more reasonable motion vector. In this way, the motion vector refinement processing can obtain the smooth MV field. Finally, the intermediate frame is reconstructed by linear MCI or OBMC according to the matching energy.

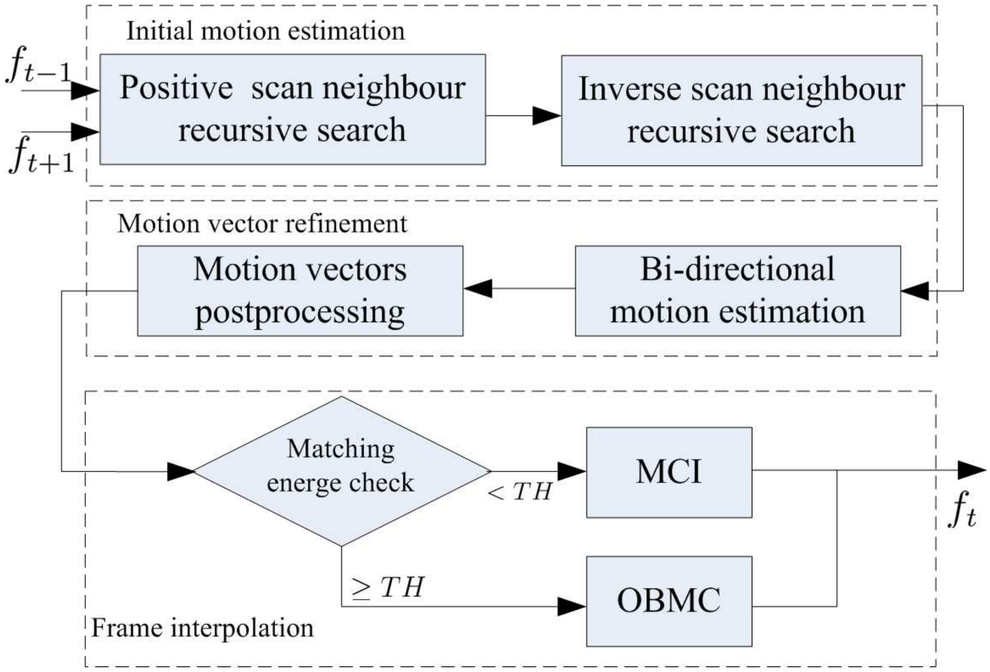

FIGURE 1. Flowchart of the proposed MC-FRUC scheme

3.1. **Initial motion estimation.** Firstly, the unidirectional motion estimation is adopted to roughly predict the motion vectors between the f_{t-1} and f_{t+1} in the proposed method. In the most of existing MC-FRUC, the motion vector relevance of the spatial or temporal neighbor blocks is the key assumption to motion estimation algorithm design, and a series of recursive search algorithm are proposed, including the 3DRS [2], true-motion estimation (TME) [6], retiming temporal and spatial candidate recursive search [7]. Here, we want keep the algorithm simply and effectively. Therefore, our algorithm only uses the similarity of neighbor spatial block MV. In general, motion estimation is performed block by block in accordance with the raster scan order, which scans from left to right and from top to bottom. If only one-pass search is used, it always has some invalid neighbor blocks MV. Hence, the two-pass recursive search is presented to make the all neighbor blocks to participate MV smooth constraint, as shown in Fig.2. In each pass, there are four motion vectors of neighbor block as reference motion vector. The entire MV candidate set can be respective defined as:

$$CS^{(1)} = C(\overrightarrow{\mathbf{mv}}_1) \cup C(\overrightarrow{\mathbf{mv}}_2) \cup C(\overrightarrow{\mathbf{mv}}_3) \cup C(\overrightarrow{\mathbf{mv}}_4) \tag{5}$$

$$CS^{(2)} = C(\overrightarrow{\mathbf{mv}}_5) \cup C(\overrightarrow{\mathbf{mv}}_6) \cup C(\overrightarrow{\mathbf{mv}}_7) \cup C(\overrightarrow{\mathbf{mv}}_8) \tag{6}$$

where $CS^{(1)}$ is MV candidate set for the first pass search, $CS^{(2)}$ for second pass, and $C(\overrightarrow{\mathbf{mv}}_i)$ is the search MV set of neighbors reference motion vector $\overrightarrow{\mathbf{mv}}_i$.

For a given motion vector $\overrightarrow{\mathbf{mv}}_i$ with (mvx_i, mvy_i), it's search MV set $C(\overrightarrow{\mathbf{mv}}_i)$ can be described as follow:

$$C(\overrightarrow{\mathbf{mv}}_i) = \bigcup_{i,j} (\overrightarrow{\mathbf{mv}}_{ci} + \mathbf{\Delta}_j) \tag{7}$$

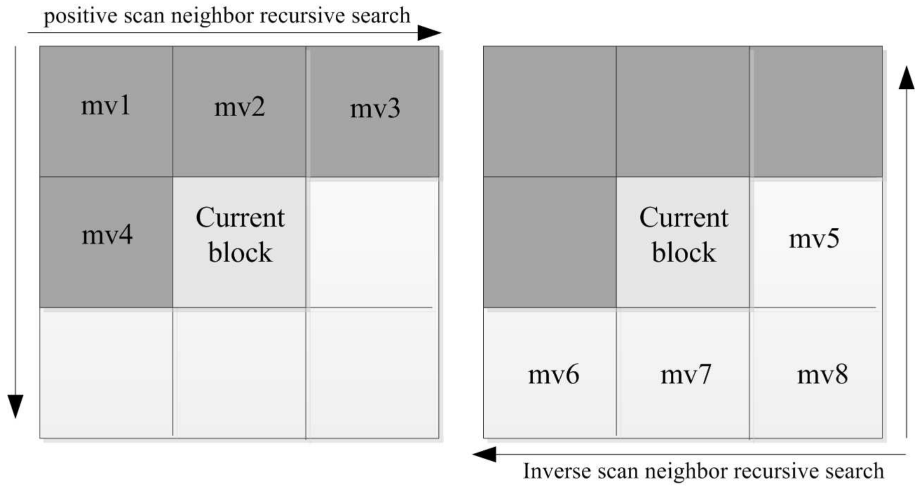

FIGURE 2. The two-pass neighbor recursive search

where the candidate vector set $\overrightarrow{\mathbf{mv}}_{ci}$ and update variable $\mathbf{\Delta}_j$ are given as follows:

$$\overrightarrow{\mathbf{mv}}_{ci} = \left\{ \begin{pmatrix} mvx_i \\ mvy_i \end{pmatrix}, \begin{pmatrix} mvx_i \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ mvy_i \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \end{pmatrix} \right\} \tag{8}$$

$$\mathbf{\Delta}_j = \left\{ \begin{pmatrix} \pm 1 \\ 0 \end{pmatrix}, \begin{pmatrix} \pm 2 \\ 0 \end{pmatrix}, \begin{pmatrix} \pm 3 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \pm 1 \end{pmatrix}, \begin{pmatrix} 0 \\ \pm 2 \end{pmatrix} \right\} \tag{9}$$

Once MV candidate set $CS^{(1)}$ and $CS^{(2)}$ is gotten in each pass motion estimation, the MV of current block between the f_{t-1} and f_{t+1} can be calculated by Eq.(2). Though the recursive search can be processed in iteration manner, non-iterative recursive search is used to keep the algorithm with low complex and effective in this paper.

3.2. **Motion estimation refinement.** In the proposed MC-FRUC method, motion estimation refinement aims to get the more fine MV field using the initial MV between the f_{t-1} and f_{t+1}. Assuming the object with constant velocity, the relationship between the three frame f_{t-1}, f_t and f_{t+1} can be given as follows:

$$\begin{aligned} f_{t-1}(\mathbf{l}) &= f_t(\mathbf{l} + 0.5 \cdot \overrightarrow{\mathbf{mv}}(\mathbf{l})) + n_t \\ &= f_{t+1}(\mathbf{l} + \overrightarrow{\mathbf{mv}}(\mathbf{l})) + n_{t+1} \end{aligned} \tag{10}$$

where n_t and n_{t+1} are noise, and $f_t(\mathbf{l} + 0.5 \cdot \overrightarrow{\mathbf{mv}}(\mathbf{l}))$ is called the virtual pixel in f_t in this paper. It means that the pixel $f_{t-1}(\mathbf{l})$ in f_{t-1} move to the virtual pixel $f_t(\mathbf{l} + 0.5 \cdot \overrightarrow{\mathbf{mv}}(\mathbf{l}))$ in f_t, and to $f_{t+1}(\mathbf{l} + \overrightarrow{\mathbf{mv}}(\mathbf{l}))$ in f_{t+1}. Hence, for the block $\mathbf{B}(i)$ in f_t, it's rough MV can be gotten as follows:

$$\overrightarrow{\mathbf{mv}}_0(B(i)) = \frac{1}{n(b)} \sum_{f_t(\mathbf{l}+0.5\cdot\overrightarrow{\mathbf{mv}}(\mathbf{l}))\in B(i)} 0.5 \cdot \overrightarrow{\mathbf{mv}}(\mathbf{l}) \tag{11}$$

where $n(b)$ is the number of the virtual pixel belong to the block $\mathbf{B}(i)$.

Then the bi-direction estimation with regularization matching can get more accuracy MV. And the search window is $mvx \in [-4 : +4]$ and $mvy \in [-3 : +3]$ with the center of $\overrightarrow{\mathbf{mv}}_0(B(i))$, taking into account the horizontal movement of the object is more common and range in the daily video. In this way, the computation cost can be saved without loss of performance. Then the matching criterion as:

$$\begin{aligned}
&\overrightarrow{\mathbf{mv}}(B(i)) \\
&= \arg\min_{\overrightarrow{\mathbf{mv}}\in S}\{\frac{1}{num}[\sum_{l\in B}\mathrm{K}\cdot|f_{t-1}(\mathbf{l}-\overrightarrow{\mathbf{mv}})-f_{t+1}(\mathbf{l}+\overrightarrow{\mathbf{mv}})| \\
&+\lambda\sum_{l\in B}|bf_{t-1}(\mathbf{l}-\overrightarrow{\mathbf{mv}})-bf_{t+1}(\mathbf{l}+\overrightarrow{\mathbf{mv}})|]+W\}
\end{aligned} \tag{12}$$

where K in (12) is same to Eq.(3). There are two regularization item in (12). The first item is DOG feature. The second item is W, and it has two factors. One is penalty term of deviation from initial $\overrightarrow{\mathbf{mv}}$ and the other is deviation from $(0, 0)$. Thus, the K can be defined as follows:

$$W = 4\left\|\overrightarrow{\mathbf{mv}}-\overrightarrow{\mathbf{mv}}_0\right\|_2 + 2\max\{\max(mvx, mvy)-2, 0\} \tag{13}$$

In the MC-FRUC, once the MV field appears to be discontinuity, the artifacts may be appear. Here, the MV postprocessing is used to make more smooth MV field in order to better subjective visual effects. The reliability of the block $\overrightarrow{\mathbf{mv}}$ is defined by:

$$R(\overrightarrow{\mathbf{mv}}) = \begin{cases} 0.1, DOB > th1 \ \ \& \ \ MVD > th2; \\ 1, DOB < th1 \ \ \& \ \ MVD < th2; \\ 0.75, others. \end{cases} \tag{14}$$

In (14), the DOB is given by:

$$DOB(\overrightarrow{\mathbf{mv}}) = \frac{1}{num}\sum_{l\in B}|f_{t-1}(l-\overrightarrow{\mathbf{mv}})-f_{t+1}(l+\overrightarrow{\mathbf{mv}})| \tag{15}$$

and the MVD is the measure for the similarity of neighborhood blocks MV, by:

$$MVD(\overrightarrow{\mathbf{mv}}(Bi)) = \sum_{\overrightarrow{\mathbf{mv}}_i\in N(Bi)}|\overrightarrow{\mathbf{mv}}_i-\overrightarrow{\mathbf{mv}}| \tag{16}$$

In this paper, based on the experimental results, $th1$ is set as 12, and $th2$ as 6.4.

Finally, for given block $B(i)$, it's all neighbor blocks including its own are involved in the MV postprocessing by:

$$\overrightarrow{\mathbf{mv}}_p(B(i)) = \frac{\sum_{i\in N}\frac{R(B(i))\cdot\overrightarrow{\mathbf{mv}}_i(B(i))}{DOB(B(i))}}{\sum_{i\in N}\frac{R(B(i))}{DOB(B(i))}} \tag{17}$$

3.3. **Frame interpolation.** When the MV in the frame f_t is gotten by Eq.(17), the intermediate frame will be reconstructed. Since the bi-directional motion estimation is used, the problem of the holes and overlaps can be avoided. However, the occlusion, which refers to the appearance of new objects and disappearance of existing objects when comparing adjacent frames, is another difficult problem. In our interpretation, when occlusion is happen to the block in f_t, the matching error of the block tend to become larger. In this case, the frame interpolation is more like video inpainting due to lack the original information of intermediate frame. Data fusion approach, such as [5] [8] [15], can better exploit the temporal correlation and deal with occluded areas. In addition, variable-size motion estimation also can reduce the influence of occlusion. Nevertheless, the main disadvantage of above methods is high computational complexity.

Since this paper mainly focus on the low complex method of MC-FRUC, the OBMC method is a rational solution to the occlusion problems because this approach not only increases prediction accuracy but also reduce blocking artifacts with acceptable computational complexity. In this paper, two methods, include linear MCI and OBMC, are

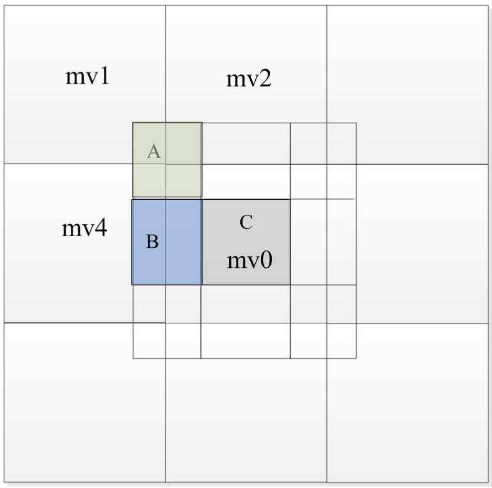

FIGURE 3. Illustration of OBMC in the proposed MC-FRUC

used to reconstruct the intermediate frame according to the matching energy error. If the DOB of the block less than $th3$, the linear MCI is used. Otherwise the OBMC is used. The threshold $th3$ in this paper is given based on the experimental results, as:

$$th3 = 2.9 \cdot \left(\frac{\sum\limits_{B(i)\in f_t} DOB(\overrightarrow{\mathbf{mv}}_p(B(i)))}{num(B(i))}\right) \tag{18}$$

The pixel $f_t(\mathbf{l})$ can be interpolated through linear MCI, as follows:

$$f_t(\mathbf{l}) = \frac{1}{2}(f_{t-1}(\mathbf{l} - \overrightarrow{\mathbf{mv}}_p) + f_{t+1}(\mathbf{l} + \overrightarrow{\mathbf{mv}}_p)) \tag{19}$$

The OBMC predict is illustrated in Fig.3, there are three different regions. The pixels in region A, which overlaps the four blocks are interpolated by

$$f_t(\mathbf{l}) = \frac{\sum\limits_{i=0,1,2,4} (MVD(\overrightarrow{\mathbf{mv}}_p(i)) \cdot f_t(\mathbf{l})^{(i)})}{\sum\limits_{i=0,1,2,4} MVD(\overrightarrow{\mathbf{mv}}_p(i))} \tag{20}$$

where the $f_t(\mathbf{l})^{(i)}$ is defined by:

$$f_t(\mathbf{l})^{(i)} = f_{t-1}(\mathbf{l} - \overrightarrow{\mathbf{mv}}_p(i)) + f_{t+1}(\mathbf{l} + \overrightarrow{\mathbf{mv}}_p(i)) \tag{21}$$

The pixels in region B, which overlaps two blocks, are interpolated by

$$f_t(\mathbf{l}) = \frac{\sum\limits_{i=0,4} (MVD(\overrightarrow{\mathbf{mv}}_p(i)) \cdot f_t(\mathbf{l})^{(i)})}{\sum\limits_{i=0,4} MVD(\overrightarrow{\mathbf{mv}}_p(i))} \tag{22}$$

The pixels in region C, which overlaps only one block, are interpolated through linear MCI by Eq.(19).

4. **Experimental Results.** In this section, we perform experiments of eight standard video, which are four video sequences with CIF (352×288), three video sequences with 720P (1280×720) and one video sequence with 1080p (1920×1080), to demonstrate the efficacy of the proposed method. Six MC-FRUC methods, which are 3DRS [2], BME [9], TF [5], DME [12], HD [7] and MHAF [15], are used for comparison. To measure the quality of the interpolated frames, the 60 even frames is removed and reconstruct them from the related 61 odd frames for each sequence using MC-FRUC techniques, and then

compare the reconstructed frames to the original frames. The three 720P video sequences are tested from the third frame since the first two frames are noise data, and the other sequences are tested from the first frame. The test platform is all the C codec for seven kind FRUC methods on the intel I7 2620(m) CPU with 8GB memory.

In the experiments, the motion search range of BME, TF, DME and HD is seted to $[-18 : +18]$, and motion estimation block size used in these benchmarks is 8×8. There are three iterations in the 3DRS method with block size 16×16. In MHAF, the block size varies from 32×32 to 4×4. In the tests, two size blocks with 8×8 and 16×16 are used and the overlap size is three for OBMC in the proposed method. Only the luminance channel is used for comparison, the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM)[17] are computed between the ground truth (original frame) and the interpolation frame and the results of the average $\overline{PSNR}$ (dB), $\overline{SSIM}$ and run time (s) per frame are shown in Table 1. From the Table 1, we can see that the

TABLE 1. $\overline{PSNR}$ (dB) (top), $\overline{SSIM}$ (middle) and the average run time (s) per frame (bottom) results (in luminance) of the interpolated frames.

squence	format	3DRS[2]	BME[9]	TF [5]	DME[12]	HD [7]	MHP[15]	Proposed 8×8	Proposed 16×16
Bus	CIF	25.89	25.83	25.95	26.01	25.60	26.72	26.35	26.18
		0.9192	0.9327	0.9382	0.9388	0.9154	0.9401	0.9391	0.9354
		0.020	0.150	80.01	0.171	0.161	4.021	0.102	0.096
City	CIF	32.11	32.19	32.45	32.91	33.01	34.2	33.52	33.70
		0.9493	0.9566	0.9571	0.9598	0.9612	0.9711	0.9622	0.9606
		0.021	0.157	98.017	0.192	0.177	6.122	0.110	0.96
Football	CIF	21.35	22.01	22.07	22.37	21.73	23.95	23.17	22.72
		0.7792	0.7962	0.7953	0.8012	0.7951	0.831	0.8259	0.8095
		0.030	0.153	100.015	0.187	0.197	5.349	0.103	0.97
News	CIF	36.59	36.62	36.64	35.73	36.82	37.91	37.52	37.02
		0.9821	0.9828	0.9810	0.9770	0.9812	0.9892	0.9838	0.9828
		0.0015	0.122	76.323	0.165	0.154	3.941	0.085	0.072
Mobcal	720p	33.62	34.34	33.01	34.53	34.11	35.19	34.98	35.20
		0.9961	0.9952	0.9938	0.9957	0.9951	0.9982	0.9973	0.9983
		0.133	1.010	721.470	1.493	1.386	32.117	0.776	0.649
Shields	720p	33.95	33.55	33.27	34.12	34.1	34.60	34.32	34.65
		0.9945	0.9901	0.9921	0.9951	0.9950	0.9961	0.9953	0.9962
		0.125	1.130	698.114	1.541	1.421	34.322	0.721	0.684
Parkrun	720p	31.71	31.53	31.61	31.82	31.98	32.39	32.25	32.41
		0.9902	0.9856	0.9891	0.9912	0.9907	0.9925	0.9910	0.9924
		0.102	1.126	714.52	1.651	1.329	33.214	0.728	0.704
Tractor	1080p	33.27	33.21	33.4	33.9	34.12	34.50	34.27	34.48
		0.9917	0.9912	0.9914	0.9921	0.9924	0.9973	0.9961	0.9977
		0.230	2.536	1609.463	3.721	2.998	74.771	1.641	1.586
Total average $\overline{PSNR}$		31.06	31.16	31.05	31.42	31.43	32.43	32.03	32.15
Total average $\overline{SSIM}$		0.9503	0.9538	0.9548	0.9564	0.9533	0.9644	0.9613	0.9591
Run time rank		1	2	8	6	5	7	4	3

proposed method perform better than 3DRS[2], BME[9], TF [5], DME [12], and HD [7]. The MHF method [15] is better than the proposed method when the sequence format is CIF, and have almost the same performance level when the sequence is high resolution. It is not surprising that the proposed method is worse than the MHF method [15] as the whole, since multiple hypotheses Bayesian estimation and VS-ME are used in MHF with increasing computational complexity. From the results, for motion estimation block size

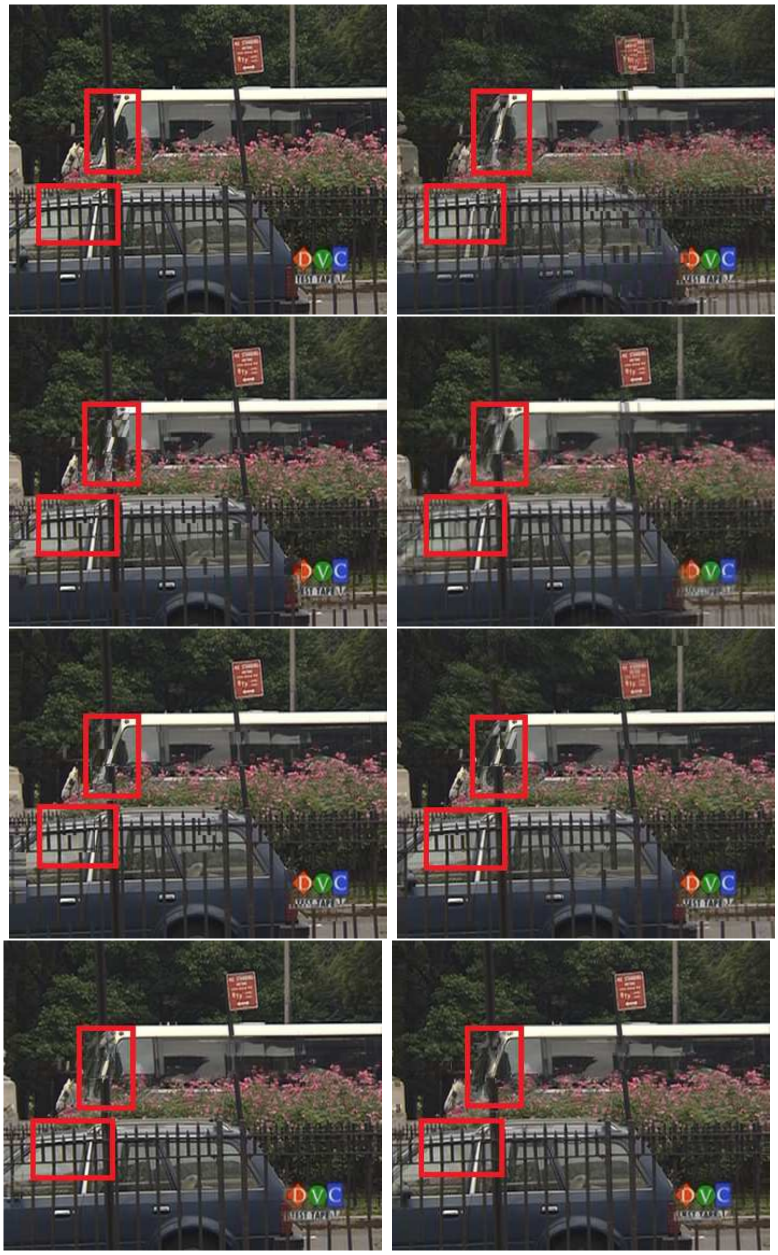

FIGURE 4. FRUC results for Bus (4th frame).Top row: Original and 3DRS[2]; Second row: BME[9] and TF[5]; Third row: DME [12], HD [7] ; Bottom row: MHP [15], Proposed method with 8×8 block.

in the proposed MC-FRUC, 16×16 size block tend to be better when the video is high definition(HD) format. We try the 32×32 and 64×64 block in the proposed MC-FRUC for the HD test video, and the results become worse. The probable reason may be that although the bigger size block used in motion estimation can get more precise motion vector in the interior areas of the object for the HD video, but it get worse in the edge areas. The 16×16 size block can achieve better compromise for HD video.

Since the adaptive content processing strategy is used in the proposed method, it is difficult to give the computational complexity of the algorithm systematically like other FRUC methods. Table 1 also shows the processing time(s) for 8 video sequence using different algorithms. The computing time of 3DRS method is minimum. The proposed method has rank 3 and has low complexity compared with other methods. The TF [5] method have the maximum run time because the trilateral filtering are used and it takes the most computation cost.

Fig. 4 shows an example with the interpolation results of the Bus sequence, it can be seen that the interpolated frame by the proposed method have less artifacts compared with other methods. In addition, it should be noted that the occlusion is still the difficult problem to MC-FRUC at present. For example, the upper red boxes in Fig.4 show that all algorithms fail to reconstruct the marked occlusion area.

5. **Conclusion.** In this paper, we proposed a novel MC-FRUC framework based on the regularization motion estimation matching criterion. The contributions of our work are two-fold. First, the DOG feature is used as a regularization term for match criterion to find the more accurate motion vectors. Second, we redesigned the MC-FRUC framework with low complexity, which has three steps: initial motion estimation, motion estimation refinement and frame interpolation. Experiment results show the advantage of our approach.

Acknowledgment. This research was supported in part by the National Nature Science Foundation, P. R. China. (No. 1172118,61471201,61501260),Jiangsu Province Universities Natural Science Research Key Grant Project (No. 13KJA510004), Natural Science Foundation of Jiangsu Province (BK20130867), the Six Kinds Peak Talents Plan Project of Jiangsu Province (2014-DZXX-008), and the "1311" Talent Plan of NUPT.

REFERENCES

[1] D. Mahajan, F. C. Huang, W. Matusik, R. Ramamoorthi, and P. Belhumeur, "Moving gradients: a path-based method for plausible image interpolation," *ACM Transactions on Graphics (TOG)*, vol. 28, no. 3, pp. 42, 2009.

[2] G. D. Haan, P. W. Biezen, H. Huijgen, and O. A. Ojo, "True-motion estimation with 3-d recursive search block matching," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 3, no. 5, pp. 368–379, 1993.

[3] D. Wang, L. Zhang, and A. Vincent, "Motion-compensated frame rate up-conversionpart i: Fast multi-frame motion estimation," *Broadcasting, IEEE Transactions on*, vol. 56, no. 2, pp. 133–141, 2010.

[4] D. Wang, A. Vincent, P. Blanchfield, and R. Klepko, "Motion-compensated frame rate up-conversionpart ii: New algorithms for frame interpolation," *Broadcasting, IEEE Transactions on*, vol. 56, no. 2, pp. 142–149, 2010.

[5] C. Wang, L. Zhang, Y. He, and Y. P. Tan, "Frame rate up-conversion using trilateral filtering," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 6, pp. 886–893, 2010.

[6] S. Dikbas and Y. Altunbasak, "Novel true-motion estimation algorithm and its application to motion-compensated temporal frame interpolation," *Image Processing, IEEE Transactions on*, vol. 22, no. 8, pp. 2931–2945, 2013.

[7] R. Han and A. D. Men, "Frame rate up-conversion for high-definition video applications," *Consumer Electronics, IEEE Transactions on*, vol. 59, no. 1, pp. 229–236, 2013.

[8] Y. B. Zhang, D. B. Zhao, S. W. Ma, R. Q. Wang, and W. Gao, "A motion-aligned auto-regressive model for frame rate up conversion," *Image Processing, IEEE Transactions on*, vol. 19, no. 5, pp. 1248–1258, 2010.

[9] B.-T. Choi, S.-H. Lee, and S.-J. Ko, "New frame rate up-conversion using bi-directional motion estimation," *Consumer Electronics, IEEE Transactions on*, vol. 46, no. 3, pp. 603–609, 2000.

[10] T. Ha, S. Lee, and J. Kim, "Motion compensated frame interpolation by new block-based motion estimation algorithm," *Consumer Electronics, IEEE Transactions on*, vol. 50, no. 2, pp. 752–759, 2004.

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

[12] S.-J. Kang, S. Yoo, and Y. H. Kim, “Dual motion estimation for frame rate up-conversion,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 12, pp. 1909–1914, 2010.

[13] 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,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 4, pp. 407–416, 2007.

[14] W. Bai, J. Liu, J. Ren, and Z. M. Guo, “Visual-weighted motion compensation frame interpolation with motion vector refinement,” in *Circuits and Systems (ISCAS), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 500–503.

[15] H. B. Liu, R. Xiong, D. B. Zhao, S. Ma, and W. Gao, “Multiple hypotheses bayesian frame rate up-conversion by adaptive fusion of motion-compensated interpolations,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 8, pp. 1188–1198, 2012.

[16] G. D. Field, J. L. Gauthier, A. Sher, M. Greschner, T. A. Machado, L. H. Jepson, J. Shlens, D. E Gunning, K. Mathieson, W. Dabrowski, et al., “Functional connectivity in the retina at the resolution of photoreceptors,” *Nature*, vol. 467, no. 7316, pp. 673–677, 2010.

[17] Z. Wang and A.C. Bovik, and H.R. Sheikh, and E.P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, 2004.