Frame Rate Up-Conversion Using Motion Vector Angular for Occlusion Detection

Yue Zhao[1,2], Ju Liu[1(✉)], Guoxia Sun[1], Jing Ge[1], and Wenbo Wan[1]

[1] School of Information Science and Engineering, Shandong University, Jinan 250100, China
{zhaoyue201211771,norling1,gjgjj1989}@163.com, {juliu,sun_guoxia}@sdu.edu.cn

[2] Suzhou Research Institute of Shandong University, Suzhou 215021, China

Abstract. In this paper, we study on handling the issue of occlusions in frame rate up-conversion (FRUC), which has been widely used to reconstruct high-quality videos presented on liquid crystal display. Depending on different types of occlusion problems, adaptive motion estimation is carried out for reducing computational complexity. Luminance information based RGB angular distance color difference formula is proposed for improving Unsymmetrical-cross multi-hexagon grid search (UMHexagonS) motion estimation, which reduces the occlusion regions resulted by the wrong motion vectors. Non-occlusion regions are determined by motion vector angular, furthermore, exposed and occluded areas are located by comparisons of interpolated frames within directional motion estimation. Consequently, adaptive motion compensation is introduced to calculate interpolated frames. Experimental results demonstrate that our scheme has a superior performance compared with preciously proposed FRUC schemes. *abstract* environment.

Keywords: Frame rate up conversion · Adaptive motion estimation · Occlusion detection · Motion vector angular

1 Introduction

With the rapid development of computer and communication networks, FRUC has been widely studied as an essential means to increase temporal resolution in multimedia information processing. The basic principle of FRUC is liner interpolation according to neighboring sequence frames. FRUC not only can reduce flashings on screen which makes images clearer and steadier, but also save expensive costs of filming, storing and transferring high frame rate video [1]. Motion compensation frame interpolation (MCFI) has been an effective FRUC method, which can get the interpolated frame taking motion information into accounted.

However, MCFI faces many inevitable problems, such as block artifacts, hole problems, occlusion problems, etc. Naturally, these problems have affected the video quality and users' visual experience seriously. Hole problems can be solved by bi-directional motion estimation (ME) efficiently [2], and overlapped block

© Springer International Publishing Switzerland 2015
Y.-J. Zhang (Ed.): ICIG 2015, Part II, LNCS 9218, pp. 301–311, 2015.
DOI: 10.1007/978-3-319-21963-9_28

motion compensation can removal the edge artifacts around blocks [3]. Much research has been conducted over the years to address the issue of the occlusion problems.

Actually, occlusion is due to relative motion between motion objects or foreground and background, where occluded areas are covered while exposed areas are uncovered. Therefore, ignoring occlusion issue can result in annoying halo artifacts on object's edge [4]. A FRUC method combined true motion estimation and directionally adaptive motion compensated interpolation is proposed [5]. In [6], M. B. put forward a motion vector steered video in-painting method. An improved FRUC method based on dual ME and regional blending can also solve occlusion problems effectively [7]. In summary, detection and refinement of occlusion regions are two key steps for handling occlusion. In this paper, motion vector angle is firstly presented for locating non-occlusion regions. Secondly, adaptive motion compensation is given to modify occluded and exposed areas, which is merged with comparison of interpolated frames obtained by directional ME.

The contributions of this paper can be summarized as: (1) Adaptive motion estimation is used to handle different occlusion problems, which can fill up exposed areas caused by scene change in the simplest way. (2) Luminance information based UMHexagonS motion estimation [8] brings RGB angular distance color difference formula to decrease occlusion regions. (3) Comparison of interpolated frame from directional ME is to distinguish the occluded or exposed areas, and adaptive motion compensation makes modifications on occlusion regions. Subjective and objective evaluation results prove the proposed method can improve the quality of interpolated frame significantly with a low-complexity.

The rest of this paper is organized as follows. In Sect. 2, the proposed method is described in details. Experimental results and performance comparisons are provided in Sect. 3. Finally, we conclude the paper in Sect. 4.

2 Proposed Scheme

The proposed FRUC method is schematically shown in Fig. 1. The detail procedure are as follows: 1. Scene change detection based adaptive motion estimation [9]. 2. Detection of non-occlusion regions based on motion vector angular. 3. Location and modification on occlusion regions. In the first stage, adaptive ME can deal with the exposed areas introduced by scene change simply and effectively. In the second stage, luminance information based UMHexagonS ME guarantees for lowering occlusion regions firstly, then the use of motion vector angular results in detecting occlusion regions effectually. In the third stage, exposed and occluded areas are located and refined in terms of directional motion estimation.

Adaptive Motion Estimation Handling Different Occlusion Problems. There are three factors causing occlusion problems. It can be concluded from Fig. 2 on different sequences. The first one is exposed regions produced by scene change, as shown in *New*. The other one is static exposed background caused by

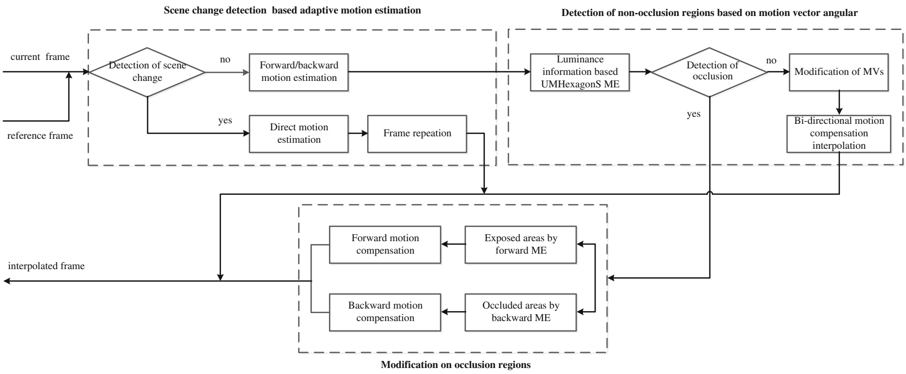

Fig. 1. Flowchart of the processed method

the motion of foreground, as illustrated in *Silent*. The last one, we can observe from *Ice* that occluded and exposed regions are induced by object motion. It is inescapably clear that the accurate motion vectors between sequence frames are absent when scene changes, so MCI method is not available. Taking computational complexity into account, frame repetition method is selected to obtain interpolated frames. Adaptive motion estimation is given by

$$f_{n+\frac{1}{2}}(x,y) = \begin{cases} f_{n+1}(x,y) \\ \frac{1}{2}f_n(x,y) + \frac{1}{2}f_{n+1}(x+v_x, y+v_y) \end{cases} \tag{1}$$

where $f_{\otimes}(\otimes)$ represents the frame, $f_{n+\frac{1}{2}}$, f_{n-1} and f_{n+1} are interpolated, previous and following frame relatively.

Generally, there is an relatively simple and efficient method based on average luminance difference for scene change detection [10], where the probability of scene change is small during FRUC. The formula is given by

Fig. 2. Different types of occlusion problems. From *left* to *right* and *up* to *down* video sequences (90th and 91th frames of *News*, 20th and 21th frames of *Silent*, 20th and 21th frames of *Ice*)

$$D(n, n+1) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} |f_n(i,j) - f_{n+1}(i,j)|$$
$$R_{\Delta D} = \frac{D(n, n+1)}{D(n, n-1)} \tag{2}$$

where $D(n, n-1)$ denotes the average luminance difference between f_n and f_{n-1}, and $D(n-1, n-2)$ is the average luminance difference between f_{n-1} and f_{n-2}. $f_n(i,j)$ and $f_{n-1}(i,j)$ are luminance values between the two neighboring sequence images. $H \times W$ represents frame revolution, and $R_{\Delta D}$ is difference ratio of adjacent frames. If $R_{\Delta D}$ is larger than established threshold which is empirically set as 5, scene is changing, otherwise, there is no changing.

Luminance Information Based UMHexagonS Motion Estimation. In general, sum of absolute difference (SAD) is the most popular matching criteria for its simplicity and effectiveness. SAD accounts for the difference of three color components in RGB color space between current block and candidate block. It can be described as follows

$$SAD_B(x, y) = \sum_{(x,y) \in B} |f_n(x, y) - f_{n+1}(x, y)| \tag{3}$$

in essence, the calculation process is to acquire absolute difference of three channels, then tot up, and given in

$$SAD_B = \sum (|r_n - r_{n+1}| + |g_n - g_{n+1}| + |b_n - b_{n+1}|) \tag{4}$$

where (r_n, g_n, b_n) and $(r_{n+1}, g_{n+1}, b_{n+1})$ are the color value of current and reference pixel respectively. The above-mentioned formula ignores some important details and facts, as shows in the following. Firstly, it has different sensitive degree to eyes, when red, green, blue is changing; Secondly, the significance of three color occupies different positions in different images. At last, the space distance and vector angular is also the important factors. In this work, a color difference formula combining the characters of distance and angular [11] is applied to calculate SAD. There are two pixels in current frame and reference frame, where their values are (r_n, g_n, b_n), $(r_{n+1}, g_{n+1}, b_{n+1})$ respectively. SAD is calculated as following

$$SAD_B = \sqrt{\frac{S_r^2 w_r \Delta_r^2 + S_g^2 w_g \Delta_g^2 + S_b^2 w_b \Delta_b^2}{(w_r + w_g + w_b) 255^2} + S_\theta S_{ratio} \theta^2} \tag{5}$$

where $\Delta_r = r_n - r_{n+1}$, $\Delta_g = g_n - g_{n+1}$, $\Delta_b = b_n - b_{n+1}$. (w_r, w_g, w_b) is $(1, 2, 1)$, and (S_r, S_g, S_b) is the significance of the three color components. θ stands for the motion vector angular normalized coefficient of the pixels in RGB color space regulated by S_θ, where S_θ is the gradient varies of three color components. S_{ratio} is an accommodation coefficient to avoid excessive value of θ. The above-mentioned formula has improved the accuracy of SAD, and, if luminance information is taken into accounted in UMHexagonS motion estimation, the preciseness will be better. Therefore, luminance information based UMHexagonS motion estimation using RGB angular distance color difference formula goes as follows:

Step 1. Transform from RGB to HSV, then abstract luminance information.

Step 2. Calculate the luminance difference ρ between the current block and candidate block, which is composed by original search point, then threshold $\phi = 1.5\rho$.

Step 3. During motion estimation, if luminance difference $\rho >= 1.5\phi$, improved RGB angular distance color difference formula is applied for figuring up SAD value. If not, carry on the next candidate block, in the same instant, update threshold.

An experiment has been conducted to test the validity of the proposed SAD algorithm under the environment of Matlab. YUV standard video sequence named soccer is selected for experiment, contrast tests are identical in algorithms and corrected experimental parameters besides calculation method of SAD. As can be seen from the comparison results in Fig. 3, proposed SAD algorithm is preferred to classical algorithm in both subjective and objective effects.

Fig. 3. Comparison of quality evaluation applying different SAD algorithm: From *left* to *right* SAD algorithm (conventional PSNR = 27.2163, proposed PSNR = 28.5002)

2.1 Detection of Non-occlusion Regions Based on Motion Vector Angular

For occluded and exposed areas, the correct pixel information is only available in previous or following frames. Occluded areas can achieve correct motion information by backward motion estimation and vice versa for exposed areas. Consequently, as for non-occlusion regions, motion vector angular from forward

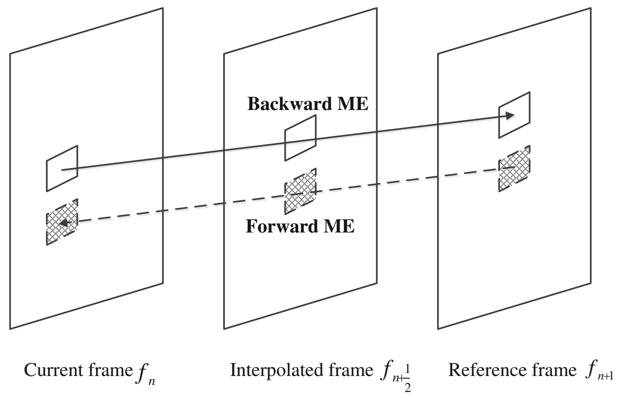

Fig. 4. A diagram of motion vector angular by directional motion estimation

and backward motion estimation is relative, where there reveals in Fig. 4. The specific process is concluded as follows. Firstly, forward and backward motion estimation provide motion vector field $F_{mv}(\boldsymbol{x}, \boldsymbol{y})$ and $B_{mv}(\boldsymbol{x}, \boldsymbol{y})$ separately. Secondly, motion vector angular is divided into eight fields according to rectangular coordinate system, which are X-axis, Y-axis and four quadrants. $F_{mv}(\boldsymbol{x}, \boldsymbol{y})$ and $B_{mv}(\boldsymbol{x}, \boldsymbol{y})$ are transformed to the motion vector field of the interpolated frame along respective motion trajectory. So two label matrices obtained to mark the directions of motion vector angular, which are denoted by F_{label} and B_{label}. Thirdly, If F_{label} and B_{label} belongs to the same relative direction, such as first quadrant and third quadrant, etc. The corresponding regions are regarded as non-occlusion areas. Otherwise, it needs further modifications.

2.2 Locations and Modifications on Occlusion Regions

The essence of occlusion is given in Fig. 5, occluded and exposed areas are judged as below. For occluded areas, the responding block in the interpolated frame acquired by backward ME should be similar to the block in previous frame f_n. Likewise, for exposed areas, the responding block obtained by forward ME in the interpolated frame should be similar to the block in the following frame f_{n+1}. The calculation formula are displayed in Eq. (6).

$$\begin{aligned}
SAD_{F.exp}(x, y) &= \sum (f_{F.(n+\frac{1}{2})}(x, y) - f_n(x + mv_{x.F}, y + mv_{y.F})) \\
SAD_{B.exp}(x, y) &= \sum (f_{B.(n+\frac{1}{2})}(x, y) - f_n(x - mv_{x.B}, y - mv_{y.B})) \\
SAD_{F.occ}(x, y) &= \sum (f_{F.(n+\frac{1}{2})}(x, y) - f_{n+1}(x - mv_{x.F}, y - mv_{y.F})) \\
SAD_{B.occ}(x, y) &= \sum (f_{B.(n+\frac{1}{2})}(x, y) - f_{n+1}(x + mv_{x.B}, y + mv_{y.B}))
\end{aligned} \tag{6}$$

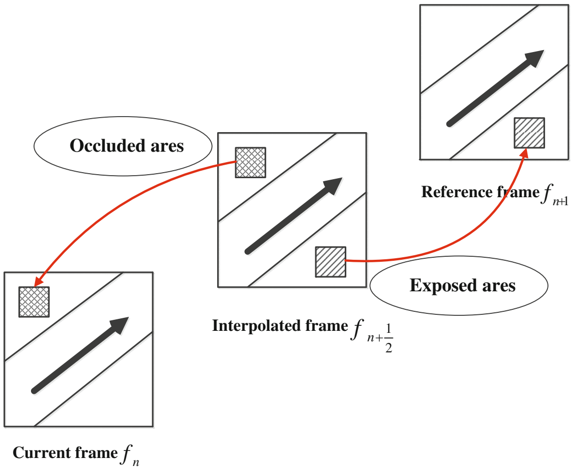

Fig. 5. A diagram of exposed and occluded areas.

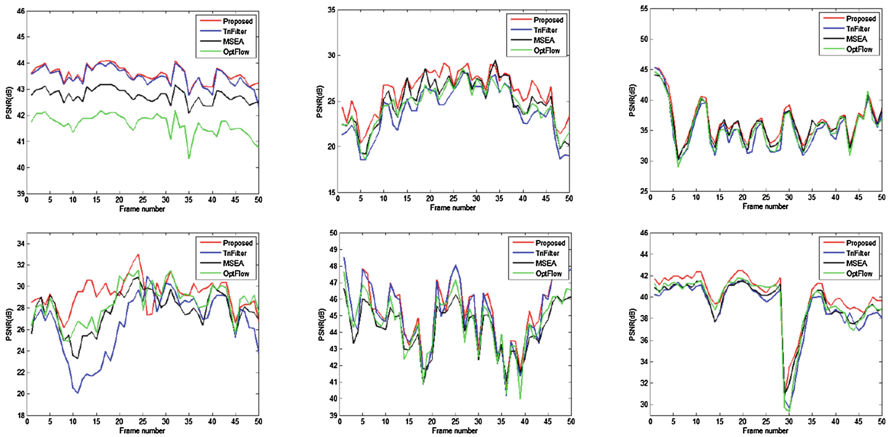

Fig. 6. PSNRs Comparison with original 50 frames on different sequences. From *left* to *right* and *up* to *down* video sequences (*Container, Ice, Silent, Soccer, Akiyo, Mother*)

where abbreviation 'F' and 'B' represents *forward ME*, *backward ME* respectively. Meanwhile, 'ex' and 'oc' stands for *exposed* and *occluded* areas. If $SAD_{F.oc} > SAD_{B.oc}$, it is located as occluded areas. While $SAD_{F.ex} < SAD_{B.ex}$, it belongs to exposed areas. Eventually, it can be drawn that forward MVs are suitable for exposed areas and backward MVs are appropriated for occluded areas. Hence, the interpolated frame is obtained as formula in Eq. (7)

$$f_{n+\frac{1}{2}}(x,y) = \frac{1}{2} \times (f_n(x - v_x, y - v_y) + f_{n+1}(x + v_x, y + v_y)). \tag{7}$$

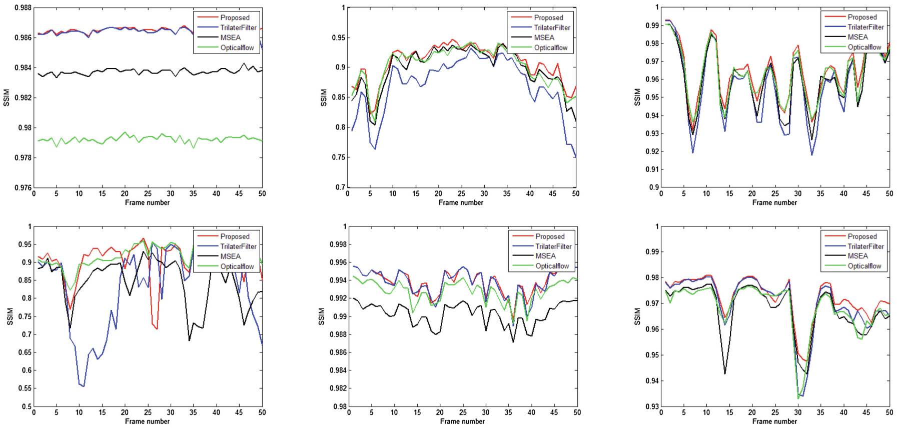

Fig. 7. SSIMs Comparison with original 50 frames on different sequences. From *left* to *right* and *up* to *down* video sequences (*Container, Ice, Silent, Soccer, Akiyo, Mother*)

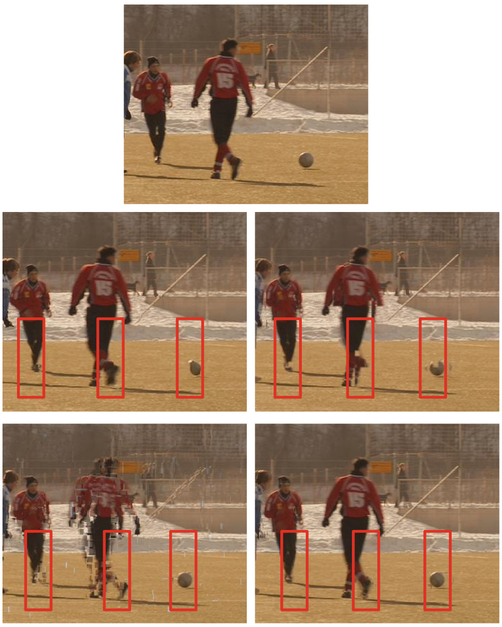

Fig. 8. Visual comparison on frame 26th of Soccer sequence. The first line is original frame and the second line is interpolated frames. From *left* to *right* FRUC algorithms (*OptFlow, MSEA, TrilFilter, Proposed*)

3 Experimental Results

In the experiments, six test video sequences with standard CIF (352×288) are adopted, where these have different spatial details and movement complexity.

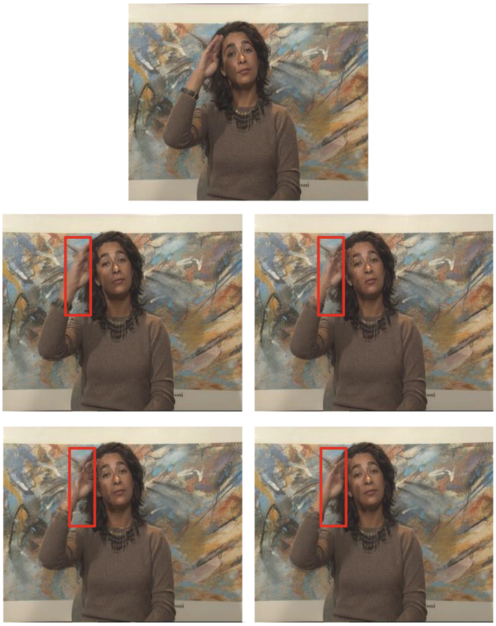

Fig. 9. Visual comparison on frame 26th of silent sequence. The first line is original frame and the second line is interpolated frames. From *left* to *right* FRUC algorithms (*OptFlow, MSEA, TrilFilter, Proposed*)

Table 1. Average PSNR comparison with original frames on 6 sequences

Test sequences	MSEA	OptFlow	TriFilter	Proposed
Container	42.7582	41.6684	43.4705	**43.5756**
Ice	24.9453	24.5195	23.8933	**25.9980**
Silent	36.0212	35.4738	35.3099	**36.4373**
Soccer	27.7633	28.4840	26.5814	**29.3107**
Akiyo	44.2779	44.5363	45.1194	**45.2931**
Mother	39.4248	39.3633	38.9698	**40.1008**

In order to evaluate the performance of proposed FRUC method, we make a comparison with three conventional FRUC methods, i.e., TriFilter [12], MSEA [13], and OptFlow [14]. Objective evaluation of the proposed method compared with above methods is on the principle of Peak signal-to-noise ratio (PSNR) and structural similarity index measurement (SSIM) [15]. Figures 6 and 7 give the results of the PSNRs and SSIMs of individual interpolated frame on different videos. Meanwhile the average PSNR and SSIM of different algorithms can be seen from Tables 1 and 2. On objective comparisons, the proposed algorithm has certain advantages over the conventional algorithms. Subjective quality can be evaluated by means of visual quality of local details, which is demonstrated in Figs. 8 and 9.

Table 2. Average SSIM comparison with original frames on 6 sequences

Test sequences	MSEA	OptFlow	TriFilter	Proposed
Container	0.9837	0.9792	0.9864	**0.9865**
Ice	0.8975	0.9020	0.8699	**0.9083**
Silent	0.9601	0.9630	0.9579	**0.9650**
Soccer	0.8468	**0.9126**	0.8286	0.9029
Akiyo	0.9903	0.9928	0.9938	**0.9939**
Mother	0.9682	0.9675	0.9701	**0.9722**

4 Conclusion

In this paper, a robust and fast FRUC method has been to address the occulsion problem. Adaptive motion estimation is first used for solving different types of occlusion problems, which reduces high complexity for filling exposed areas caused by scene change. Additionally, luminance information and RGB angular distance color difference formula union use into UMHexagonS ME is helpful to decrease the occlusion regions. What's more, motion vector angular based on directional ME is designed to detect non-occlusion regions, and the exposed and occluded areas is located according to the comparison of interpolated frames obtained by directional ME. In the end, we make use of adaptive MC to get the high quality interpolated frame. It can be supported that our newly proposed FRUC method improves the quality of interpolated frames both in objective and subjective evaluation by the results of experiments conducted.

Acknowledgments. This work was supported partially by Open Foundation of State Key Laboratory of Digital Multimedia Technology Laboratory (11131326) and Suzhou Science and technology plan (SYG201443).

References

1. Huang, C.L., Chao, T.T.: Motion-compensated interpolation for scan rate up-conversion. Opt. Eng. **35**(1), 166–176 (1996)
2. Li, R., Gan, Z., Cui, Z., Tang, G., Zhu, X.: Multi-channel mixed-pattern based frame rate up-conversion using spatio-temporal motion vector refinement and dual-weighted overlapped block motion compensation. J. Disp. Technol. **10**(12), 1010–1023 (2014)
3. Wang, D., Vincent, A., Blanchfield, P., Klepko, R.: Motion compensated framerate up-conversion-part ii: New algorithms for frame interpolation. IEEE Trans. Broadcast. **56**(2), 142–149 (2010)
4. Ozkalayci, B., Alatan, A.A.: Occlusion handling frame rate up-conversion. In: Proceedings of the ISPA, pp. 149–152 (2011)
5. Biswas, M., Namboodiri, V.: On handling of occlusion for frame rate up-conversion using video in-painting. In: Proceedings of the ICIP, pp. 785–788 (2010)

6. Guo, Y., Gao, Z., Chen, L., Zhang, X.: Occlusion handling frame rate up-conversion. In: Proceedings of the ICASSP, pp. 1424–1428 (2013)
7. Bellers, E. B., et al.: Solving occlusion in frame-rate up-conversion. In: Proceedings of the ICCE, pp. 1–2 (2007)
8. Zou, B., Shi, C., Xu, C.H., Chen, S.: Enhanced hexagonal-based search using direction-oriented inner search for motion estimation. IEEE Trans. Circuits Syst. Video Technol. **20**, 156–160 (2010)
9. Zhao, Y.: Frame rate up conversion method based on motion compensation. Master thesis, Shandong University (2015)
10. Ren, K., Jin, X.: Research on scene change detection algorithm based on H.264/SVC. In: Proceedings of the ICECC, pp. 1639–1642 (2011)
11. Yang, Z., Wang, Y., Yang, Z.: Vector-angular distance color difference formula in RGB color space. Broadcast. Comput. Eng. Appl. **46**, 154–156 (2010)
12. He, Y.H., Wang, C., Zhang, L., Tan, Y.P.: Frame rate up-conversion using trilateral filtering. IEEE Trans. Circuits Syst. Video Technol. **20**(6), 886–893 (2010)
13. Lee, Y. L., Nguyen, T.: Fast one-pass motion compensated frame interpolation in high-definition video processing. In: Proceedings of the ICIP, pp. 369–372 (2009)
14. Liu, C.: Beyond pixels: exploring new representations and applications for motion analysis. Doctoral thesis, Massachusetts Institute of Technology, pp. 369–372 (2009)
15. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process **13**(4), 600–612 (2004)