FRAME RATE UP-CONVERSION USING BI-DIRECTIONAL OPTICAL FLOWS WITH DUAL REGULARIZATION

Rahul Vanam and Yuriy A. Reznik
{rvanam, yreznik}@brightcove.com
Brightcove Inc., USA

ABSTRACT

In this paper, we present a frame rate up-conversion (FRUC) scheme that uses optical flows in forward and backward directions and with two different regularization parameters to derive alternative motion vector fields. The motion vectors are then assessed for their relevance, resulting in prediction weights assigned to each candidate motion vector. Multiple hypothesis reconstruction is then performed, where overlapping patches of pixels associated with multiple motion vectors are fused using their corresponding weights to yield pixels of an interpolated frame. Four flavors of our approach are proposed having different tradeoffs between quality and computational complexity. Experiments conducted using standard test set show that all our schemes yield significantly higher quality compared to existing FRUC algorithms.

***Index Terms*—** Frame rate up-conversion, optical flow, regularization, multiple hypothesis, motion compensation.

1. INTRODUCTION

Display technologies have been steadily improving over the years, resulting in devices such as TVs, tablets and mobiles with much higher spatial resolutions, HDR capability, and higher refresh rates. Many commercially available TVs and tablets already support 120 Hz refresh rate. High-action videos (e.g. sports, action movies, etc.) are typically played back at higher frame rates since it reduces motion blur and judder while maintaining smoothness during fast motion [1]. The importance of rendering at higher frame rates also increases with the increase of display form-factors and rendering of brighter/HDR content.

A video may have lower frame rate either because of the limits of the camera / production equipment, or due to temporal subsampling intentionally applied to make it more suitable for compression and transmission. When a low frame rate video is received on a device, its frame rate can be increased by interpolating intermediate frames using a frame rate up-conversion (FRUC) scheme.

Among the different categories of FRUC schemes, motion-compensated FRUC are most common. These methods typically have two main stages: (a) motion estimation for determining the motion vectors, and (b) interpolation of pixels using motion compensation.

Optical flow algorithms are known to yield motion vectors with higher accuracy [2], and are employed by a number of FRUC schemes [3, 4, 5, 6, 7]. Lucas-Kanade method (LK) [8] is a classic optical flow algorithm that has been used in FRUC designs [5]. Local All-Pass (LAP) algorithm [9] is another recently proposed method that was shown to be superior to LK-based FRUC [5]. Among recently developed optical flow methods, the algorithm of C. Liu [10], [11] was also shown to be promising for FRUC designs [3]. This method uses the Iterative Reweighted Least Squares for solving the optical flow problem instead of Euler-Lagrange approach. This leads to a simpler and more practical implementation [11].

The use of convolutional neural networks (CNNs) was also studied for the design of FRUC schemes. In [12], CNNs were designed to learn four 1D filters per pixel. However, such a scheme was shown to have enormous computational complexity [5] making it impractical to use.

Most optical flow algorithms include a regularization parameter that controls the smoothness of the generated motion field [13]. However, since motion within a video may vary, a single regularization parameter may not yield accurate motion vectors across the sequence. There could be scenes where a lower or higher regularization may improve accuracy of prediction. Many existing FRUC algorithms also apply optical flow only in one direction for deriving the motion field. However, such schemes may not yield accurate prediction, especially when there is irregular motion and there are occlusions within the sequence.

Therefore, unlike prior approaches we do not use a single regularization parameter and a single optical flow direction. Our FRUC algorithm incorporates the following:

1. To capture varying motion characteristics and varying content in a video, we use *bi-directional optical flows with different regularization parameters*, to yield diverse motion vector fields with different smoothness characteristics.
2. We use *multiple-hypothesis reconstruction* process for fusing the results of various predictions based on motion vectors produced by optical flows.
3. To drive such fusion, *relevance scores* are computed for all candidate motion vectors and are translated to *prediction weights*.

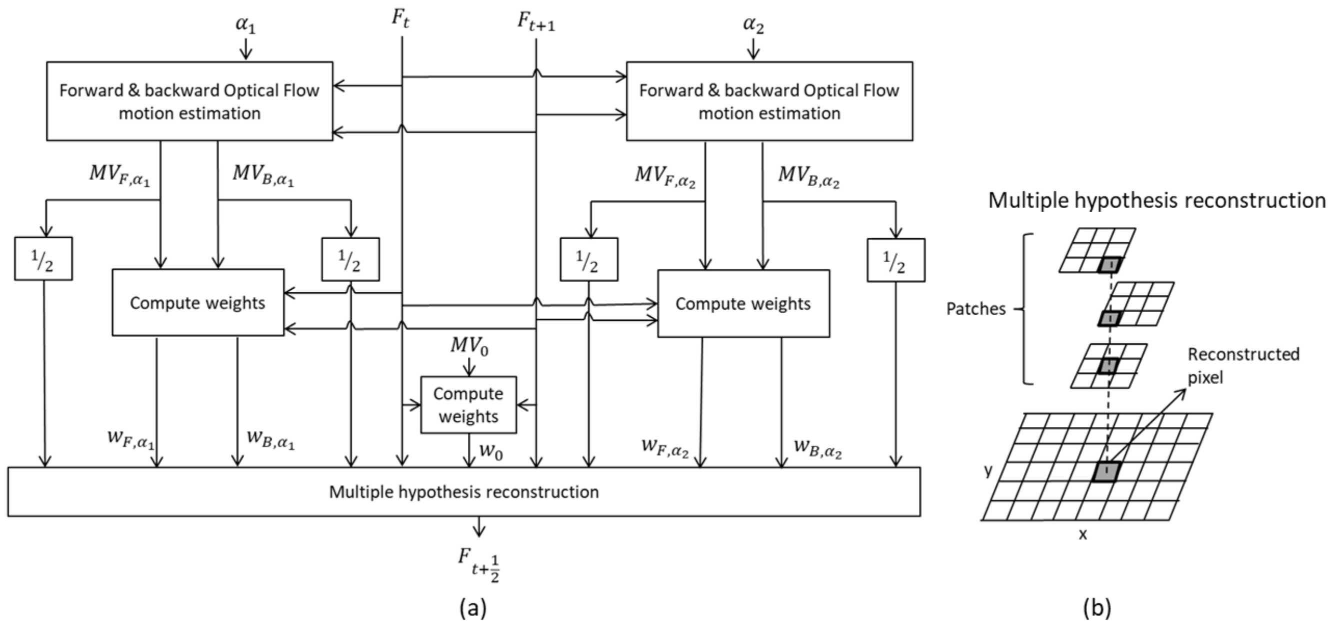

Figure 1. (a) Architecture of the proposed FRUC method. (b) Illustration of multiple hypothesis reconstruction.

We shall show that combination of these techniques allow us to achieve improved performance over existing FRUC methods based on LK [5], LAP [5], and CNN [12].

The paper is organized as follows. The proposed FRUC algorithm is described in Section 2. Experimental results and performance comparison with existing FRUC algorithms are presented in Section 3. Conclusions are provided in Section 4.

2. PROPOSED FRUC METHOD

The architecture of our FRUC algorithm is shown in Figure 1(a). Different FRUC schemes can be derived from this architecture as discussed later in Section 2.4. The input to our algorithm includes two consecutive frames, denoted as F_t and F_{t+1}, where t is a frame instance counter along time domain. The output is interpolated frame $F_{t+1/2}$ temporally positioned in the middle between the input frames. The values α_1 and α_2 denote the regularization parameters that we use with optical flow algorithms.

2.1. Generation of Motion Vector Fields

We use an optical flow algorithm of C. Liu [10, 11] for generating motion vector field. This algorithm uses a regularization parameter α for controlling the smoothness of the motion vector field. A larger value of α yields motion vectors that are smoother both spatially and temporally. As shown in Figure 1(a), we apply two pairs of optical flows with different regularization parameters α_1 and α_2, and they yield motion vector fields having different smoothness characteristics.

For a given input frame pair (F_t, F_{t+1}) and a given α, a pair of optical flows can be applied in the forward and backward directions to yield two motion vector fields $MV_{F,\alpha}$ and $MV_{B,\alpha}$ corresponding to the forward and backward predictions, respectively. Therefore, for α_1 and α_2, the optical flows generates four motion vector fields (MVFs): MV_{F,α_1}, MV_{B,α_1}, MV_{F,α_2}, and MV_{B,α_2}. Additionally, we also consider motion vector field MV_0 corresponding to zero motion vector.

2.2. Derivation of Prediction weights

The MVFs may yield motion vectors that are less reliable for predicting a given pixel. Therefore, predictions from more reliable motion vectors are to be weighed higher than those from less reliable motion vectors. The process of determining the reliability of motion vectors is described next, followed by the weight derivation procedure.

Let $MV_{F,\alpha_i}(x_1,y_1) = (u_{F,\alpha_i}(x_1,y_1), v_{F,\alpha_i}(x_1,y_1))$ represent the forward motion vector associated with regularization parameter α_i at pixel position (x_1,y_1) in frame F_t, and let (x_2,y_2) represent the predicted pixel position in frame F_{t+1}. Since the original pixels are available for F_{t+1}, the sum of the absolute difference (SAD) between a window of predicted pixels and the original pixels at F_{t+1} is computed as

$$SAD_{F,\alpha_i}(x_1,y_1) = \sum_{i=-\frac{M}{2}}^{\frac{M}{2}} \sum_{j=-\frac{M}{2}}^{\frac{M}{2}} |F_t(x_1-i,y_1-j) - F_{t+1}(x_2-i,y_2-j)|, \quad (1)$$

where $x_2 = x_1 + u_{F,\alpha_i}(x_1, y_1)$, and $y_2 = y_1 + v_{F,\alpha_i}(x_1, y_1)$, and M is the dimension of the SAD window. We use $SAD_{F,\alpha_i}(x_1, y_1)$ to measure the reliability of the motion vector $MV_{F,\alpha_i}(x_1, y_1)$, since smaller SAD values indicates higher reliability. The SAD is mapped to a weight value $w_{F,\alpha_i}(x_1, y_1)$ as follows

$$w_{F,\alpha_i}(x_1, y_1) = f\left(\frac{1}{SAD_{F,\alpha_i}(x_1, y_1)}\right), \tag{2}$$

where $f(.)$ is a smooth-step function defined as

$$f(x) = \begin{cases} 0, x < 0 \\ 3x^2 - 2x^3, 0 < x < 1 \\ 1, x > 1. \end{cases} \tag{3}$$

The above procedure is applied to $MV_{F,\alpha_1}, MV_{B,\alpha_1}, MV_{F,\alpha_2}, MV_{B,\alpha_2}$, and MV_0 to yield their associated set of weights $w_{F,\alpha_1}, w_{B,\alpha_1}, w_{F,\alpha_2}, w_{B,\alpha_2}$, and w_0.

2.3. Multiple hypothesis reconstruction

Multiple hypothesis reconstruction involves not only projecting a single reference pixel associated with a motion vector during reconstruction, but also its neighboring pixels, i.e., a patch. Pixels belonging to different patches can overlap at a given position (x, y) as illustrated in Figure 1(b), and these overlapping pixels are then used for interpolating the reconstructed pixel.

Let the patch size be $p \times p$. If (x', y') is an arbitrary position in the reconstructed frame $F_{t+\frac{1}{2}}$, all the overlapping patches at this position belong to the following set:

$$\begin{aligned} &C(x', y') \\ &= \{(x, y) | (x - m, y - n) + \frac{1}{2} MV'(x, y) = (x', y'), \\ &\quad MV' \in \{MV_{F,\alpha_1}, MV_{F,\alpha_2}, MV_{B,\alpha_1}, MV_{B,\alpha_2}, MV_0\}, \\ &\quad 0 \le x \le W, 0 \le y \le H, \\ &\quad \exists\ (m, n) \text{ where } -\tfrac{p}{2} \le m, n \le \tfrac{p}{2}\}, \end{aligned} \tag{4}$$

where W and H are the width and height of the frame, respectively.

The reconstructed pixel $F_{t+\frac{1}{2}}(x', y')$ is computed using Equation (5) by fusing multiple predictions associated with forward and backward motion vectors for α_1 and α_2, and zero motion vector. In Equation (5), $w_{F,\alpha_i}(x, y)$ and $w_{B,\alpha_i}(x, y)$ are the forward and backward prediction weights associated with α_i, respectively, and $w_0(x, y)$ is the weight associated with zero motion vector prediction.

2.4. Variants of the proposed FRUC method

Following four FRUC variants are derived from Figure 1(a):

i. *Forward flow (FF)*: uses single optical flow for deriving forward motion field, and performs forward prediction for frame interpolation.
ii. *Forward and inverse flow (FIF)*: derives forward motion field as in FF, and derives the backward motion field by flipping the forward motion field. Frame is interpolated using forward and backward motion fields.
iii. *Bi-directional flow (BF)*: derives forward and backward motion fields using two separate optical flows.
iv. *Dual regularization with bi-directional flow (DRBF)*: this is shown in Figure 1(a), where two regularization parameters are used, and for each regularization parameter a bi-directional flow is derived as in BF.

3. EXPERIMENTAL RESULTS

In our experiments, both the patch size p and SAD window size M are set to 7. For FF, FIF, and BF variants, the regularization parameter of the optical flow is set to 0.02, 0.02, and 0.025, respectively. For DRBF, the two regularization parameters are set to 0.02 and 0.036. Remaining optical flow parameters are set to the default values, as defined in [10].

As in [5], the following standard test sequences from [14] are used: “city”, “crew”, “harbor”, “ice”, “soccer”, and “stockholm”. All sequences are 4CIF resolution except for “stockholm”, which is 640×360. All sequences are temporally subsampled by two by dropping alternate frames.

Our schemes are compared with the FRUC approaches based on Lucas-Kanade method (LK) [5], CNN [12], and LAP algorithm [5]. The average mean squared error (MSE) of the luma component per sequence for the above three prior-works were obtained from [5], and compared with our DRBF scheme in Table 1. The results show that DRBF yields significantly lower MSE for all test sequences. On an average, DRBF yields MSE that is lower by a factor of 9, 4.7, and 9.6 compared to LK, CNN, and LAP, respectively. Furthermore, all of our four schemes yield significantly lower average MSE across all test sequences compared to prior-works as shown in Tables 1 and 2.

We measure the average execution time of our four schemes for interpolating a frame of resolution 960×540 using a machine with Intel i7-8850H processor and 16 GB RAM, using the methodology in [5]. Table 3 compares the execution times of our schemes with the three prior-works whose execution times were obtained from [5]. While our schemes have computation time larger than LAP and LK, they are significantly faster than CNN as shown in Table 3.

Among our four schemes, FF is found to be the fastest,

$$F_{t+\frac{1}{2}}(x', y') = \frac{\sum_{(x,y),(m,n,k,l) \in C(x',y')} \left(F_t(x-m, y-n)\ w_0(x,y) + \sum_{i=1}^{2} F_t(x-m, y-n) w_{F,\alpha_i}(x,y) + F_{t+1}(x-k, y-l) w_{B,\alpha_i}(x,y)\right)}{\sum_{(x,y),(m,n,k,l) \in C(x',y')} \left(w_0(x,y) + \sum_{i=1}^{2} w_{F,\alpha_i}(x,y) + w_{B,\alpha_i}(x,y)\right)} \tag{5}$$

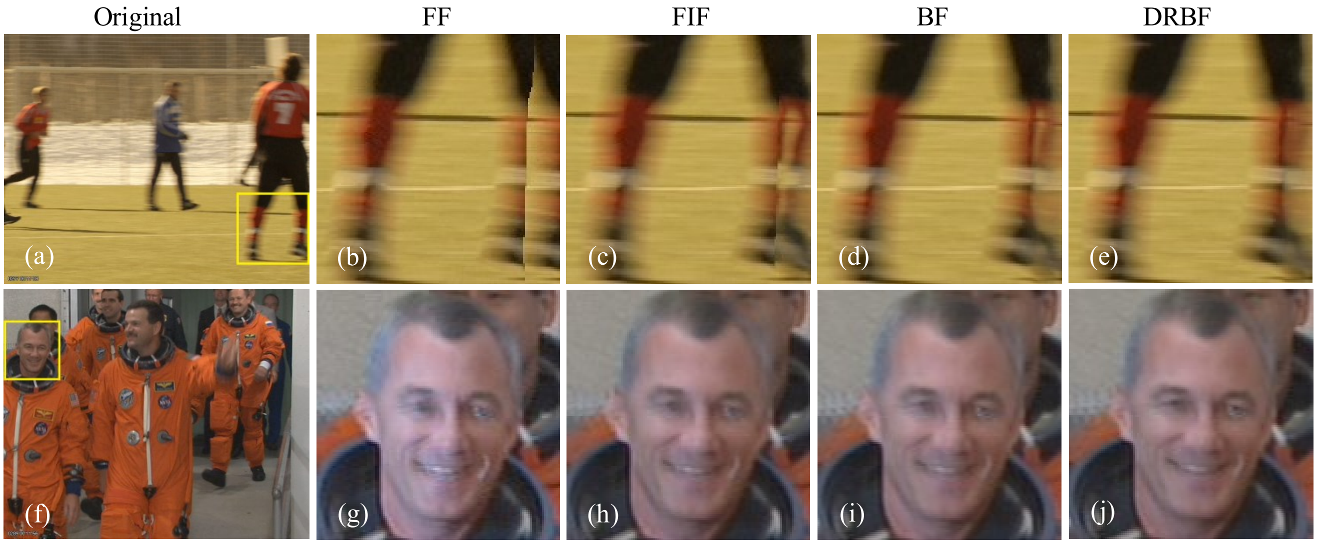

Figure 2. Visual quality comparison of the proposed FRUC schemes: FF, FIF, BF, and DRBF. (a) The 278th frame of the soccer sequence, and (b) – (e) are sections from the interpolated frames. (f) The 290th frame of the crew sequence, and (g) – (j) are sections from the interpolated frames.

Table 1. Average MSE of the three prior-works and the proposed DRBF approach.

Sequence	**LK** [5]	**CNN** [12]	**LAP** [5]	**DRBF**
city	111.7	79.8	62.1	**18.4**
crew	806.5	406.1	522.1	**77.8**
harbour	117.1	94.8	112.2	**29.1**
ice	109.9	57.4	129.6	**8.5**
soccer	399.5	141.6	848.3	**33.0**
stockholm	77.6	60.4	54.6	**12.7**
Avg. MSE	270.4	140.0	288.2	**29.9**

Table 2. Average MSE and PSNR across all six test sequences for the proposed four FRUC schemes.

Metric	**FF**	**FIF**	**BF**	**DRBF**
Avg. MSE	40.0	31.0	30.3	**29.9**
Avg. PSNR (dB)	34.11	35.60	35.69	**35.77**

Table 3. Average execution time in seconds required for interpolating a frame of resolution 960×540.

LK [5]	**CNN** [12]	**LAP** [5]	**FF**	**FIF**	**BF**	**DRBF**
0.2	76204	4.3	10	11.5	19.7	38.3

as shown in Table 3, since it uses single optical flow with forward prediction. Since FIF performs additional backward prediction using the forward flow field the computation time slightly increases by 15% with 1.49 dB PSNR improvement over FF as shown in Table 2. Since BF uses two optical flows for two prediction directions, the computation time is almost twice that of FF with 1.58 dB PSNR improvement. Finally, DRBF uses BF with two regularization parameters, thereby increasing its computation time to almost four times that of FF with 1.66 dB PSNR improvement.

Figure 2 compares the visual quality of interpolated frames of our four schemes using the 278th and 290th frames of soccer and crew sequences, respectively. In Figure 2(e), the right leg of the soccer player is shown to be better interpolated by DRBF, while other schemes produce severe artifacts as shown in Figure 2(b) – (d). In the crew sequence, camera light momentary flashes only for the 289th frame. Since FF performs forward prediction only, the camera flash light propagates to the interpolated 290th frame, as shown in Figure 2(g). Comparing Figure 2(h) – (j), interpolated face generated by DRBF is found to be less blurry compared to other schemes. These examples illustrate that although DRBF yields small PSNR gain over FIF and BF, it yields interpolated frames with higher visual quality.

4. CONCLUSIONS

In this paper, we have presented a novel FRUC algorithm called *Dual regularization with bi-directional flow* (DRBF) that utilizes bi-directional optical flows with two regularization parameters and multiple hypothesis reconstruction. The results show that our approach is highly promising, and it yields MSE that is lower by a factor of 9.6, 9, and 4.7 compared to LAP [5], LK [5], and CNN [12] based schemes, respectively. Although our approach has computational time larger than the LK and LAP based schemes, it is significantly faster than the CNN-based scheme. Finally, we compare four variants of our approach and show that DRBF yields significantly better visual quality for the interpolated frames. As future work, we plan to improve the speed of DRBF by designing efficient implementation of optical flows when using bi-directional and dual regularization process. We also plan to apply our FRUC method to other temporal sampling conversion problems such as conversion between interlaced and progressive videos, telecine pattern detection, etc.

5. REFERENCES

[1] "4K high frame rate (HFR)," LG, [Online]. Available: https://www.lg.com/us/experience-tvs/oled-tv/4k-hfr. [Accessed January 2020].

[2] T. Brox, A. Bruhn, N. Papenberg and J. Weickert, "High Accuracy Optical Flow Estimation Based on a Theory for Warping," in *European conference on computer vision*, Berlin, Heidelberg, 2004.

[3] H. R. Kaviani and S. Shirani, "Frame rate upconversion using optical flow and patch-based reconstruction," *IEEE Transactions on Circuits and Systems for Video Technology,* Vols. 26, No. 9, pp. 1581-1594, 2015.

[4] W. Bao, X. Zhang, L. Chen, L. Ding and Z. Gao, "High-order model and dynamic filtering for frame rate up-conversion," *IEEE Transactions on Image Processing,* Vols. 27, No. 8, pp. 3813-3826, 2018.

[5] T. Jayashankar, P. Moulin, T. Blu and C. Gilliam, "LAP-based video frame interpolation," in *IEEE International Conference on Image Processing (ICIP)*, 2019.

[6] W. H. Lee, K. Choi and J. B. Ra, "Frame rate up conversion based on variational image fusion," *IEEE Transactions on Image Processing,* Vols. 23, No. 1, pp. 399-412, 2013.

[7] R. Li, H. Liu, Z. Liu, Y. Li and Z. Fu, "Motion-compensated frame interpolation using patch-based sparseland model," *Signal Processing: Image Communication,* vol. 54, pp. 36-48, 2017.

[8] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the International Joint Conference on Artificial Intelligence*, 1981.

[9] C. Gilliam and T. Blu, "Local all-pass filters for optical flow estimation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[10] C. Liu, "Optical Flow Matlab/C++ Code," August 2011. [Online]. Available: https://people.csail.mit.edu/celiu/OpticalFlow/.

[11] C. Liu, "Beyond Pixels: Exploring New Representations and Applications for Motion Analysis," Ph.D. Thesis, Massachusetts Institute of Technology, 2009.

[12] S. Niklaus, L. Mai and F. Liu, "Video frame interpolation via adaptive separable convolution," in *IEEE International Conference on Computer Vision*, 2017.

[13] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial intelligence,* vol. 17, no. 1-3, pp. 185-203, 1981.

[14] "Xiph.org Video Test Media," [Online]. Available: https://media.xiph.org/video/derf/.