

# MOTION VECTOR REFINEMENT FOR FRUC USING SALIENCY AND SEGMENTATION INFORMATION

Natan Jacobson, Yen-Lin Lee, Vijay Mahadevan, Nuno Vasconcelos, Truong Q. Nguyen

ECE Dept, UCSD, La Jolla, CA 92093-0407  
<http://videoprocessing.ucsd.edu/>

## ABSTRACT

An algorithm for motion vector refinement is proposed which makes use of saliency as well as segmentation information to improve performance of Frame Rate Up Conversion. Multi-scale motion vector refinement is applied to highly salient scene regions, while motion consistency is enforced for non-salient regions. This produces a noticeably improved up-converted video sequence to human observers. Saliency information is obtained by a motion-based discriminant saliency algorithm using a dynamic texture model for its feature set. Over-segmentation is performed by normalized-cuts followed by a graph-based region merging algorithm.

**Index Terms**— Frame Rate Up Conversion, Discriminant Saliency, Motion Vector Refinement, Image Segmentation, Motion Compensated Frame Interpolation

## 1. INTRODUCTION

This work proposes a novel method for Frame Rate Up Conversion (FRUC) targeted at HD content. FRUC has become an area of significant research as LCD displays have become omnipresent in the market. Unlike CRTs or Plasmas which are impulse-driven, LCDs are driven in a sample-and-hold pattern, causing noticeable motion blur at low frame rates. Newer LCDs on the market are capable of displaying at 120 to 240Hz thus reducing the sample period and, consequently, the amount of motion blur. Because very little data exists natively at above 60Hz, FRUC is employed to increase the frame rate without causing motion judder as in frame repetition and downpull schemes.

FRUC is composed of two portions: Motion Estimation (ME) and Motion Compensated Frame Interpolation (MCFI). A block-based ME algorithm operates by partitioning each frame into uniform blocks (generally 8x8 pixels) and determining the relative translation between each block in successive video frames. The MCFI engine creates an intermediate frame by interpolating along the motion field direction. Given a motion vector  $(v_x, v_y)$ , a block in the interpolated frame  $f_t$  is calculated as follows from the current frame  $f_{t+1}$  and ref-

erence frame  $f_{t-1}$ :

$$f_t(x, y) = 0.5f_{t-1}\left(x + \frac{v_x}{2}, y + \frac{v_y}{2}\right) + 0.5f_{t+1}\left(x - \frac{v_x}{2}, y - \frac{v_y}{2}\right) \quad (1)$$

Because FRUC is performed on a block basis, there are several issues which we aim to resolve. One limitation of a block-based method is that objects in the scene generally do not conform to block boundaries. Therefore, a single block may contain conflicting motion. Another limitation is that the motion vector which minimizes predicted block error may in fact not be the best choice. This can occur because of changes in luminance between frames or due to repetitive structures. Finally, FRUC can suffer from ghosting artifacts which are caused by large motions being assigned outside of object boundaries. We address these issues in the proposed work.

We propose a novel method for FRUC aimed at improving both objective and subjective quality compared with previous methods. Saliency detection is employed in order to determine which regions of the scene are visually important to a human observer, thereby requiring additional attention. Motion Vector (MV) smoothness and consistency are enforced for non-salient regions using a fast segmentation.

The paper is organized as follows. A detailed overview of discriminant saliency is introduced in Section 2 and of segmentation in Section 3. Our proposed algorithm is detailed in Section 4. Section 5 presents both objective and subjective experimental results for our proposed method. Finally, we conclude in Section 6.

## 2. DISCRIMINANT SALIENCY

Human observers typically focus their visual attention on small regions of the video frame that appear interesting. By subjecting only these attended regions to post-processing such as Motion Vector Refinement (MVR), the quality of FRUC can be improved while keeping computational complexity manageable. The automatic selection of the regions of interest as perceived by the human visual system (HVS) has been well studied in the context of bottom-up saliency, and

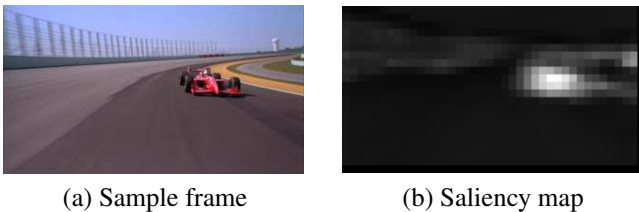
has been applied to improve video compression [1]. However, these techniques have been developed for static images and are not suitable for motion based region of interest identification. Therefore, in this work, we use the recently proposed discriminant center-surround model for motion saliency [2] to automatically identify salient moving objects.

Discriminant center-surround saliency is a biologically plausible algorithm that has been shown to replicate the psychophysics of saliency mechanisms in the HVS. It can directly be applied to motion saliency simply by using appropriate motion models such as optical flow or dynamic textures [3].

Discriminant saliency is defined with respect to two classes of stimuli and a feature  $\mathbf{Y}$ : the class of visual stimuli in the center (with label  $C = 1$ ), and class of visual stimuli in the *background* or surround (with label  $C = 0$ ). The saliency of location  $l$  of the video, denoted  $S(l)$ , is the extent to which the feature  $\mathbf{Y}$  can discriminate between *center* and *surround* at  $l$ . This is quantified by the mutual information between features,  $\mathbf{Y}$ , and class label,  $C$ ,

$$S(l) = I_l(\mathbf{Y}; C) = \sum_{c=0}^1 \int p_{Y,C(l)}(\mathbf{y}, c) \log \frac{p_{Y,C(l)}(\mathbf{y}, c)}{p_Y(\mathbf{y})p_{C(l)}(c)} d\mathbf{y}. \quad (2)$$

A large  $S(l)$  implies that center and surround have a large disparity of feature responses, i.e. large *local feature contrast* indicating that the location is salient. By selecting an appropriate feature  $\mathbf{Y}$  that encodes both spatial and temporal characteristics of the video (e.g. dynamic textures, optical flow) we can obtain regions that are spatiotemporally salient. A saliency map for the ‘‘Speedway’’ sequence obtained using dynamic textures is demonstrated in Figure 1. This result shows that the regions predicted to have high saliency (e.g the rapidly moving car) are indeed the regions that appear visually salient to a human observer.



**Fig. 1.** Saliency map produced for ‘‘Speedway’’ sequence where the saliency value,  $S(l)$ , has been normalized to the range  $[0, 255]$  prior to rendering.

### 3. SEGMENTATION

For the scope of this paper, we make use of the segmentation algorithm provided in [4], which is based on Normalized Cuts [5]. As is common in the literature, this segmentation scheme will be used to oversegment the image, with

each frame being segmented into a large number of distinct regions. For the N-Cuts algorithm, the  $n_{ev} = 100$  eigenvectors with smallest eigenvalue determine the segmentation of the frame. After Normalized Cuts, the image is further segmented into a coarse oversegmentation of  $n_{spc} = 200$  regions by using K-means on the initial segmentation. This step is repeated once more to produce a fine oversegmentation with  $n_{spf} = 400$  regions.

With the frame oversegmented, the next step is to merge regions with similar characteristics. Regions with similar color and texture are merged on the assumption that they belong to the same object. This process is repeated until a small number of regions exist. The merge operation terminates when no two nodes can be located with a sufficiently small dissimilarity.

In order to compute the texture measure, the variance of the AC coefficients of the Discrete Cosine Transform (DCT) of each  $8 \times 8$  block is computed. This is consistent with the definition of texture in the related literature.

The superpixel merge process is posed as a problem over the graph  $G = (V, E)$ . Here,  $\{v_1, \dots, v_n\} \in V$  is the set of all superpixel regions, and the edges  $\{e_{i,j}\} \in E$  for  $i, j \in [1, n]$  contain a dissimilarity measure between each pair of nodes. The edge  $E_{ij} = 0$  if nodes  $v_i, v_j \in V$  are non-adjacent. We use an indicator function  $b_{i,j}$  to represent adjacent nodes.  $b_{i,j} = 1$  if  $v_i, v_j \in V$  are adjacent, and zero otherwise.

$$E_{i,j} = b_{i,j} \left[ \lambda \max \left\{ \mathbf{I}_i^{RGB} - \mathbf{I}_j^{RGB} \right\} + (1 - \lambda) |T_i - T_j| \right] \quad (3)$$

$$\mathbf{I}_i^{RGB} = \frac{1}{|\{v_i\}|} \left[ \sum_{j \in v_i} R(j), \sum_{j \in v_i} G(j), \sum_{j \in v_i} B(j) \right]^T \quad (4)$$

where  $\mathbf{I}_i^{RGB}$  is the average intensity over the RGB color planes and  $T_i$  is the average texture measure for superpixel region  $v_i$ . The tuning parameter  $\lambda$  allows the user to emphasize either color or texture for the merging process. For all experiments conducted in this paper, the parameter is set to  $\lambda = 0.5$ . The merge procedure requires iteratively locating the pair of nodes  $v_i, v_j \in V$  such that  $E_{i,j}$  is minimized. These nodes are then merged, and the process continues.

### 4. PROPOSED ALGORITHM

The proposed FRUC algorithm improves the accuracy of the motion field for salient regions while enforcing smoothness for non-salient regions. As a preprocessing step, the saliency map is calculated for each frame as according to Eq. (2). In addition, each frame is oversegmented and merged as discussed in the previous section. The proposed algorithm performs the following tasks to refine the input motion field:

- thresholds the saliency map to produce a mask

- for each segmentation region below threshold, enforce region consistency
- for each segmentation region above threshold, apply multi-scale MVR

The saliency map is generated according to [2] using the dynamic texture coefficients for the feature set  $\mathbf{Y}$ . For the dynamic texture model, a spatial window of 8x8 pixels, and temporal window of 11 frames is used. The saliency map is thresholded by  $\tau$ . MVR will take place for each block with a saliency value exceeding  $\tau$  subsequent to the promotion of segmentation consistency.

With the saliency mask computed, we enforce smoothness for non-salient regions. This is accomplished by taking a histogram of the motion vectors within each region and promoting those which occur most frequently. Assume a segmentation of  $n$  regions denoted as  $\{R_1, \dots, R_n\}$ . For each region  $R_i$ , a MV histogram is computed, and the  $m$  most commonly occurring MVs becomes the candidate set:  $CS(R_i) = \{mv_1, \dots, mv_m\}$ . For each candidate  $mv_j \in CS(R_i)$ , the Total SAD (TSAD) is computed. This is the SAD error incurred by applying candidate  $mv_j$  to all blocks within region  $R_i$ . The candidate with the lowest TSAD provides the best description of the overall motion of  $R_i$ . Penalties are applied to the candidate set based on the TSAD of each region. For candidate  $mv_j \in CS(R_i)$ :

$$p(mv_j) = \frac{TSAD(mv_j, R_i)}{\sum_{k \neq j} TSAD(mv_k, R_i)} \quad (5)$$

Finally, the Region Consistent MV ( $mv_{rc}$ ) for a block  $M \in R_i$  is computed.

$$mv_{rc} = \min_{j: mv_j \in CS(R_i)} \sum_{x, y \in M} |f_{t-1} \left( x + \frac{v_{jx}}{2}, y + \frac{v_{jy}}{2} \right) - f_t \left( x - \frac{v_{jx}}{2}, y - \frac{v_{jy}}{2} \right)| p(mv_j) \quad (6)$$

Next, MVR is applied to regions which exceed the saliency threshold. Refinement is computed at multiple scales in order to improve the accuracy of the motion field around object boundaries. This is captured in three stages of decreasing scale. In the first stage, enlarged block matching is considered with a 24x24 pixel measurement window for each 8x8 block. A MV histogram is created containing the original block motion and all spatial neighbors within  $\pm 2$  blocks. These 25 motion vectors are analyzed, and the three most commonly occurring motions, as well as the original block motion, are promoted as a candidate set. As before, the candidate which produces the smallest error is chosen as the motion vector. For stage one, the error is calculated as:

$$SAD_1(v_x, v_y) = \sum_{x, y \in M_1} |f_{t-1} \left( x + \frac{v_x}{2}, y + \frac{v_y}{2} \right) - f_t \left( x - \frac{v_x}{2}, y - \frac{v_y}{2} \right)| \quad (7)$$

$$M_1 = \{x, y : i - 8 \leq x \leq i + 15, j - 8 \leq y \leq j + 15\} \quad (8)$$

where  $M_1$  is defined as in Eq. (8) for a 24x24 pixel enlarged measurement window with upper-left pixel located at  $(i, j)$ . The second stage proceeds in a similar fashion. The candidate set is increased to four motion histogram candidates and the original block motion. An 8x8 block is selected with no enlarged matching to improve the motion accuracy around object boundaries. The error for stage 2 is computed using block  $M_2$ .

$$M_2 = \{x, y : i \leq x \leq i + 7, j \leq y \leq j + 7\} \quad (9)$$

In the third stage, the resolution of the motion field is increased by a factor of two in each direction. Each block is partitioned into four 4x4 subblocks (quadrants), and refinement proceeds as in previous stages. The four subblocks are defined by  $M_{3i}, i = 1, \dots, 4$

$$\begin{aligned} M_{31} &= \{x, y : i \leq x \leq i + 3, j \leq y \leq j + 3\} \\ M_{32} &= \{x, y : i \leq x \leq i + 3, j + 4 \leq y \leq j + 7\} \\ M_{33} &= \{x, y : i + 4 \leq x \leq i + 7, j \leq y \leq j + 3\} \\ M_{34} &= \{x, y : i + 4 \leq x \leq i + 7, j + 4 \leq y \leq j + 7\} \end{aligned} \quad (10)$$

## 5. RESULTS

Both objective and subjective results are presented for the proposed algorithm. In order to compare these methods, four HD sequences have been selected. Each sequence is decimated temporally by removing even frames. These frames are then reconstructed using our method and three competing methods (3D Recursive Search, Full Search and Multi Scale Enlarged Matching.) For each method, the PSNR and SSIM errors are calculated by comparing original even frames with their reconstructed counterparts. Objective results are presented in Table 2 for the salient portions of each sequence. Additional results are presented online <sup>1</sup>. The objective results indicate an increase in the PSNR and SSIM values using the proposed algorithm for the salient region of all four test sequences.

Obtaining high perceptual video quality is crucial for FRUC applications. In order to assess subjective video quality, we conducted a double-blind test on a group of 20 human observers. The method employed was the stimulus comparison non-categorical judgment method as described in [6]. Our test was manifested as a continuous scoring system on the range  $[-3, 3]$ . Each participant was shown a set of side-by-side processed videos using the proposed and competing methods. A positive score corresponds to a preference for the proposed FRUC algorithm when compared against a competing method. Results are displayed in Table 1 with the scores averaged over all 20 participants. A confidence interval of

<sup>1</sup>[http://videoprocessing.ucsd.edu/~NatanHaim/icassp\\_2010/](http://videoprocessing.ucsd.edu/~NatanHaim/icassp_2010/)

**Table 1.** Subjective Testing Results. Positive average score represents a preference for the proposed algorithm.

Sequence	Method	Std. Dev.	Rej.	Avg.
Planes	Prop vs. 3DRS	0.55	0.21	<b>2.24</b>
	Prop vs. FULL	1.26	0.49	<b>1.11</b>
	Prop vs. MSEA	0.77	0.30	<b>1.48</b>
Speedway	Prop vs. 3DRS	0.30	0.12	<b>2.81</b>
	Prop vs. FULL	0.99	0.38	<b>0.78</b>
	Prop vs. MSEA	1.15	0.44	<b>0.85</b>

**Table 2.** Objective results for HD720p sequences using PSNR (first row) and SSIM (second row) metrics. Error measured for 25% of frame with highest saliency index.

Sequence	3DRS	FS	MSEA	Proposed
Dolpins	30.6850	31.8903	31.8539	<b>31.9006</b>
	0.9417	0.9504	0.9511	<b>0.9537</b>
Limit	37.5492	38.2784	38.5500	<b>38.6604</b>
	0.9855	0.9866	0.9871	<b>0.9876</b>
Planes	36.6685	37.1436	37.2119	<b>38.2912</b>
	0.9940	0.9950	0.9950	<b>0.9952</b>
Speedway	25.7847	26.6485	26.6092	<b>26.6846</b>
	0.9335	0.9407	0.9404	<b>0.9411</b>

95% is used, as is the case in the recommendation. As the average scores for the proposed algorithm competing with other methods exceed the rejection region, these results suggest a perceptual improvement of the proposed algorithm for the “Planes” and “Speedway” sequences.

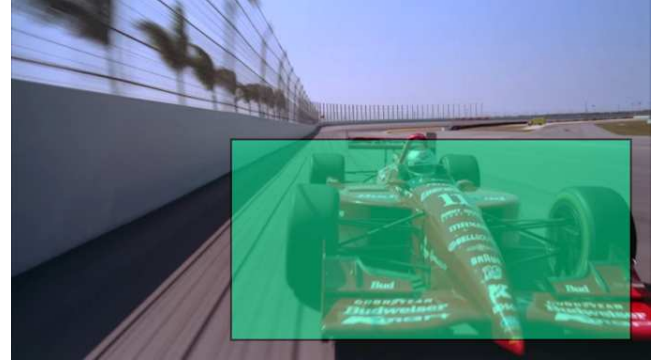
## 6. CONCLUSION

An algorithm to improve MVR for FRUC applications has been proposed and tested in this work. This method has proved beneficial both in terms of objective results and in perceived video quality among observers. In future work, we aim to combine saliency information directly into a scheme for motion estimation, rather than applying it strictly towards refinement.

## 7. REFERENCES

[1] Laurent Itti, “Automatic foveation for video compression using a neurobiological model of visual attention,” *Image Processing, IEEE Transactions on*, vol. 13, no. 10, pp. 1304–1318, Oct. 2004.

[2] Dashan Gao, Vijay Mahadevan, and Nuno Vasconcelos, “On the plausibility of the discriminant center-surround hypothesis for visual saliency,” *Journal of Vision*, vol. 8(7):13, no. 7, pp. 1–18, 2008.



(a) Region with highest saliency value



(b) 3DRS



(c) Full Search



(d) MSEA



(e) Proposed

**Fig. 2.** Interpolated frame 88 of “Speedway” sequence, (a) Bounding box indicates salient portion of frame. For each method, (PSNR, SSIM) values presented: (b) 3DRS: (25.48dB, 0.812), (c) Full Search: (27.02dB, 0.861), (d) MSEA: (26.99dB, 0.861), (e) Proposed: (27.06dB, 0.862)

[3] S. Soatto, G. Doretto, and Ying Nian Wu, “Dynamic textures,” *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 2, pp. 439–446 vol.2, 2001.

[4] G. Mori, Xiaofeng Ren, A.A. Efros, and J. Malik, “Recovering human body configurations: combining segmentation and recognition,” *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2, pp. II–326–II–333 Vol.2, June-2 July 2004.

[5] Jianbo Shi and J. Malik, “Normalized cuts and image segmentation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 888–905, Aug 2000.

[6] ITU-R Recommendation BT.500-11, “Methodology for the subjective assessment of the quality of television pictures,” in *Geneva*, 2002.