MOTION VECTOR REFINEMENT FOR FRUC USING SALIENCY AND SEGMENTATION

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

ECE Department, University of California, San Diego La Jolla, CA 92093-0407 E-mail: {njacobso,yel004,vmahadev}@ucsd.edu, {nuno,nguyent}@ece.ucsd.edu

ABSTRACT

Motion-Compensated Frame Interpolation (MCFI) is a technique used extensively for increasing the temporal frequency of a video sequence. In order to obtain a high quality interpolation, the motion field between frames must be well-estimated. However, many current techniques for determining the motion are prone to errors in occlusion regions, as well as regions with repetitive structure. An algorithm is proposed for improving both the objective and subjective quality of MCFI by refining the motion vector field. A Discriminant Saliency classifier is employed to determine regions of the motion field which are most important to a human observer. These regions are refined using a multi-stage motion vector refinement which promotes candidates based on their likelihood given a local neighborhood. For regions which fall below the saliency threshold, frame segmentation is used to locate regions of homogeneous color and texture via Normalized Cuts. Motion vectors are promoted such that each homogeneous region has a consistent motion. Experimental results demonstrate an improvement over previous methods in both objective and subjective picture quality.

Keywords— Frame Rate Up-Conversion (FRUC), Discriminant Saliency, Motion Segmentation, Motion Refinement, Motion Compensated Frame Interpolation (MCFI).

1. INTRODUCTION

Frame Rate Up-Conversion is an area of significant research with many important applications. In mobile video, bandwidth restrictions make it infeasible to transmit at high frame rates. Instead the focus is on increasing spatial video quality while reducing the number of frames transmitted. FRUC is employed at the receiver to recreate a smooth video. Typically, mobile video is transmitted at 15Hz and up-converted by a factor of two to 30Hz by the FRUC engine. Another important application is motion blur reduction for Liquid Crystal Display (LCD) televisions. This is necessary because of the sample-and-hold nature of LCD displays, which causes noticeable motion blur at low frame rates. LCD displays recently released to the market are capable of displaying at 120Hz and 240Hz, thus significantly reducing the noticeable effect of motion blur. In order to take

advantage of these high frame rates, FRUC is required to upconvert source material to the required rate.

FRUC is composed of two portions: Motion Estimation (ME) and Motion Compensated Frame Interpolation (MCFI). A block-based ME algorithm partitions each frame into uniform blocks (generally 8x8 pixels) and determining the relative translation between each block in successive video frames. The result of the ME step is a motion field for the entire frame. Next, the MCFI engine creates an intermediate frame by interpolating along the motion field direction. Interpolation is performed bidirectionally to avoid any holes in the resultant frame. Given a motion vector (v_x, v_y) from the motion estimator, a block in the interpolated frame f_t is calculated as follows from the current frame f_{t+1} and reference frame f_{t-1} :

$$f_t(x,y) = 0.5f_{t-1}\left(x + \frac{v_x}{2}, y + \frac{v_y}{2}\right) + 0.5f_{t+1}\left(x - \frac{v_x}{2}, y - \frac{v_y}{2}\right)$$
(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 multiple objects with conflicting motion. Another limitation is that the motion vector which minimizes predicted block error may not produce the most consistent motion field. This can occur because of changes in luminance between frames or due to repetitive structures. Finally, FRUC can suffer from a ghosting artifact which is caused by large motions being assigned outside of object boundaries. These shortcomings are addressed in the presented work.

A novel method for FRUC is proposed, 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 very accurate motion vectors. Conversely, motion-vector smoothness and consistency are enforced for non-salient regions using a fast frame segmentation. While these methods are computationally intensive, they provide necessary information in order to increase perceptual quality of salient scene regions.

This work is supported in part by the Center for Wireless Communications at UCSD and by Samsung Information Systems America (SISA).

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, 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]. In this work, a dynamic texture model is used to determine the motionbased feature response.

Dynamic texture data is obtained by determining an Autoregressive Moving Average (ARMA) model for a small piece of spatiotemporal data. This data is a three-dimensional volume with two spatial dimensions and one time dimension. The volume of data represents an observed sequence $\{y(t)\}$ seen as the output of a dynamic texture $\{I(t)\}$ with added noise n(t). Using this notation, the dynamic texture coefficients can be determined using the following process:

$$\begin{cases} x(t) = \sum_{i=1}^{k} A_i x(t-i) + Bv(t) \\ y(t) = \phi(x(t)) + n(t) \end{cases}$$
(2)

where ϕ is a spatial filter, $I(t) = \phi(x(t)), v(t)$ is selected IID from an unknown distribution, and n(t) is selected IID from a given distribution $p_n(\cdot)$.

Discriminant saliency is defined with respect to two classes of stimuli and a feature 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 Y can discriminate between *center* and *surround* at l. This is quantified by the mutual information between features, Y, and class label, C,

$$S(l) = I_l(\mathbf{Y}; C) = \sum_{c=0}^{1} \int p_{Y,C(l)}(\mathbf{y}, c) \log \frac{p_{Y,C(l)}(\mathbf{y}, c)}{p_Y(\mathbf{y})p_{C(l)}(c)} d\mathbf{y}$$
(3)

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 **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. Figure 1 shows the saliency map for the "Speedway" sequence obtained by using dynamic textures. The map shows that the regions predicted

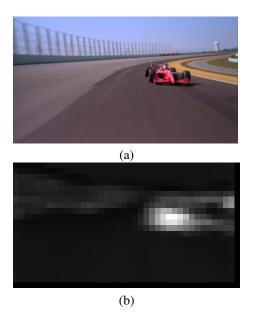


Fig. 1. Discriminant Saliency map using dynamic texture model, (a) input frame from **speedway** sequence, (b) saliency map. Larger pixel intensity (closer to white) represents higher saliency value.

to have high saliency (e.g the car) are indeed the regions that appear visually salient to a human observer.

3. SEGMENTATION

The goal of a segmentation algorithm is to partition each frame into distinct objects. Significant progress has been made on this research topic, although the problem itself is fundamentally illposed. For the scope of this paper, the segmentation algorithm provided in [4] is used, which is based on Normalized Cuts as well as Probability of Boundary (pB) for detection of edges using color and texture. Normalized Cuts treats the image as a connected graph with each pixel being represented by a single node. The algorithm then iteratively cuts the graph so that dissociation between regions is high, while dissociation within regions remains small. As is common in the literature, this segmentation scheme will be used to oversegment the image. Each frame is segmented into a predetermined number of regions determined by the image size. For consistency, this has been fixed at 200 regions for CIF sequences and 400 regions for HD720p sequences.

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. Color information is obtained directly in the RGB color space. It is important to use color rather than relying on luminance information alone, since a boundary between two objects may be isoluminant while varying in chrominance. Texture is also computed as this proves to be a useful cue for merging segmentations. The texture measure is computed as the variance of the AC coefficients of the Discrete Cosine Transform (DCT) of each 8x8 block.

The superpixel merge procedure is posed as a problem over the graph G = (V, E). Here, $\{v_1, \ldots, v_n\} \in V$ 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. $E_{ij} = 0$ if nodes $v_i, v_j \in V$ are non-adjacent. We use an indicator function $b_{i,j}$ to represent node adjacency. $b_{i,j} = 1$ if $v_i, v_j \in V$ are adjacent and $b_{i,j} = 0$ otherwise.

$$E_{i,j} = b_{i,j} \left(\lambda \max\left\{ \boldsymbol{I}_i^{RGB} - \boldsymbol{I}_j^{RGB} \right\} + (1-\lambda) \left| T_i - T_j \right| \right)$$
(4)

$$\boldsymbol{I}_{i}^{RGB} = \frac{1}{|\{v_{i}\}|} \left[\sum_{j \in v_{i}} R(j), \sum_{j \in v_{i}} G(j), \sum_{j \in v_{i}} B(j) \right]^{T}$$
(5)

where 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 λ 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 architecture improves MV accuracy for salient regions while enforcing smoothness of the MV field for non-salient regions. In this way, both objective and subjective video quality will be increased. The proposed architecture is detailed in Algorithm 1.

Algorithm 1 Proposed MV Consistency and Refinement
input: frame data, oversegmented and merged region map
R_1, \ldots, R_n , saliency map S, saliency threshold τ
for region $R_i \in \{R_1, \ldots, R_n\}$ do
if $\frac{1}{ j \in R_i } \sum_{j \in R_i} S(j) < \tau$ then
enforce region consistency for R_i as discussed in Sec-
tion 4.2
else
for all blocks contained in region R_i do
perform MVR as described in Section 4.3
end for
end if
end for
output: refined MV field

4.1. Saliency Map Generation

The saliency map is generated according to [2] with a dynamic texture model used for the feature **Y**. A spatial window size

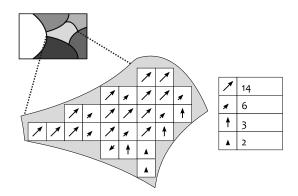


Fig. 2. Region consistency: the upper left portion demonstrates a frame which has been segmented into n = 6 regions. An MV histogram is constructed for region R_3 and the m = 4 most commonly occurring motions are selected for $CS(R_3)$

of 8x8 pixels, and a temporal window size of 11 frames is employed for the spatiotemporal volume. The saliency map is normalized to have a maximum value of 1 pertaining to the most salient points, and a minimum value of 0 for non-salient points. The average saliency value for each region is calculated and compared with a threshold τ to determine whether region consistency or MVR is employed.

4.2. Region Consistency

The result of the frame oversegmentation and merging process is a segmentation with n distinct regions $\{R_1, \ldots, R_n\}$ where $R_1 \cup \ldots \cup R_n = I$. In order to promote *natural mo*tion, we restrict the candidate set of available motions to those which are statistically most likely. A MV histogram is computed for each region R_i consisting of the motions assigned to all blocks $B \in R_i$. From this histogram, the m most commonly occurring motions are promoted as a candidate set. This process is demonstrated in Fig. 2. The parameter m was experimentally determined to be ideal at m = 2 across all sequences tested. Denote the candidate set for region R_i as: $CS(R_i) = \{mv_1, \dots, mv_m\}$. For each candidate mv_j in the candidate set, the Total Error $TE(mv_i, R_i)$ is calculated over region R_i to determine which candidate best explains the total motion of the region. Denote the x and y-components of candidate mv_j as v_{jx} and v_{jy} , respectively. For reference frame f_{t-1} and current frame f_t , TE is computed as:

$$TE(\boldsymbol{m}\boldsymbol{v_j}, R_i) = \sum_{M \in R_i} \sum_{x,y \in M} |f_{t-1}\left(x + \frac{v_{jx}}{2}, y + \frac{v_{jy}}{2}\right) - f_t\left(x - \frac{v_{jx}}{2}, y - \frac{v_{jy}}{2}\right)|$$
(6)

where M is a block contained in region R_i with upper-left pixel index (i, j). Block ownership is determined by which region owns a majority of the block's pixels. Ties are broken arbitrarily. Penalties are applied to these candidates based on the total distortion produced by the candidate for the region R_i . In case of non-integer offsets $(\frac{v_{jx}}{2}, \frac{v_{jy}}{2} \notin \mathbb{Z})$, bilinear interpolation is

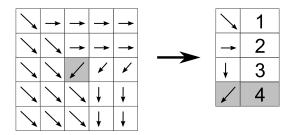


Fig. 3. MVR for center block (gray) with m = 3. The top three most commonly occurring motions in the neighborhood are considered as the first three candidates. The original MV for the center block is the fourth candidate.

used to determine *TE*. For candidate $mv_i \in CS(R_i)$:

$$p(\boldsymbol{m}\boldsymbol{v}_{j}) = \frac{TE(\boldsymbol{m}\boldsymbol{v}_{j}, R_{i})}{\sum_{\boldsymbol{k}\neq j} TE(\boldsymbol{m}\boldsymbol{v}_{\boldsymbol{k}}, R_{i})}$$
(7)

With the penalties determined over the candidate set, we are now able to promote MV consistency for each superpixel region. The Region Consistent MV (mv_{rc}) for a block $B \in R_i$ is computed as:

$$\boldsymbol{mv_{rc}} = \min_{\boldsymbol{j}:\boldsymbol{mv_{j}}\in CS(R_{i})} \sum_{\boldsymbol{x},\boldsymbol{y}\in M} |f_{t-1}\left(\boldsymbol{x}+\frac{v_{j\boldsymbol{x}}}{2},\boldsymbol{y}+\frac{v_{j\boldsymbol{y}}}{2}\right) \\ -f_{t}\left(\boldsymbol{x}-\frac{v_{j\boldsymbol{x}}}{2},\boldsymbol{y}-\frac{v_{j\boldsymbol{y}}}{2}\right) |p\left(\boldsymbol{mv_{j}}\right)$$
(8)

4.3. Motion Vector Refinement

For scene regions which exceed the saliency threshold τ , Motion Vector Refinement (MVR) is applied to increase the accuracy of the motion field. The refinement is computed without motion re-estimation [5]. MVR is computed in three stages of decreasing local neighborhood, which is particularly important at object boundaries, where the MV field is difficult to determine. The method is based on the idea of natural motion, this is the assumption that, for any given area, there are a limited number of motions which need to be considered. The candidate selection process is demonstrated in Fig. 3. MVR is computed in multiple stages in order to improve the accuracy of the motion field around object boundaries. At each stage, the local neighborhood of consideration is decreased in order to consider more relevant MV candidates. 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 a neighborhood of ± 2 blocks. These 25 MVs are analyzed, and the m = 3 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 MV. For stage one, the error is calculated as:

$$SAD_{1}(v_{x}, v_{y}) = \sum_{x, y \in M_{1}} |f_{t-1}\left(x + \frac{v_{x}}{2}, y + \frac{v_{y}}{2}\right) - f_{t}\left(x - \frac{v_{x}}{2}, y - \frac{v_{y}}{2}\right)|$$
(9)

using the Sum of Absolute Differences (SAD) error measure where M_1 is defined as in Eq. (10) 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 .

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, ..., 4

$$M_{1} = \{x, y : x \in [i - 8, i + 15], y \in [j - 8, j + 15]\}$$

$$M_{2} = \{x, y : x \in [i, i + 7], y \in [j, j + 7]\}$$

$$M_{31} = \{x, y : x \in [i, i + 3], y \in [j, j + 3]\}$$

$$M_{32} = \{x, y : x \in [i, i + 3], y \in [j + 4, j + 7]\}$$

$$M_{33} = \{x, y : x \in [i + 4, i + 7], y \in [j, j + 3]\}$$

$$M_{34} = \{x, y : x \in [i + 4, i + 7], y \in [j + 4, j + 7]\} (10)$$

5. EXPERIMENTAL SETUP

Objective results are calculated using the following experimental procedure. Each 24 frame per second (fps) video sequence is temporally reduced by a factor of two to 12fps. The 12fps sequence is then up-converted using MCFI via one of the FRUC algorithms discussed in this paper. The resulting interpolated frames are compared with the originals to determine the error.

5.1. Objective Results

The proposed algorithm is tested against several competing methods for FRUC. Among these are: Full Search (FS) with bidirectional MCFI [6], 3D Recursive Search [7], MSEA method with bidirectional MCFI [8] and a Multistage Motion Vector Processing method (MMVP) [9]. The metrics for comparison are Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM), which models error as perceived by a Human observer. Eight sequences have been chosen for comparison. Among these are four CIF sequences (352x288) and four HD720p sequences (1280x720). The CIF sequences are: coastguard, football, foreman and tennis. These sequences are prevalent in the video processing literature. The HD sequences are: dolphins, limit, planes and speedway. All objective results for these sequences are tabulated in Table 1. Results are also provided for salient scene regions. The top 25% of each saliency map is considered as the mask for calculation of objective results in the salient region. This is consistent with the goal of improving the performance of FRUC must in salient regions.

The **football** sequence is examined in Fig. 4, which presents many different motions, occlusions and therefore numerous object boundaries. For the competing methods, noticeable distortion occurs around the object boundary of player #41 in the middle of the frame. This is especially evident in the 3DRS

 Table 1. Objective results for CIF and HD720p test sequences. Each cell provides results in PSNR dB (top) and SSIM (bottom)

 Sequence
 3DRS [7]
 FS [6]
 MSEA [8]
 MMVP [9]
 Proposed
 3DRS
 FS
 MSEA
 MMVP
 Proposed

Sequence	3DRS [7]	FS [6]	MSEA [8]	MMVP[9]	Proposed	3DRS	FS	MSEA	MMVP	Proposed
CIF	Entire frame					Salient frame region				
coastguard	34.4422	36.9724	37.0120	36.0431	37.5361	32.4071	33.8265	33.8948	33.7229	34.8572
	0.8973	0.9444	0.9448	0.9401	0.9505	0.9833	0.9866	0.9868	0.9858	0.9893
football	24.9455	25.7013	25.7035	24.5524	26.0087	23.0517	23.8816	23.9127	22.5563	24.2448
	0.7422	0.7602	0.7616	0.6847	0.7885	0.9441	0.9511	0.9513	0.9334	0.9571
foreman	37.6367	38.5156	38.5159	34.6369	38.4558	36.1712	37.5657	37.6198	36.4874	37.6827
	0.9413	0.9499	0.9502	0.9416	0.9530	0.9869	0.9904	0.9905	0.9883	0.9911
tennis	31.3513	31.6365	31.5762	28.7834	31.8027	29.7027	30.7389	30.6444	27.3591	31.3106
	0.8689	0.8559	0.8575	0.7393	0.8737	0.9667	0.9664	0.9666	0.9430	0.9733
HD720p	Entire frame					Salient frame region				
dolphins	34.0322	35.1030	35.0952	35.1120	34.9936	30.6850	31.8903	31.8539	31.6042	31.9006
	0.8585	0.8790	0.8814	0.8835	0.8832	0.9417	0.9504	0.9511	0.9497	0.9537
limit	39.3535	39.2591	39.2382	39.4234	39.5608	37.5492	38.2784	38.5500	38.2704	38.6604
	0.9151	0.9156	0.9150	0.9159	0.9209	0.9855	0.9866	0.9871	0.9866	0.9876
planes	34.2114	36.3117	36.2967	36.3942	36.8768	36.6685	37.1436	37.2119	37.1292	38.2912
	0.9258	0.9517	0.9510	0.9469	0.9516	0.9940	0.9950	0.9950	0.9944	0.9952
speedway	28.9685	29.3508	29.3658	29.3960	29.3729	25.7847	26.6485	26.6092	26.6632	26.6846
	0.8517	0.8673	0.8670	0.8638	0.8687	0.9335	0.9407	0.9404	0.9408	0.9411

interpolation in Fig. 4(b). Here, significant blocking artifacts can be seen on the arms of player #41, as well as the leg of the player on the right side of the frame. Interpolation performance increases for FS and MSEA in Figs. 4(c,d), which can be seen in the improved boundary of player #41. However, there are still errors in the leg of the player on the right of the frame. MMVP in Fig. 4(e) combines block motions with high residuals, thus changing the appearance of the leg of the player on the right. The merging of motion vectors creates a consistent motion in this region, however the motion is too large. The result is duplication of the leg appearing as a ghosted copy. Finally, the proposed interpolation in Fig. 4(f) demonstrates consistent motion of the player on the right side of the frame. In addition, the saliency map allows for MVR on player #41, resulting in further improvement over competing methods.

As FRUC is pertinent both to mobile video and HD content, the proposed algorithm is tested against several HD720p sequences. Results for frame 88 of the **speedway** sequence are discussed here. MCFI via 3DRS presents a poor interpolation for the salient car in the foreground of the scene. While FS and MSEA estimate a superior motion field to 3DRS, foreground distortion is still visible. MMVP performs extremely well in this case. Block merging is well suited to correcting artifacts in the background fence region and in the foregruond car region. Finally, the proposed algorithm makes use of saliency and segmentation information to separate the region into a highly salient foreground region (car) and a non-salient background (fence and road). The objective performance is slightly lower than MMVP for the selected frame, but is higher when averaged over the entire video sequence.

5.2. Subjective Results

In addition to objective results, it is crucial to determine the perceptual quality of the proposed algorithm. This is accomplished by performing double-blind subjective testing on a group of Human observers. Subjective results are obtained using the stimulus comparison non-categorical judgment method as described in [10]. A selected group of 20 observers were shown video clips which had been processed by the proposed method, in addition to 3DRS, FS and MSEA. In each instance, two video clips are shown side-by-side with each processed by a different method. The observer is presented with a rating scale on the range [-3, 3], where a score of -3 corresponds with the left side appearing "much better" than the right side, and 3 corresponding with the right side "much better" than the left side. Any score between these two values is acceptable with 0 representing "no difference" between the two sequences. Findings are tabulated in Table 2 for the sequences: football, planes, speedway and tennis across all 20 observers. In this table, the mean (μ) and standard deviation (σ) are calculated for each sequence where a positive score on the mean corresponds to a perceptual improvement of the proposed method over the competing method. The rejection region (γ) is calculated using the Student's T-Test, where a decision is made between the null hypothesis (the proposed algorithm has no positive affect over the competing method) and the alternative hypothesis. Therefore, a mean score exceeding the calculated rejection region corresponds to a statistical improvement of the proposed method. It can be observed from the subjective results that the proposed algorithm demonstrates a significant improvement over the competing methods for both HD sequences. However, no telling results are obtained for the CIF sequences. While the objective results are positive for the CIF sequences, the video size is too small for a significant perceptual improvement.

6. CONCLUSION

There has been significant progress in FRUC research over the past two decades, fueled by the high adoption rate of LCD



(a) Original

(b) 3DRS (27.22dB, 0.7284)



(d) MSEA (27.16dB, 0.7527)

(e) MMVP (26.03dB, 0.7004)

(f) Proposed (27.67dB, 0.7779)

Fig. 4. Objective results for football sequence frame 74 (PSNR (db), SSIM).

Table 2. Subjective testing results. Proposed method is compared with a competing method in each row. Standard deviation is given by σ , rejection region γ and mean μ .

Sequence	Comp. Method	σ	γ	μ
	3DRS	0.50	0.19	2.34
Football	FS	1.02	0.39	0.21
	MSEA	0.74	0.29	-0.15
Tennis	3DRS	1.22	0.47	1.51
	FS	0.51	0.20	0.21
	MSEA	0.86	0.33	0.26
	3DRS	0.55	0.21	2.24
Planes	FS	1.26	0.49	1.11
	MSEA	0.77	0.30	1.48
	3DRS	0.30	0.12	2.81
Speedway	FS	0.99	0.38	0.78
	MSEA	1.15	0.44	0.85

television and increasing demand for mobile video. However, few of these methods are perceptually-based and none consider saliency information for the purpose of increasing perceived video quality. The algorithm presented in this work has addressed these issues and has demonstrated an improvement both in objective and subjective video quality.

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, October 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.

- [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, 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 2004.
- [5] Jeongnam Youn, Ming-Ting Sun, and Chia-Wen Lin, "Motion vector refinement for high-performance transcoding," Multimedia, IEEE Transactions on, vol. 1, no. 1, pp. 30-40, March 1999.
- [6] M.-J. Chen, L.-G. Chen, and T.-D. Chiueh, "One-dimensional full search motion estimation algorithm for video coding," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 4, no. 5, pp. 504-509, October 1994
- [7] G. de Haan, P.W.A.C. 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, 388, October 1993.
- [8] X.Q. Gao, C.J. Duanmu, and C.R. Zou, "A multilevel successive elimination algorithm for block matching motion estimation," Image Processing, IEEE Transactions on, vol. 9, no. 3, pp. 501-504, March 2000.
- [9] Ai-Mei Huang and T.Q. Nguyen, "A multistage motion vector processing method for motion-compensated frame interpolation," Image Processing, IEEE Transactions on, vol. 17, no. 5, pp. 694-708, May 2008.
- [10] "Methodology for the subjective assessment of the quality of television pictures," ITU-R Recommendation BT.500-11, 2002.