

Spatio–Temporal Regularity Flow (SPREF): Its Estimation and Applications

Orkun Alatas, Pingkun Yan, *Member, IEEE*, and Mubarak Shah, *Fellow, IEEE*

Abstract—Feature selection and extraction is a key operation in video analysis for achieving a higher level of abstraction. In this paper, we introduce a general framework to extract a new spatio–temporal feature that represents the directions in which a video is regular, i.e., the pixel appearances change the least. We propose to model the directions of regular variations with a 3-D vector field, which is referred to as *spatio–temporal regularity flow* (SPREF). SPREF vectors are designed to have three cross-sectional parallel components \mathcal{F}_x , \mathcal{F}_y , and \mathcal{F}_t for convenient use in different applications. They are estimated using all the frames simultaneously by minimizing an energy functional formulated according to its definition. In this paper, we first introduce translational SPREF (T-SPREF) and then extend our framework to affine SPREF (A-SPREF). The successful use of SPREF in a few applications, including object removal, video inpainting, and video compression, is also demonstrated.

Index Terms—Cross-sectional parallelism, regularity modeling, spatio–temporal feature, video compression, video inpainting.

I. INTRODUCTION

A N important task of low level video analysis is to extract useful information from a video sequence. The purpose of the extraction is to convert the raw appearance values into meaningful features in order to achieve higher level of abstraction. The choice of features in this process depends on the nature of the problem at hand. In image and video processing, tasks such as motion analysis, compression, and video inpainting, usually require extracting the spatio–temporal features of the data. On the other hand, for other problems, such as key frame extraction, scene segmentation, and database queries, even a simple histogram may sufficiently represent the data. Hence, the complexity of the features may range from simple color histograms, to eigenvalues and eigenvectors, optical flow vectors, wavelet coefficients, and so on, depending on the complexity of the problem.

The regularity direction of a video is an important feature that can be useful in many video processing applications. A video is determined to be regular along the directions where pixel intensities change the least [1]. These directions depend on both the type of the motion and the spatial structure of the scene. There is quite a bit of previous work on spatio–temporal analysis of image sequences in video analysis. A large body of those works focused on motion analysis in the spatio–temporal space. For instance, Heeger [2] proposed to estimate optical flow by using Gabor filter-based spatio–temporal energy models to deal with the aperture problem. Adelson and Bergen [3] started another research direction by showing that the edges of objects moving

in time create 3-D surfaces. Many of the studies that followed used this fact, where the edge maps of the images were first computed, contours from them were extracted in each frame, and then the spatio–temporal surfaces that these contours swept were analyzed [4], [5]. Due to the problems with edge detection and the increasing complexity of video sequences, the more recent studies started using spatio–temporal tensors for particular applications [6], [7]. However, these spatio–temporal features, as well as the extraction methods, are specific for those applications and hence short of generality. It is difficult to apply them for other applications.

In this paper, we propose a systematic approach for finding a new spatio–temporal feature, the local regular directions, along which a spatio–temporal region is regular, i.e., the pixel appearances vary the least. The proposed approach for regularity flow estimation does not rely on edge detection, hence its success does not depend on the presence of strong edges in the scene. Instead it analyzes the whole region, and tries to find the best directions that model the overall regularity of the region. In our work, the directions of regularity are modeled with a 3-D vector field, called the *spatio–temporal regularity flow* (SPREF) field. The strength of SPREF lies in treating the data not as a sequence of 2-D images but as a 3-D volume, and processing all of its information simultaneously. We first introduce the translational SPREF (T-SPREF) with much simplified computation. T-SPREF gives good estimation results when the directions of regularity of the spatio–temporal region is a function of the flow propagation axis. However, the precision of the translational flow model goes down when the directions of regularity depend on multiple axes. In order to deal with such cases, we introduce the affine SPREF (A-SPREF) model. The components of A-SPREF still propagate along one major axis, respectively. However, each component is also a function of the other axes.

II. SPATIO–TEMPORAL REGULARITY FLOW (SPREF)

SPREF (\mathcal{F}) is a 3-D vector field that shows the directions, along which intensity I in a spatio–temporal region Ω is regular, i.e., the pixel intensities in the region change the least. The condition that the intensity should vary regularly in the flow direction can also be perceived as a requirement to follow the directions, in which the sum of directional gradients is minimum. This allows us to write the general flow energy function, for \mathcal{F} as

$$E(\mathcal{F}) = \int_{\Omega} \left| \frac{\partial(I \star H)(x, y, t)}{\partial \mathcal{F}(x, y, t)} \right|^2 dx dy dt \quad (1)$$

where H is a regularizing filter, such as a Gaussian.

The particular definition of SPREF \mathcal{F} depends on the flow model that is used. In this section, we introduce two types of SPREFs based on two different flow models: translational (T-SPREF) and affine (A-SPREF). In the T-SPREF model, we choose one of the main coordinate axes (x , y , or t) to be the axis of flow propagation for simplicity. The magnitude of the flow component along the propagation axis is taken as 1. The

Manuscript received January 30, 2006; revised December 10, 2006. This paper was recommended by Associate Editor T. Fujii.

The authors are with the School of Computer Science, University of Central Florida, Orlando, FL 32816 USA (e-mail: pingkun@cs.ucf.edu; shah@cs.ucf.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2007.893832

magnitudes of the remaining components are determined by minimizing the flow energy function (1) according to the flow models, which is only relevant to the propagation axis. Thus, the components of the SPREF along each propagation direction are translational. The A-SPREF can be considered as a general extension of the T-SPREF model, which also propagates along one major axis. However, each component of the A-SPREF is a function of other axes as well. Therefore, the affine motion and/or complex structure can be captured.

A. Translational (T-) SPREF

In the T-SPREF model, the flows are approximated by block translations orthogonal to the directions of flow propagation. This results in *planar (cross-sectional) parallelism* in the SPREF, which is defined as all the vectors on a plane being uniform (equal in magnitude and direction). In our framework, a *cross-sectional parallel* flow field consists of the following three components: *xy-parallel* (\mathcal{F}_t), *xt-parallel* (\mathcal{F}_y), and *yt-parallel* (\mathcal{F}_x). In an *xy-parallel* flow, the vectors on the *xy* plane of the flow field for a particular *t* are cross-sectional parallel. The planar parallelisms are similarly defined for the *xt* and *yt-parallelism*, where the flow propagation axes are *x* and *y*, respectively. Modeling of SPREF using three cross-sectional parallel components is motivated by the requirement of different applications. For example, in video compression, wavelet basis can be warped along the flow directions to exploit the spatio-temporal redundancy in the video [8]. Depending on the video, this parallelism can be *xy-parallel*, *xt-parallel*, or *yt-parallel*. Parallelism is required to force the warped wavelet basis to be orthogonal. In addition, having three separate components provides more flexibility to the scheme. The physical meaning of each component can be easily exploited. For example, a moving object can be efficiently removed from a video by using only the *xy-parallel* flow, which describes the motion regularity of the video. Similarly in video inpainting, if the missing part undergoes global motion, the spatio-temporal hole can be completed by using only the *xy-parallel* flow, which can greatly simplify the inpainting process.

Since the *xy-parallel* flow propagates in the temporal axis, it models the regularity that depends on the motion in a spatio-temporal region Ω . The other two flow types, *xt-parallel* and *yt-parallel*, can model the temporal regularity to some extent but they can also model the spatial regularity of Ω when there is no motion. All the three components of SPREF can be formulated by discretizing the continuous flow energy function (1), and tailoring it according to how \mathcal{F} is defined. For more information on computing the T-SPREF, please refer to the detailed description in [9].

B. Affine (A-) SPREF

The T-SPREF gives good results when the directions of regularity of the spatio-temporal region is a function of the flow

propagation axis. In other words, when the motion is translational, or in the absence of motion when all the edges in the scene extend along the same direction, T-SPREF performs well. However, the precision of the translational flow model goes down when the directions of regularity depend on multiple axes. This is the case, for example, when the motion is zooming in/out or rotation, where the true flow is not only a function of time, but also a function of spatial location. Similarly, in the absence of motion, when two edges extend in different directions along the *x* or *y* axes, T-SPREF cannot find the correct directions of regularity. For such cases, we change the flow model from translational to affine, where the flow still propagates along one major axis, however, it is a function of all the axes.

Since the flow vector field \mathcal{F} is defined according to the affine model, the general flow energy function (1) is expanded accordingly. When the propagation axis is *t*, \mathcal{F} is defined as $\mathcal{F}_t = (c'_1(x, y, t), c'_2(x, y, t), 1)$, and formulated by (2), shown at the bottom of the page, where

$$\begin{bmatrix} c'_1(x, y, t) \\ c'_2(x, y, t) \end{bmatrix} = \begin{bmatrix} a_{11}(t) & a_{12}(t) & a_{13}(t) \\ a_{21}(t) & a_{22}(t) & a_{23}(t) \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \quad (3)$$

Just like in T-SPREF, the flow parameters $a_{ij}(t)$ can be obtained by directly solving the flow energy function (2). However, since we want to achieve a global solution that uses all the information in the spatio-temporal region Ω , we approximate these parameters by splines. Hence, $a_{ij}(t)$ is expanded as

$$a_{ij}(t) = \sum_n \alpha_n^{ij} b(2^{-l}t - n) = \sum_n \alpha_n^{ij} b_n^l(t). \quad (4)$$

The spline function b used in our experiments is defined as

$$b(z) = \begin{cases} 1 - |z|, & \text{if } |z| < 1 \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

The parameters for the flows propagating along the *x* or *y* axis can be estimated in a similar way.

The first row of Fig. 1 shows a synthetic sequence generated from the Lena image, where her eye is zoomed in successive frames. The directions of regularity obtained by an *xy-parallel* T-SPREF is shown in the second row of Fig. 1, where it can clearly be seen that the translational approximation cannot estimate the underlying motion. On the other hand, since A-SPREF can handle this type of motion, the estimated flow vectors in the third row of Fig. 1 reveal the true directions of regularity for the image sequence.

After finding the flow directions, the next step in A-SPREF is computation of the *flow curves*. Let's assume that the axis of propagation is *t*. The flow $(c'_1(x, y, t), c'_2(x, y, t), 1)$ only maps the pixels in frames *t* to *t+1*. To compute the flow curves, however, one needs to map the pixels in one frame to all the others, with a new set of parameters. Given two sets of affine parameters estimated according to (4): $A_{t \rightarrow t+1} = \{a_1, a_2, \dots, a_6\}_{t \rightarrow t+1}$ and $A_{t+1 \rightarrow t+2} = \{b_1, b_2, \dots, b_6\}_{t+1 \rightarrow t+2}$, the *propagate*

$$E(\mathcal{F}) = \sum_{\Omega} \left| \left(F \star \frac{\partial H}{\partial x} \right) c'_1(x, y, t) + \left(F \star \frac{\partial H}{\partial y} \right) c'_2(x, y, t) + F \star \frac{\partial H}{\partial t} \right|^2 \quad (2)$$

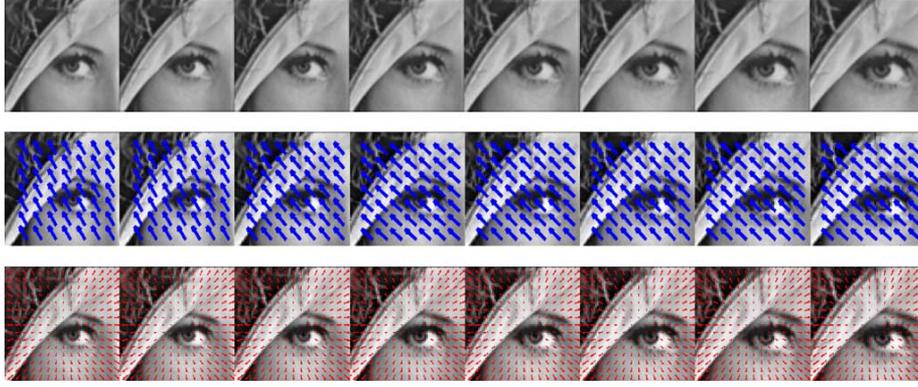


Fig. 1. First row: A synthetic sequence from the Lena image where Lena's eye is zoomed in successive frames. Second row: T-SPREF approximation to the underlying directions of regularity, shown with blue flow vectors superimposed on the images. Since zooming in is approximated by translations, the approximation is not successful. Third row: A-SPREF approximation of the directions of regularity. The flow vectors are clearly more precise than T-SPREF.

operation \mathcal{G} produces the new parameter set $\hat{A}_{t \rightarrow t+2} = \{\hat{a}_1, \hat{a}_2, \dots, \hat{a}_6\}_{t \rightarrow t+2} = \mathcal{G}(A_{t \rightarrow t+1}, A_{t+1 \rightarrow t+2})$ as follows:

$$\begin{aligned}\hat{a}_1 &= a_1 + b_1 * (1 + a_1) + b_2 * a_4 \\ \hat{a}_2 &= a_2 + b_2 * (1 + a_5) + b_1 * a_2 \\ \hat{a}_3 &= b_3 + a_3 * (1 + b_1) + b_2 * a_6 \\ \hat{a}_4 &= a_4 + b_4 * (1 + a_1) + b_5 * a_4 \\ \hat{a}_5 &= a_5 + b_5 * (1 + a_5) + b_4 * a_2 \\ \hat{a}_6 &= b_6 + a_6 * (1 + b_5) + b_4 * a_3.\end{aligned}\quad (6)$$

With the propagate operation \mathcal{G} in mind, the new parameters that will be used to compute the flow curves are written as $\hat{A}_{0 \rightarrow t+1} = \mathcal{G}(\hat{A}_{0 \rightarrow t}, A_{t \rightarrow t+1})$, where $\hat{A}_{0 \rightarrow 1} = A_{0 \rightarrow 1}$. After the new set of parameters are computed, the flow curve coordinates that they imply are stored in coordinate grids, which are given as $(\hat{a}_{11}(t)x + \hat{a}_{12}(t)y + \hat{a}_{13}(t), \hat{a}_{21}(t)x + \hat{a}_{22}(t)y + \hat{a}_{23}(t))$ for the SPREF propagating along axis t . When the flow propagation axis is x or y , the same algorithm can be applied after doing the necessary change of variables.

C. Modeling Nonuniform Regularities

Extending the SPREF framework to model the whole video is a must in many video applications. SPREF is designed to compute the local directions of regularity of a spatio-temporal region, and the whole video can be considered as a local region only when it undergoes global motion, or when there is no motion and the spatial structure of the scene is simple. However, usually this is not the case; the videos often are mixtures of regions with both local and global motion. Also, the scenes are usually highly textured, which hurts the xt and yt -parallel SPREF approximations. In such cases, the video needs to be segmented into smaller spatio-temporal regions until the regularity of each region is uniform. Then the SPREF can be computed. To do this, the video is first divided into *group(s) of frames* (GOF) and then each GOF is partitioned into smaller *subgroups of frames* (sub-GOF) using an octree. This segmentation allows us to analyze the regularity of the GOF at multiple locations and various sizes. This step may or may not be followed by merging the sub-GOFs depending on the application. The quality of each SPREF is determined by a metric, specific for the goal of the application. For instance, the metric can be

the flow error for inpainting applications, or the total bit cost for compression applications.

III. APPLICATIONS

In this section, the applications of SPREF in object removal, video inpainting, and video compression are demonstrated. The results obtained using SPREF are also compared with other state-of-the-art approaches for performance evaluation. Note that SPREF is a general framework. Here we just demonstrate a few example applications. The use of SPREF is certainly not limited to the applications shown here.

A. Object Removal

Object removal is to remove a target object from the video [10]. In many video applications, this is one of the key techniques for video processing. Unfortunately, manual selection of the object from each frame is normally required to remove the object. The procedure is labor intensive and therefore time consuming. However, by using the SPREF, the amount of manual work can be significantly decreased. The basic idea is to remove the moving object by following the xy -parallel SPREF curve. In our approach, the manual selection is only required for the first and the last frames of the GOF. The object in the other frames can then be removed automatically by using the xy -parallel SPREF.

Fig. 2 shows an example of object removal using the SPREF. The objective there is to remove the airplane from the video. SPREF was computed from the original video sequence. The airplane in the first and the eighth frames is erased manually. As shown in Fig. 2, the airplane in the other six frames is then removed successfully, although the background is also moving. Compared to the way of manual selection in each frame, the amount of manual work has been reduced by 75%.

B. Video Inpainting

When an object is removed from a video, it leaves a spatio-temporal hole behind. Video inpainting is filling this hole naturally, while preserving the video's temporal regularity. In previous studies, this regularity has been preserved *implicitly* by various techniques, which may cause the inpainted results inconsistent. The problem can be solved by using SPREF since the regularity of a spatio-temporal region is modeled *explicitly*. In this section, we first explain how to inpaint a group of frames

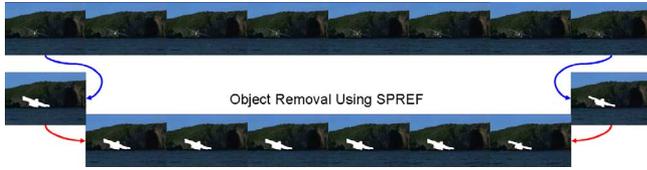


Fig. 2. Example of object removal. The first row shows the original video frames containing an airplane. In the first and the last frames, the airplane is manually removed. It is then automatically removed from other frames according to the SPREF computed from the original sequence.

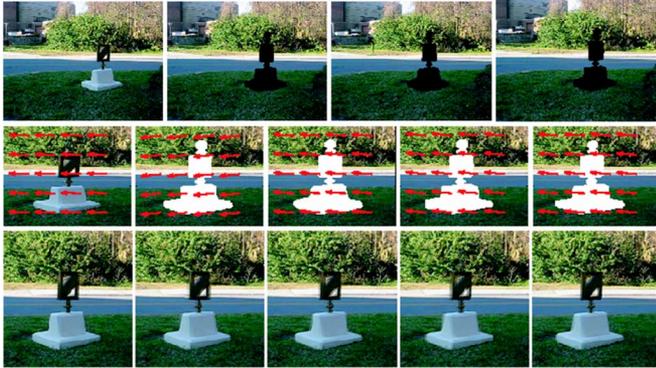


Fig. 3. First row: Frames of a video sequence, where the statue is artificially removed from the midframes. Second row: SPREF directions are superimposed on their respective frames. Third row: Results of the SPREF-based video inpainting.

GOF, when the motion of the pixels surrounding the spatio-temporal hole can be modeled by a single SPREF. Next, we extend this solution to the cases, where the hole may lie on the motion boundaries of the frames. We present an algorithm based on the segmentation of the video for this purpose.

1) *Inpainting With Single SPREF*: When a group of frames that undergo global motion contain a spatio-temporal hole, we can assume that the directions of regularity of the hole is the same as that of the pixels surrounding it. The first row of Fig. 3 shows a clip from the Statue sequence, where the statue is removed from frames in the middle. Since the video undergoes a global translational motion, the hole can be inpainted using only the *xy*-parallel T-SPREF component. Although the sequence is damaged by the removal of the statue, its SPREF can still be computed from the undamaged pixels. In order to do this, a sub-GOF that fully contains this spatio-temporal hole is automatically selected, and then T-SPREF energy function is solved to find the *xy*-parallel flow directions. These directions, along which the sub-GOF varies the least, are converted to a set of coordinates by computing the *SPREF* curves.

The second row of Fig. 3 shows the selected sub-GOF, and its SPREF vectors superimposed on each frame. Note that the flow directions are parallel in each frame, due to the constraint of *xy*-parallelism. The flow directions reveal certain coordinates, represented by *SPREF* curves, along which the sub-GOF varies the least. Hence, we fit a spline to the known pixel intensities on the flow curve, and interpolate the missing appearance values from this spline. Note that we allow no damaged pixels at the sub-GOF boundaries to guarantee avoiding any extrapolations that may result in extremely high or low intensity values in this case. The third row of Fig. 3 shows the results of our inpainting



Fig. 4. First row: Clip of the original walking human sequence. Second row: Sequence after removing the sign board. Third row: Inpainting results of our algorithm based on the A-SPREF. Fourth row: Inpainting results of our implementation of [11]. Last row: Zoomed view of the inpainted human in the sequence where (a)–(c) are the results of our algorithm and (d)–(f) are the results of the method in [11].

algorithm, where we inpaint the Statue back in the intermediate frames successfully.

2) *Moving Object Inpainting*: In many natural video sequences, we need to inpaint the hole left by removing a static object, which partly occludes another moving object. In such cases, the single SPREF inpainting algorithm does not work and a more sophisticated method is needed. The first row of Fig. 4 shows a clip from the walking human sequence in which a man is partly occluded by a sign board. In the second row, the sign board is removed and we need to inpaint the hole marked by red. In this example, the spatio-temporal hole is on the boundary of the background and the walking human. The background is static while the man is moving forward. Since there exist multiple directions of regularities due to multiple layers of motion, a single SPREF is not able to handle it. The solution lies in segmenting the GOF, which creates many sub-GOFs so that each sub-GOF contains a unique SPREF. Then we perform the inpainting along the flow curves with interpolation and/or extrapolation. Our inpainting results are shown in the third row of Fig. 4.

For comparison, we also implemented the method proposed by Wexler *et al.* [11] for inpainting the same video sequence. The results are shown in the fourth row of Fig. 4. Their method repairs the video by searching the missing part blindly in global spatio-temporal space, which is very time consuming. Furthermore, in their work, each pixel is processed independently and the color value associated with the pixel is obtained by computing the weighting average of the matched patches. Although the colors of the recovered pixels are consistent with

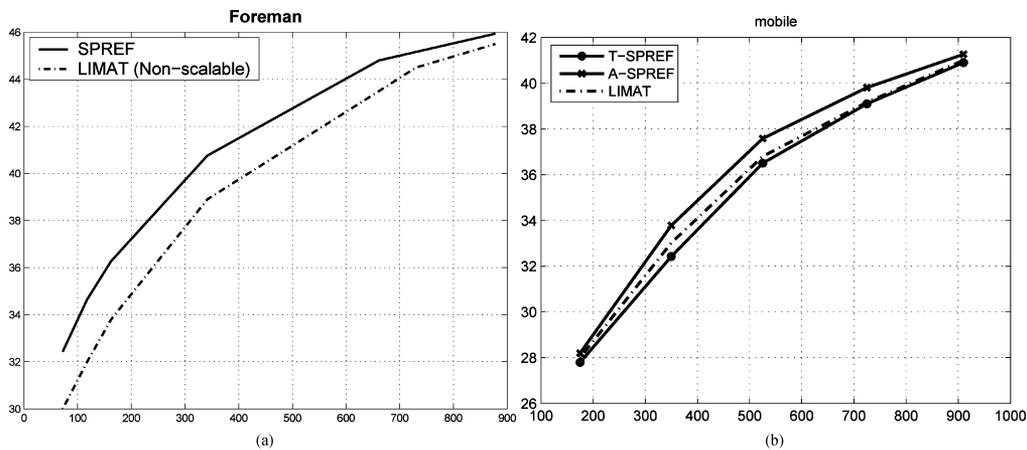


Fig. 5. Bit rate versus peak signal-to-noise ratio (PSNR) plots of (a) Foreman and (b) Mobile. Both SPREF-based compression and LIMAT framework are shown in the results.

those of their surroundings, the inpainting result sometimes may not be meaningful [see (d)–(f) in the last row of Fig. 4]. On the contrary, our video inpainting algorithm is based on the computed SPREFs, which provide the regularity directions. Therefore, once the SPREF is available, the missing parts of the video can be easily recovered by interpolation. In addition, the basic unit of inpainting is sub-GOF with unique SPREF, which facilitates the integrity of the video [see (a)–(c) in the last row of Fig. 4].

C. Video Compression

According to the information theory, lower entropy results in higher compression ratio. Thus, if a spatio-temporal region Ω is filtered along the directions of regularity, where entropy is lower, better compression can be obtained. Since SPREF indicates the directions of regularity, it is a very suitable tool to increase the efficiency of the compression. Moreover, its compactness due to the spline representation has a low compression overhead. SPREF-based video compression can be possible by warping the 3-D wavelet basis along the flow directions. For this purpose, a warping operator \mathcal{W} is defined so that the filtering can be performed on the flow curves. The compression can be further improved by converting the warped wavelet basis into a *bandelet* basis, which was introduced by Pennec and Mallat [12]. The efficiency of the compression depends on the closeness of the approximation of the regularity. For more details of the algorithm, please refer to [9].

We show the results of our SPREF based bandelet video compression scheme on some standard Foreman and Mobile video sequences. All sequences are at QCIF resolution. We also compare the results of our algorithm with those of the Lifting-based invertible motion adaptive transform (LIMAT) framework of Secker and Taubman [13], a motion-compensated wavelet video compression technique, for performance evaluation.

We give a comparison of the SPREF-based compression and wavelet video compression at various bit rates in Fig. 5. The improvement as a result of the directional decomposition and bandeletization in SPREF-based compression can be clearly observed in these plots. In the experiment on Foreman sequence, T-SPREF is used since the motion in this sequence is basically translational. Fig. 5(b) shows the compression results using LIMAT and our algorithm based on T-SPREF and

A-SPREF, respectively. It is noted that the A-SPREF based algorithm performs best, while LIMAT outperforms T-SPREF in this case. The reason is that the motion in the Mobile sequence consists of many nonrigid components such as global zooming out, the swinging toy, and the rotating ball. As we have discussed in Section II-B, the T-SPREF is not able to approximate nonrigid motion of the objects well. The mesh model used in LIMAT can model some of these nontranslational motion types better than SPREF. Hence, LIMAT performs marginally better than SPREF in this particular example. However, the nonrigid motion can be well approximated by A-SPREF and the video compression algorithm based on A-SPREF consequently performs much better than the other two.

IV. CONCLUSION

We presented a new general framework called SPREF that shows the local directions, along which a spatio-temporal region changes the least. SPREF is a 3-D vector field that approximates these directions with splines. In terms of an image sequence, using splines allows us to incorporate all frames in the solution, which results in a better estimation. The directions of regularity depend on the motion content and the spatial structure of the region. All these cases are handled by three components of SPREF. We have shown successful use of SPREF in three popular applications: object removal, video inpainting, and wavelet based video compression.

REFERENCES

- [1] H.-H. Nagel and A. Gehrke, H. Burkhardt and B. Neumann, Eds., "Spatio-temporally adaptive estimation and segmentation of OF-fields," in *Proc. Eur. Conf. Comput. Vision*, 1998, vol. 2, pp. 86–102.
- [2] D. J. Heeger, "Optical flow using spatio-temporal filters," *Int. J. Comput. Vis.*, pp. 279–302, 1988.
- [3] E. H. Adelson and J. Bergen, "Spatio-temporal energy models for the perception of motion," *J. Opt. Soc. Amer.*, vol. 2, pp. 284–299, Feb. 1985.
- [4] P. Anandan, "A computational framework and an algorithm for the measurement of visual motion," *Int. J. Comput. Vis.*, vol. 2, no. 3, pp. 283–310, 1989.
- [5] W. T. Freeman and E. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 9, pp. 891–906, Sep. 1991.
- [6] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "Motion analysis and segmentation through spatio-temporal slices processing," *IEEE Trans. Image Process.*, vol. 12, no. 3, pp. 341–355, Mar. 2003.

- [7] I. Laptev and T. Lindeberg, "Space-time interest points," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, pp. 432–439.
- [8] O. Alatas, O. Javed, and M. Shah, "Video compression using spatio-temporal regularity flow," *IEEE Trans. Image Processing*, vol. 15, no. 12, pp. 3812–3823, Dec. 2006.
- [9] O. Alatas, O. Javed, and M. Shah, "Spatio-temporal regularity flow (SPREF): Its Estimation and applications," University of Central Florida, Orlando, Tech. Rep. 122606, 2006 [Online]. Available: <http://www.cs.ucf.edu/~vision/publications.html>
- [10] Y. Zhang, J. Xiao, and M. Shah, "Motion layer based object removal in videos," in *Proc. IEEE WACV'05*, 2005, pp. 516–521.
- [11] Y. Wexler, E. Shechtman, and M. Irani, "Space-time video completion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2004, vol. 1, pp. 120–127.
- [12] E. L. Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 423–438, Apr. 2005.
- [13] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1029–1041, Aug. 2004.