Video Compression Using Structural Flow

Orkun Alatas, Omar Javed and Mubarak Shah
School of Computer Science
University of Central Florida
4000 Central Florida Blvd
Orlando, FL 32816
Email: alatas, ojaved, shah@cs.ucf.edu

***Abstract*— This paper proposes a new technique in wavelet video compression that exploits the spatiotemporal regularity of the video. A sequence of frames is said to be regular along the directions in which the pixels vary the least. The directions of regularity of a sequence depend on both its motion content and its spatial structure. We model these directions by a 3D vector field, referred as the *Structural Flow*. This flow determines the paths of regularity along which the entropy of the data is smaller. We use these paths to construct a special class of wavelet basis, i.e., the *3D orthonormal bandelet basis* for the directional decomposition of the sequence. Our experiments on several standard video sequences demonstrate the significant improvement in compression compared to the standard wavelet video coding.**

I. Introduction

Video compression is a very important part of many applications, such as video-conferencing, video storage, and broadcasting, since their performance largely relies on the efficiency of the compression. The wavelet coding, which proved to be very efficient in image compression, is also used in this area since it outperforms the standard DCT (Discrete Cosine Transform) based methods, such as MPEG1 and MPEG2.

In standard wavelet video coding, a *group of frames* (gof) is decomposed along the three major axes: temporal, horizontal and vertical. However, this decomposition does not take the regularity of the gof into account. In the presence of global motion, uniform 3D paths of regularity are defined in a gof, which extend along the direction of motion. The situation gets more complicated when the motion is a mixture of the local and global components. In this case, *subgroups of frames* ($subgof$s) with different motion types result in multiple directions of regularity. One way of modelling this regularity is modelling the motion. The pixel correspondence information over multiple frames gives the directions of regularity of the gof. The motion-compensated (MC) wavelet coding algorithms use this approach. The choice of the motion model is an important factor in such algorithms, as its precision and compressibility directly affect the bit rate. In the recent literature, the researchers have used dense motion fields modelled by Markov Random Fields [1] and deformable triangular meshes ([2]). All these models, however, use only consecutive pairs of frames to compute the directions of regularity of the whole gof. Hence, the overhead is a problem since the temporal redundancy in the model cannot be removed when frame pairs have similar motions. Moreover, the (MC) wavelets reduce to the standard wavelets when there is no motion in the gof. This means that it cannot exploit the spatial regularity of the frames.

In this paper, we propose to model the spatiotemporal directions of regularity of a gof by a 3D vector field, called the *structural flow*. The structural flow can be modelled in different ways, depending on whether the regularity is spatial or spatiotemporal. Once the flow is computed, the wavelet basis can be warped along the directions of regularity to decompose the gof. Then the warped basis is *bandeletized*, a technique first introduced by Mallat et al in [3], in order to take further advantage of the regularity. The overall compression requires partitioning the gof into $subgofs$, whose regularities can be as closely modelled as possible by their respective structural flows. This is achieved by using an oct tree segmentation of the gof, such that the reconstruction error and the bit rate of the gof are optimized.

In this paper, Section II explains the main steps of constructing a bandelet basis for a $subgof$: Section II-A goes into the details of *structural flow*, presenting some mathematical background. Section II-B and II-C describe how this flow can be used to construct a bandelet basis. Next, we discuss the optimal segmentation of a gof into $subgof$s in Section III. Finally, we demonstrate our results on standard video sequences in Section IV, and conclude with a discussion in Section V.

II. The Orthonormal 3D Bandelet Basis

In wavelet video coding, the efficiency can be improved by analyzing the directions of regularity of the gof (F), which are represented by the *structural flow*. Unlike the standard wavelets, the orthonormal bandelets can greatly benefit from this direction information, and can achieve higher compression rates. In this section, we will explain the main steps of constructing a bandelet basis.

A. The Structural Flow

The structural flow, $\zeta(x, y, t)$, is a 3D vector field that shows the directions, in which a $subgof$ (F_i) varies regularly. Decomposing the $subgof$ along this field requires its directions to be orthogonal. This is guaranteed by a *planar (cross-sectional) parallelism* of the flow field, which is defined as all the vectors on a plane being equal in magnitude and direction. In our framework, a *cross-sectionally parallel* flow field can belong to one of the following three classes: (1) $x-y$ *parallel*, (2) $x-t$ *parallel*, and (3) $y-t$ *parallel*.

The $x-y$ *parallel structural flow* models the temporal directions of regularity in the frames. The other two classes of structural flow, i.e., $x-t$ and $y-t$ *parallel* flows, generally model the spatial regularity of the $subgof$. The regularity condition that the structural flow needs to satisfy can also be perceived as a requirement to follow the directions, in which the sum of the directional gradients is minimum. Describing the problem in this way allows us to write a continuous flow energy equation for ζ as

$$E(\zeta) = \int_{F_i} \left| \frac{\partial(F \star H)(x,y,t)}{\partial\zeta(x,y,t)} \right|^2 dxdydt, \qquad (1)$$

where H is a regularizing filter, such as a Gaussian. This equation can be discretized, and then tailored to different types of parallelism depending on how ζ is defined. If the flow is $x-y$ *parallel*, then $\zeta(x,y,t) = \zeta_{xy}(t) = (c'_1[t], c'_2[t], 1)$, resulting in

$$E(\zeta_{xy}) = \sum_{F_i} \left| (F \star \frac{\partial H}{\partial x})c'_1[t] + (F \star \frac{\partial H}{\partial y})c'_2[t] + F \star \frac{\partial H}{\partial t} \right|^2. \qquad (2)$$

Notice that the formulation of $\zeta_{xy}(t)$ implies that the x and y components of the flow, $(c'_1[t], c'_2[t])$, are functions of time only, which is constant for all the pixels of a certain frame, i.e., $x-y$ cross-section of the $subgof$. Fig. 1(a) shows the frames of a synthetic gof, which has been sampled from the Lena image by imitating a global translational motion in the diagonal direction. Hence the direction of motion for all frames is uniform. Fig. 1(b) shows the subsampled $x-y$ *parallel* flow field (shown with blue arrows), and the $x-y$ cross section of the flow at $t=1$, superimposed on the first frame of the synthetic gof. The flow equations can be written similarly when the flow class is $y-t$ *parallel*, $\zeta_{yt}(x) = (1, c'_2[x], c'_3[x])$, or $x-t$ *parallel*, $\zeta_{xt}(y) = (c'_1[y], 1, c'_3[y])$. Due to space considerations, we will describe our method only for the $x-y$ *parallel* flow. However, note that the same formulas apply to all flow classes by doing a simple change of functions.

A very important requirement on the flow representation is that it should be compact, such that the flow overhead is minimum. In order to satisfy this condition, the directions, $c'_m[u]$ ($m \in \{1,2,3\}$), are approximated with 1^{st} degree translated box spline functions ($S(u)$), as $c'_m[u] = \sum_n \alpha_n^m S(2^{-l}u - n)$, where α_n ($n = 1...2^l$) is the n^{th} spline coefficient, $l = 1..k$ is a scale factor, 2^k is the width of F_i on the axis along which the flow is not parallel, and u is the index of this axis. With this representation, the whole flow can be recovered by storing

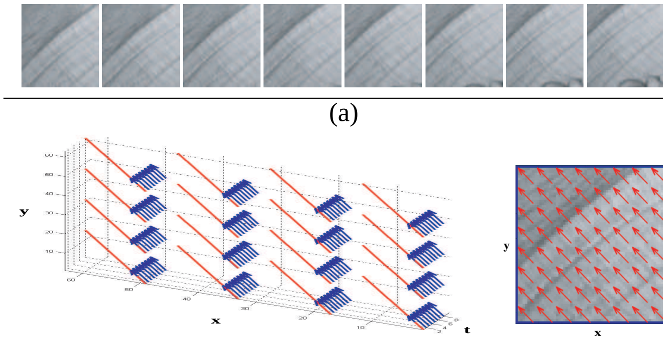

Fig. 1. The $x-y$ *parallel* structural flow field for a gof with a global motion along the diagonal direction. (a) The original sequence (8 frames). (b) (Left) The $x-y$ *parallel* flow field. (Right) The $x-y$ cross-section of the flow field at $t=1$, superimposed on the first frame of the gof.

only the spline coefficients. The function, $S(u)$, is formulated as $S(u) = 1 - |u|$ if $|u| < 1$ and 0 otherwise. The coefficients α_n can be solved for by quadratic minimization of the energy function (1), the choice depending on the parallelism class. In the final step, the spline coefficients are quantized.

The conversion of the flow directions into actual spatiotemporal paths of regularity requires computing the *structural flow curves*. A structural flow curve, $c[u]$, is an integral curve, whose tangents are parallel to ζ. It defines the paths, on which F_i varies regularly, so that the wavelet basis can be warped along the directions of regularity. The structural flow curve in the discrete domain is given by the equation,

$$c[u] = \sum_{k=1}^{u} c'[k], \qquad (3)$$

The coordinates of an $x-y$ *parallel* flow curve are given as, $(x + c_1[t], y + c_2[t], t)$ for a constant (x,y) and a varying t. The curves of the flow in Figure 1 are shown by the red lines on the vector field.

B. Constructing the Orthonormal Bandelet Basis

The standard $3D$ wavelets decompose the data along the three major axes.

However, if the directions of regularity are known, then the data can be decomposed along those directions using the bandelets. The construction of this basis consists of two steps: (1) Warping the standard $3D$ wavelet basis, (2) *Bandeletization*.

The 3D wavelets can be warped along the flow curves with the operator W, which is defined as $W_{xy}(F_i(x,y,t)) = F_i(x + c_1[t], y + c_2[t], t)$ for the $x-y$ *parallel* flow, $\zeta_{xy}(t)$. Decomposing the $subgof$ along the directions of regularity, where the entropy is lower, results in less number of significant wavelet coefficients.

After warping the wavelet basis, the last step is the *bandeletization*. The wavelet function family $\{\psi(t)\}_{j,m3}$ consists of high-pass filters and it has a vanishing moment at lower resolutions. The scaling function family $\{\phi(t)\}_{j,m3}$, however, consists of low-pass filters and it does not have a vanishing moment at lower resolutions. Hence, it cannot take advantage of the regularity of the gof along the flow curves. This problem is solved by *bandeletizing* the warped wavelet basis, i.e., replacing $\{\phi(t)\}_{j,m3}$ with $\{\psi(t)\}_{l,m3}$ for $l > j$. The bandeletization further decreases the number of significant wavelet coefficients.

C. The Orthonormal Bandelet Decomposition

Decomposing a $subgof$ with the orthonormal bandelet basis can be implemented by slightly modifying the fast discrete wavelet transform. Once the structural flow is known, the warped wavelet transform can be computed by a subband filtering that goes along the flow curves.

The subband filtering can be implemented by using the lifting scheme [4], which requires that the neighbors of a point be known. In warped wavelet filtering, these neighborhoods are defined on the flow curves. The curve coordinates are stored in a grid $G(x, y, t)$. For the $x - y$ *parallel* flow, $G(x, y, t) = (x+c_1[t], y+c_2[t], t)$ when $(x+c_1[t], y+c_2[t], t)$ is in the limits of F_i. Since the $x - y$ *parallel* flow curves are the sets of points with fixed (x, y) and varying t, the pixels stored at the locations G$(x, y, t - 1)$, G(x, y, t) and G$(x, y, t + 1)$ are temporal neighbors on the same curve. The spatial neighborhoods of the pixels, on the other hand, are still defined based on their spatial coordinates, not the flow curves.

After computing the warped wavelet coefficients, the next step is the bandeletization. The coefficients resulting from the scaling function are further decomposed by subband filtering at lower resolutions. This concludes the bandelet transformation of the $subgof$.

The reconstruction of the $subgof$ is implemented by inverting the decomposition steps. After reconstructing G, the rest is simply an inverse bandeletization that recovers the warped wavelet coefficients, followed by an inverse subband filtering of these coefficients along the flow curves.

III. VIDEO COMPRESSION

So far, the bandelet decomposition has been defined for a $subgof$, where the compression rate depends on how well the structural flow models the directions of regularity. Direct extension of this method to a gof does not always work because a gof usually has multiple directions of regularity due to different types of motions taking place in the video, and/or the different spatial arrangements in it.

The solution is segmenting a gof into $subgof$s, such that the directions of regularity of each $subgof$ is as closely estimated as possible. This is directly related to minimizing the compression cost of each F_i such that the total cost, $D + \lambda R = \sum_i D_i + \lambda R_i$ is minimized, where D_i is the sum of squared reconstruction error of F_i, R_i is the bit cost of the bandelet and flow coefficients, and λ is a Lagrange multiplier. In order to achieve this segmentation, we initially partition the gof into rectangular prisms (cuboids) using an *oct tree*. The width of each dimension of a cuboid is 2^{k_j}, where $j \in \{1, 2, 3\}$ denotes the particular dimension. The bandelet coefficients are uniformly quantized with the quantization parameter, Δ, and then encoded. Since the bit cost of these coefficients is almost proportional to the number of non-zero coefficients, as shown in [5], the bit cost of the bandelet coefficients can be approximated as $R_{b,i} = \gamma_0 M_i$, where M_i is the number of non-zero coefficients and $\gamma_0 = 7$. The bit rates that we will present in the next section will be based on this approximation.

The choice of λ as a function of the quantization parameter Δ can be computed by minimizing the total cost equation with respect to Δ. This minimization results in the definition of λ as $\lambda = \frac{3\Delta^2}{4\gamma_0}$. The minimization of the total cost starts with computing the cost of all cuboids in the oct tree. The cost, $(D_i + \lambda R_i)$, can be minimum for only one of the four flow classes, including the no-flow case. When computing the cost of a certain flow class, the scale parameter l $(1 \leq l \leq k)$ in the spline equation is found by trying all possible values of l, and selecting the one that results in the smallest cost. In the end, the flow class that has the minimum cost determines the flow type of F_i.

The optimal segmentation of F is found by a split/merge algorithm starting from the leaf nodes of the oct tree. At each level, eight child nodes are merged into a single node if their cumulative cost is greater than the parent's cost, otherwise they stay split. The split-merge algorithm is applied until the top of the tree is reached, which concludes the segmentation of the gof in terms of the bit rate and the reconstruction error. The basis for the whole gof is called the *block orthonormal bandelet basis*, and it consists of the union of the bases of the $subgof$s in the final segmentation, on their own supports.

IV. RESULTS

In this section, we show the results of the bandelet vs wavelet video compression on some standard sequences. In our experiments, we employed the Daubechies 7-9 filters, using the lifting scheme for both bandelets and wavelets. The bit rates of both compression schemes are computed the same way. In the bandelet decomposition, the smallest $subgof$ in the oct tree is $16 \times 16 \times 8$ ($x \times y \times$ t). The motion parameters are quantized with a step size of 1/8.

The improvement in the compression can be observed when the directions of regularity are not uniform. Our algorithm handles this nonuniformity by automatically segmenting the gof to minimize the compression cost. In Table I we show the compression results for various sequences at multiple bit rates, which are estimated according to [5]. Figure 2 shows the segmentation of a gof from the Flower sequence for $\Delta = 20$. The 1^{st} row shows the segmented frames of a gof from this sequence. There are two motions in the gof due to the parallax effect. The bandelet compression results in the segmentation of the gof until each $subgof$ contains a particular motion. The frames of the subgof marked by the red rectangle are shown in the 2^{nd} row of Fig. 2, with the flow vectors superimposed on

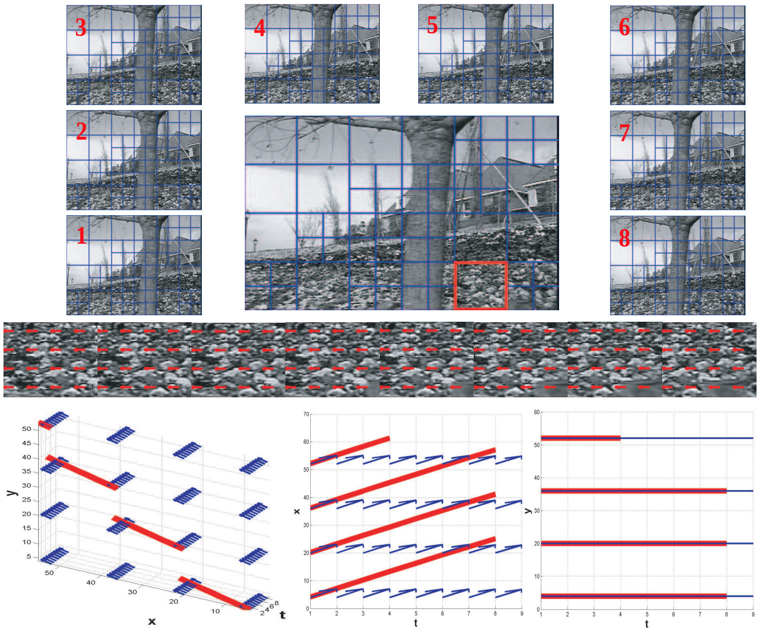

Fig. 2. The results for a gof from the Flower Garden sequence ($\Delta = 20$). (1^{st} Row) The original frames and their segmentation. The middle image is the 3^{rd} frame, zoomed-in for details. Since the tree moves very fast, it results in segmentation of the $subgof$s at the boundaries. ($2^{nd} Row$) The structural flow of the $subgof$ drawn with red boundaries, superimposed on the sub-frames. (3^{rd} Row) The same flow from several (oblique, top and side) views. The direction of the motion has been captured correctly.

Flower	Bit Rate (kbps)	330	250	80	30
	PSNR (Bandelet)	44.80	39.40	27.85	23.72
	PSNR (Wavelet)	7.81	6.77	5.19	5
Alex	Bit Rate (kbps)	220	121	80	75
	PSNR (Bandelet)	45.28	42.50	39.22	32.21
	PSNR (Wavelet)	9.73	8.8	8.74	8.69
Susie	Bit Rate (kbps)	700	510	318	226
	PSNR (Bandelet)	38.7	36.03	33.4	30.86
	PSNR (Wavelet)	26.83	18.98	11.18	9.88
Akiyo	Bit Rate (kbps)	60	70	90	105
	PSNR (Bandelet)	27.55	30.09	33.77	36.35
	PSNR (Wavelet)	10.81	11.5	12.5	14.17

TABLE I

THE BANDELET VS WAVELET COMPRESSION OF VARIOUS SEQUENCES AT MULTIPLE BIT RATES.

them. Close inspection of these frames shows that the motion directions have been captured correctly.

V. CONCLUSION

In this paper, we presented a new framework in video coding that first computes the directions in which a video is regular, then decomposes it along those directions. Our representation, the structural flow field, not only models the directions of regularity due to not only motion, but also the spatial structure of the frames.

As a future direction, we will search for better ways to represent the structural flow in 3D. We also want to explore the possibilities of computing good optical flow using this framework. We believe that the analysis of the flow at different parallelisms hold the key to computing a good optical flow.

REFERENCES

[1] S. Han and C. Podilchuk, “Video compression with dense motion fields,” *IEEE Transactions on Image Processing*, vol. 10, no. 11, November 2001.

[2] A. Secker and D. Taubman, “Highly scalable video compression with scalable motion coding,” *to appear IEEE Transactions on Image Processing*, 2004.

[3] E. Le Pennec and S. Mallat, “Sparse geometric image representation with bandelets,” *to appear in IEEE Transaction on Image Processing*.

[4] I. Daubechies and W. Sweldens, “Factoring wavelet transforms into lifting steps,” in *J. Fourier Anal. Appl.*, 1996, vol. 4, pp. 245–267.

[5] F. Falzon and S. Mallat, “Analysis of low bit rate image transform coding,” *IEEE Transactions on Image Processing*, 1 1998.