

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/2895914>

Motion Feature Detection Using Steerable Flow Fields

Article in *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition* · May 1998

DOI: 10.1109/CVPR.1998.698620 · Source: CiteSeer

CITATIONS

25

READS

32

3 authors, including:



David J. Fleet

University of Toronto

206 PUBLICATIONS 20,162 CITATIONS

[SEE PROFILE](#)



Michael J Black

Max Planck Institute for Intelligent Systems

408 PUBLICATIONS 35,999 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Physics-Based Models of Human Motion [View project](#)



Aerial Outdoor Motion Capture (AirCap): Perception-Based Control [View project](#)

Motion Feature Detection Using Steerable Flow Fields

David J. Fleet*

Michael J. Black†

Allan D. Jepson‡

* Department of Computing and Information Science, Queen's University, Kingston, Ontario K7L 3N6

† Xerox Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304

‡ Department of Computer Science, University of Toronto, Toronto, Ontario M5S 1A4

fleet@qucis.queensu.ca, black@parc.xerox.com, jepson@vis.toronto.edu

Abstract

The estimation and detection of occlusion boundaries and moving bars are important and challenging problems in image sequence analysis. Here, we model such motion features as linear combinations of steerable basis flow fields. These models constrain the interpretation of image motion, and are used in the same way as translational or affine motion models. We estimate the subspace coefficients of the motion feature models directly from spatiotemporal image derivatives using a robust regression method. From the subspace coefficients we detect the presence of a motion feature and solve for the orientation of the feature and the relative velocities of the surfaces. Our method does not require the prior computation of optical flow and recovers accurate estimates of orientation and velocity.

1 Introduction

Constant and affine parameterized models have been used successfully with area-based regression methods for estimating the optical flow arising from smooth surfaces. These methods have been extended to situations involving multiple moving surfaces through the use of robust statistical techniques and layered motion representations. But they do not explicitly model the spatiotemporal structure of *motion features* such as motion discontinuities or moving bars. These features occur frequently in natural scenes due to depth discontinuities, moving objects, and the relative motion of objects such as sign posts, telephone poles, and trees. Such features provide important information about the structural properties of the scene and their modeling and detection remain challenging problems in image sequence analysis.

In this paper we show how to model motion features such as moving edges and bars, and how to detect these features *directly* from spatiotemporal image derivatives. By comparison, most previous approaches attempt to detect motion features (e.g., occlusion) by first computing dense optical flow, or by using indirect energy measures. Alternatively,



Figure 1: Example motion features (in white circles) and explicit models for a motion discontinuity and a moving bar. Both motion features are parameterized by the mean velocity \vec{u}_t , the feature orientation θ , and the change in velocity across the feature $\Delta\vec{u}$.

flow discontinuities have been treated as statistical outliers. One key insight here is that motion features can be modeled in the same way as traditional constant or affine optical flow models, as a linear combination of basis flow fields. The linear coefficients of these models can therefore be estimated using robust area-based regression techniques [3, 4].

This paper addresses three main issues: modeling, estimation, and detection. First, consider the modeling of motion edges and bars like those in Figure 1. A motion edge can be described by a mean (DC) motion vector \vec{u}_t , an orientation θ , and a velocity change across the boundary $\Delta\vec{u}$. Let $\vec{u}(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$ be the corresponding flow field over spatial positions $\vec{x} = (x, y)$ in an image region R . Because $\vec{u}(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$ is non-linear in the parameters of the motion feature, direct estimation of the parameters can be difficult.

Our approach, depicted in Figure 2, is to approximate \vec{u} as a linear combination of a small number of *basis flow fields*

$$\vec{u}(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta) \approx \vec{u}^s(\vec{x}; \vec{c}) = \sum_{j=1}^n c_j \vec{f}_j(\vec{x}), \quad (1)$$

where $\vec{u}^s(\vec{x}; \vec{c})$ is a flow field in the subspace spanned by the basis flow fields $\vec{f}_j(\vec{x})$, and the vector \vec{c} contains the linear coefficients, c_j . While the basis flow fields can be learned from examples as in [4], we take a different approach here

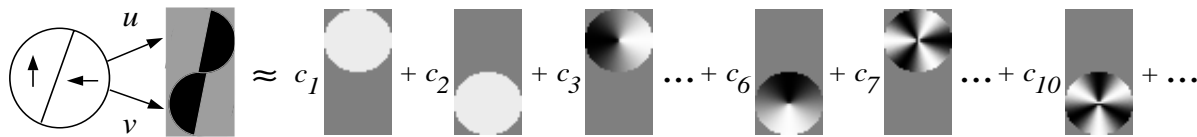


Figure 2: A motion discontinuity can be approximated by a linear combination of basis flow fields within a circular window. Here, the flow fields are shown with the horizontal (u) and vertical (v) components as images, one on top of the other. Black indicates motion to the left or up respectively. Gray is zero motion and white is motion to the right or down. The coefficients, c_j , are scalar weights.

and construct a *steerable* set of basis flow fields. This steerable basis set provides a parameterized model of the motion within an image region.

We estimate the linear coefficients, \vec{c} , that produce a flow field satisfying the brightness constancy assumption

$$I(\vec{x} + \vec{u}^s(\vec{x}; \vec{c}), t - 1) = I(\vec{x}, t), \quad \forall \vec{x} \in R. \quad (2)$$

Equation (2) states that the image, I , at frame $t - 1$ is a warped version of the image at time t . Recent work on optical flow estimation has focused on area-based regression methods for recovering the coefficients of parameterized models for affine motion [1, 3, 7]. These methods recover the model coefficients directly from the spatiotemporal image derivatives, and can be used similarly to recover the coefficients of our motion feature models.

It is important to note that linear models (1) provide an explicit representation of image motion. We do not first recover a general flow field and then project it onto the model basis functions to find the subspace coefficients \vec{c} . Instead, we exploit the linear model directly, estimating the subspace coefficients using the brightness constancy assumption (2).

Finally, given the coefficients, \vec{c} , we need to detect the presence/absence of a motion feature, since not all flow fields in the subspace spanned by the basis flow fields, \vec{f}_j , correspond to actual motion features. We also use the coefficients to compute accurate estimates of the parameters of the motion feature ($\vec{u}_t, \Delta\vec{u}, \theta$).

2 Related Work

Early work on motion boundaries focused on their detection in a dense optical flow field. Authors explored region or edge segmentation techniques [13] and analyzed the distribution of flow vectors in local neighborhoods [14]. These methods are unreliable as they require accurate estimates of optical flow near the motion boundary which are difficult to obtain without an explicit model of motion discontinuities.

Methods that estimate piecewise smooth flow fields using line processes or robust statistics (see [3]) treat motion discontinuities as violations of a spatial smoothness assumption. This is a weak model of motion boundaries that cannot recover more complex features such as moving bars.

Another approach uses filters tuned to the local spatiotemporal image structure of a motion discontinuity [6, 8, 11]. These methods model image structure over time rather than the motion field. For low-level motion-based detection, Black and Anandan [2] look for multiple peaks in a sum-of-squared difference surface as evidence of an occlusion/disocclusion boundary. Although they detect occlusions, these methods do not explicitly model the image motion present at a motion feature.

In [4], the authors propose “learning” explicit linear models of motion features from training examples using principal component analysis (PCA). In contrast to this work, we directly construct a steerable basis set for motion edges and bars. Unlike [4], we are also interested in the *detection* problem; i.e., given the estimated subspace coefficients of the linear feature model, determine whether the coefficients correspond to a motion feature and, if so, recover the feature parameters.

The problem of detecting motion features is, in some respects, similar to the detection of features in static images. Static image features, such as lines, edges, and junctions, are often seen as primitive structures in images upon which later stages of processing, such as object recognition, are based. Remarkably, the primary focus of work on feature detection has been in static images rather than in image sequences. Unlike static image features, *motion features* are typically related directly to scene structure.

Here we construct a set of basis flow fields for motion features that is similar to the construction of steerable filters for edge detection [9, 12]. Our motion feature models are also analogous to the static feature models described by Nayer et al. [10] for edges, bars, and corners. In the case of linear models for static features, the model coefficients are obtained by convolving the image with the basis images. With motion models we cannot take the same approach because the optical flow field is not known. Rather than try to recover the flow field densely and then filter it, one should use the strong constraints imposed by the motion-feature models to improve the accuracy and stability of the motion estimates.

In [10], the detection problem for static features is addressed by taking tens of thousands of example features and

projecting them onto the basis set. These training coefficients provide a dense sampling of a manifold in the subspace spanned by the basis vectors. Given the coefficients corresponding to an image feature, the closest point on the manifold is found. If the point lies sufficiently close to the manifold then the parameters of the nearest training example are taken to be the feature parameters. Alternatively, one could interpolate feature parameters over the manifold [5]. This approach to detection is appropriate for complex features where no underlying model is available (e.g., the motion of human mouths [4]). In our case, the underlying models are relatively simple and we solve for the parameters directly, $(\vec{u}_t, \Delta\vec{u}, \theta)$, given the subspace coefficients, \vec{c} .

3 Linear Bases for Flow Features

In this paper we construct a basis set that is steerable in orientation and velocity, and that provides a reasonable approximation to the motion features of interest, namely, motion edges and motion bars. The resulting basis set is identical to that learned from a sufficiently large set of examples using PCA [4], up to rotations of invariant subspaces. Here we control the structure of the basis flow fields that span invariant subspaces to facilitate detection and interpretation.

To construct basis flow fields for motion edges and bars, we first construct a basis for the spatial structure of the features. These basis images are then combined with a basis for velocity, to produce the basis flow fields.

3.1 Spatial Bases for Edges and Bars

A generic template for the spatial structure of a motion boundary, $S(\vec{x})$, is given by the step edge in Figure 3 (left). This is a mean-zero, unit amplitude step edge within a circular, 32 pixel diameter, window. A circular window is used to avoid orientation anisotropies in the basis set, so that all orientations are treated in the same manner. A steerable basis set for this spatial structure is constructed using the method outlined in [12]. This yields a set of complex-valued basis images, $b_k(\vec{x})$, at specific angular harmonics with wavenumber $k \in H(n)$, where $H(n)$ denotes the set containing the n most significant harmonic wavenumbers in the approximation.

The real and imaginary parts of $b_k(\vec{x})$ form a quadrature pair, and for convenience, we normalize the basis functions so $\|b_k(\vec{x})\| = 1$. The features are then steered (rotated) by taking a linear combination of the basis functions with sinusoidal weights (steering functions). Thus, the edge template, $S(\vec{x})$, and rotated versions of it, $S_\theta(\vec{x})$, are approximated by a linear sum of the basis images as follows:

$$S_\theta(\vec{x}) \approx \Re \left[\sum_{k \in H(n)} \sigma_k a_k(\theta) b_k(\vec{x}) \right], \quad (3)$$

where $\theta \in [0, 2\pi)$ is the rotation angle, $a_k(\theta)$ are the steering functions, and $\Re[z]$ denotes the real-part of z . The

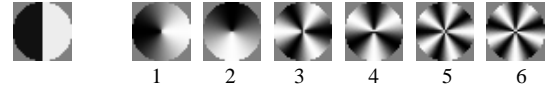


Figure 3: (left) Edge template. (right) Real and imaginary parts of the first 3 complex-valued basis images.

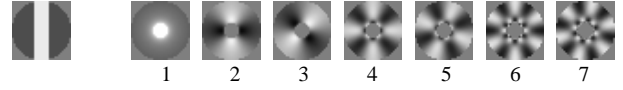


Figure 4: (left) Bar template. (right) Real and imaginary parts of the first 4 complex-valued basis images. The first basis image has wavenumber zero and therefore its imaginary part is zero and not shown.

weights, σ_k , encode the relative magnitudes of the harmonics that are required to approximate the spatial structure of the feature. The set $H(n)$, for a given feature, contains the n wavenumbers with the largest values of $|\sigma_k|$.

The steering functions are angular harmonics, and for a rotation angle of θ , they are given by

$$a_k(\theta) = e^{-ik\theta}. \quad (4)$$

We are only interested in the real part of the expression because the templates are real-valued. One could also rewrite (3) in terms of real-valued basis images and weights as

$$S_\theta(\vec{x}) \approx \sum_{k \in H(n)} \sigma_k (\cos(k\theta) \Re[b_k(\vec{x})] + \sin(k\theta) \Im[b_k(\vec{x})]),$$

where $\Re[b_k(\vec{x})]$ and $\Im[b_k(\vec{x})]$ are the real and imaginary parts of $b_k(\vec{x})$. The quality of the approximation improves with the number of harmonics, n , in the summation.

The basis set for the static edge structure includes a constant (DC) image and a set of images at nonzero angular wavenumbers. Because the edge is odd-symmetric it contains only odd-numbered harmonics. The real and imaginary parts of the first three odd-numbered harmonics, ignoring the DC component, are shown in Figures 3(1–6). These three harmonics account for approximately 94% of the power (variance) in the edge template.

By comparison, the template for the spatial structure of a bar is shown in Figure 4. The template is mean-zero, and the bar has an amplitude of 1. The bar is 8 pixels wide, and the diameter of the circular region R is 32 pixels. The basis set for the bar is composed of even-numbered harmonics. Ignoring the constant basis image, the real and imaginary parts of the basis images for the first four even-numbered harmonics are shown in Figure 4 (1–7). These harmonics account for approximately 90% of the power in the template.

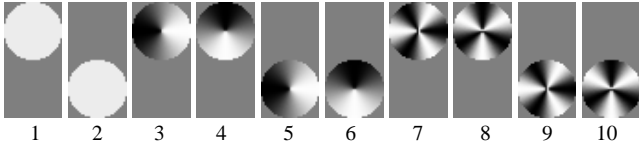


Figure 5: Steerable basis flow fields for occluding edges.

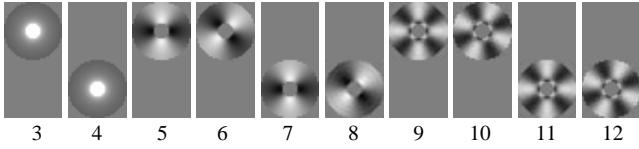


Figure 6: Steerable basis flow fields for motion bars. The DC basis flow fields are equivalent to the first two basis flow fields in the motion edge basis set in Figure 5.

3.2 Basis Flow Fields for Motion Edges and Bars

The basis flow fields for the motion features are formed by combining a basis for velocity with the basis for the spatial structure of the features. Two vectors, $(1, 0)^T$ and $(0, 1)^T$, provide a basis for translational flow. Thus, the basis flow fields for the horizontal and vertical components of the motion features are given by

$$\vec{\mathbf{h}}_k(\vec{\mathbf{x}}) = \begin{pmatrix} b_k(\vec{\mathbf{x}}) \\ 0 \end{pmatrix}, \quad \vec{\mathbf{v}}_k(\vec{\mathbf{x}}) = \begin{pmatrix} 0 \\ b_k(\vec{\mathbf{x}}) \end{pmatrix}.$$

For each angular harmonic, k , there are four real-valued flow fields. These are the real and imaginary parts of $b_k(\vec{\mathbf{x}})$, each multiplied by the horizontal and the vertical components of the velocity basis.

The real and imaginary parts of the basis flow fields for the motion edge are depicted in Figures 5(1-10). Here, there are two angular harmonics, each with 4 real-valued flow fields, along with the DC basis flows that encode constant translational velocity. Similarly, Figures 6(3-12) show the basis flow fields for the motion bar. Figures 6(3-4) encode the basis flow fields for wavenumber $k = 0$, for which only the real-part is nonzero. Figures 6(5-8) and 6(9-12) depict the basis flow fields for wavenumbers $k = 2, 4$ respectively.

3.3 Direct Estimation of Subspace Coefficients

We can now write the linear model (1) in a more useful form that explicitly identifies which linear coefficients are associated with horizontal and vertical basis components at particular wavenumbers:

$$\vec{\mathbf{u}}^s(\vec{\mathbf{x}}; \vec{\mathbf{c}}) = \vec{\mathbf{u}}_{dc} + \Re \left[\sum_{k \in H(n)} \alpha_k \vec{\mathbf{h}}_k(\vec{\mathbf{x}}) + \beta_k \vec{\mathbf{v}}_k(\vec{\mathbf{x}}) \right]. \quad (5)$$

Here, $\vec{\mathbf{u}}_{dc}$ denotes the first two coefficients in $\vec{\mathbf{c}}$ that weight the DC basis functions. Similarly, α_k and β_k are

the complex-valued coefficients from $\vec{\mathbf{c}}$ that weight the complex-valued basis functions that encode horizontal and vertical components of the flow, for wavenumber k .

To recover the subspace coefficients we use the brightness constancy assumption in Equation (2), and minimize the following objective function:

$$E_s(\vec{\mathbf{c}}) = \sum_{\vec{\mathbf{x}} \in R} \rho(I(\vec{\mathbf{x}} + \vec{\mathbf{u}}^s(\vec{\mathbf{x}}; \vec{\mathbf{c}}), t-1) - I(\vec{\mathbf{x}}, t), \sigma). \quad (6)$$

Here, σ is a scale parameter and $\rho(\cdot, \sigma)$ is a robust error function applied to the residual error

$$r(\vec{\mathbf{x}}; \vec{\mathbf{c}}) = I(\vec{\mathbf{x}} + \vec{\mathbf{u}}^s(\vec{\mathbf{x}}; \vec{\mathbf{c}}), t-1) - I(\vec{\mathbf{x}}, t). \quad (7)$$

For the experiments below, $\rho(r, \sigma) = r^2 / (\sigma^2 + r^2)$.

Equation (6) is minimized as described in [4]. Briefly, we replace $\vec{\mathbf{c}}$ in (6) with $\vec{\mathbf{c}} + \delta\vec{\mathbf{c}}$ and linearize the equation with respect to the update vector $\delta\vec{\mathbf{c}}$. We then use iterative coordinate descent to solve for $\delta\vec{\mathbf{c}}$. This involves taking partial derivatives of the linearized objective function with respect to the real and imaginary parts of $\delta\vec{\mathbf{c}}$.

Gaussian pyramids of the input images and the basis flow fields are constructed. The minimization is performed first at a coarse level and then the coefficients are successively updated at each finer level using the motion estimated at the previous level to warp the image at time $t-1$ towards the image at time t .

To deal with the non-convexity of E_s , the robust scale parameter, σ , is initially set to a large value and then slowly reduced during the minimization [3]. The region R is a circular region 32 pixels in diameter. For every pixel in the image (excluding a 16 pixel boundary) we center a region R , and estimate the subspace coefficients using the robust regression method. While this is computationally expensive, the minimization at each pixel could be done in parallel. Currently we do not employ any coherence constraints on the spatial variation of the coefficients.

Translating Disk Example. We constructed a synthetic sequence of a disk translating horizontally at 2 pixels/frame across a stationary background (Figure 7). The estimation of the optical flow, with a motion discontinuity basis (Figure 5), provides a set of coefficients for each estimation neighborhood in the image. We show the coefficient values as images, as a function of spatial position. The real-valued coefficients corresponding to the real-valued basis flow fields in Figure 5(1-10) are shown in Figure 7(1-10).

For example, the real and imaginary parts of α_1 are shown in Figures 7(3-4). Each image depicts the value of a model coefficient (α_1) computed at every pixel, to model the optical flow in the circular region centered at that pixel. Similarly, the real and imaginary parts of β_1 are shown in Figures 7(5-6). The flow vector corresponding to the center pixel in each region is used to generate the dense flow field shown in Figure 7(c).

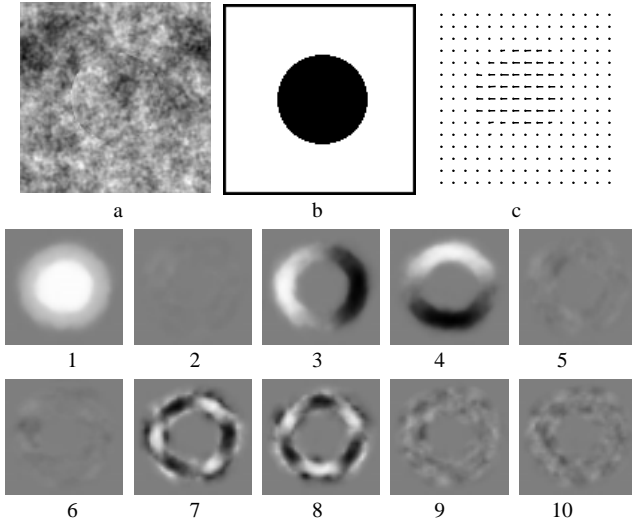


Figure 7: Translating Disk. (a) Image of the disk and background (same random texture). (b) Mask showing location of disk. (c) Recovered flow. (1-10) Recovered coefficient images for the motion edge basis set (scaled differently at each wavenumber to maximize the contrast range).

4 Feature Detection and Estimation

The estimated flow field $\vec{u}^s(\vec{x}; \vec{c})$ for a particular motion feature is, by definition, representable in the subspace spanned by the basis flow fields for that feature. It is not the case, however, that every flow field $\vec{u}^s(\vec{x}; \vec{c})$ corresponds to a valid motion feature. A valid feature, $\vec{u}^m(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$, is the projection of the ideal feature model $\vec{u}(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$ onto the subspace. Here, the ideal features project onto a 5-dimensional nonlinear manifold within the subspace. To detect the presence of a motion feature and to estimate its parameters $(\vec{u}_t, \Delta\vec{u}, \theta)$, we find the flow field on this manifold that is closest to the estimated flow field $\vec{u}^s(\vec{x}; \vec{c})$.

4.1 Nonlinear Least-Squares Estimation

Let $\vec{u}^m(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$ be the projection of the motion feature $\vec{u}(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$ onto the subspace. One can show that these flow fields, which lie on the manifold, have the form

$$\vec{u}^m = \vec{u}_t + \Re \left[\sum_{k \in H(n)} \sigma_k e^{-ik\theta} \left(\Delta u \vec{h}_k(\vec{x}) + \Delta v \vec{v}_k(\vec{x}) \right) \right] \quad (8)$$

The five parameters of the manifold include the two components of the mean translational velocity $\vec{u}_t = (u_t, v_t)$, the two components of the velocity change $\Delta\vec{u} = (\Delta u, \Delta v)^T$, and the feature orientation θ .

Our goal is to determine the feature parameters, $(\vec{u}_t, \Delta\vec{u}, \theta)$, that produce the closest flow field $\vec{u}^m(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$ to the estimated flow field $\vec{u}^s(\vec{x}; \vec{c})$ in the

subspace, and to decide whether the model is a sufficiently good fit to allow us to infer the presence of the feature. For a region R , we seek the parameters that minimize

$$\sum_{\vec{x} \in R} \|\vec{u}^m(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta) - \vec{u}^s(\vec{x}; \vec{c})\|^2 \quad (9)$$

Because \vec{u}^s and \vec{u}^m are represented using the same orthogonal basis it suffices to find the parameters that minimize the sum of squared differences between model coefficients and estimated coefficients \vec{c} . Thus, the translational velocity \vec{u}_t is given directly by the linear coefficients that correspond to the DC basis flow fields, namely, \vec{u}_{dc} . The remaining parameters, $\Delta\vec{u}$ and θ , are found by minimizing

$$E_m(\Delta\vec{u}, \theta) = \sum_{k \in H(n)} \|(\alpha_k, \beta_k) - \sigma_k e^{-ik\theta} (\Delta u, \Delta v)\|^2 \quad (10)$$

where α_k and β_k are the known estimated coefficients. This is a nonlinear least-squares problem that can be solved iteratively with a sufficiently good initial guess.

4.2 Suboptimal Direct Estimation

The least-squares minimization enforces two constraints on the motion feature parameters. First, the velocity structure of the motion feature, $\Delta\vec{u}$, must be the same for all angular harmonics. Second, the orientation of the motion feature, θ , in the phase angles of the model coefficients, must be consistent across all of the angular harmonics and both components of flow. The constraint on $\Delta\vec{u}$ is related to the magnitudes of the complex-valued model coefficients, while the constraint on θ concerns their phase values. To obtain an initial guess for minimizing $E_m(\Delta\vec{u}, \theta)$, we first decouple these constraints. This provides a suboptimal, yet direct, method for estimating θ and $\Delta\vec{u}$.

Direct Estimation of Velocity Change. To formulate a constraint on $\Delta\vec{u}$ it is convenient to collect the complex-valued coefficients of the model flow in (8) into a matrix

$$M = \begin{pmatrix} \Delta u \sigma_{k_1} e^{-ik_1\theta} & \dots & \Delta u \sigma_{k_n} e^{-ik_n\theta} \\ \Delta v \sigma_{k_1} e^{-ik_1\theta} & \dots & \Delta v \sigma_{k_n} e^{-ik_n\theta} \end{pmatrix}, \quad (11)$$

where k_j is the j^{th} wavenumber in $H(n)$. The top row of M contains the model coefficients for the horizontal component of the model velocity field, $\vec{u}^m(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$. The second row of M contains the model coefficients for the vertical component of $\vec{u}^m(\vec{x}; \vec{u}_t, \Delta\vec{u}, \theta)$.

To obtain a set of transformed coefficients that depend solely on $\Delta\vec{u}$, we construct $A = M M^{*T}$, where M^{*T} is the conjugate transpose of M . To understand the form of A , we can rewrite M as an outer product

$$M = \Delta\vec{u} \left(\sigma_{k_1} e^{-ik_1\theta}, \dots, \sigma_{k_n} e^{-ik_n\theta} \right). \quad (12)$$

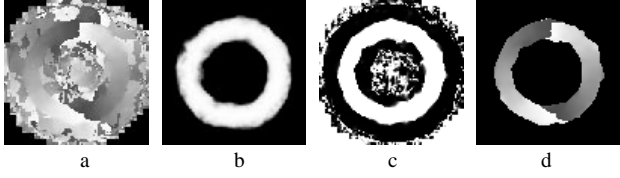


Figure 8: Translating Disk. (a) Raw orientation estimates from direct method. Black pixels indicate that no estimate was possible. Intensity varies from white to dark grey as orientation goes from $\pi/2$ to $-\pi/2$. (b,c) Confidence measures from velocity constraint and orientation constraint. (d) orientation estimates masked by joint confidence measure.

This explicitly shows the separability of the model coefficients with respect to $\Delta\vec{u}$ and θ . From this it is also straightforward to show that A given above is equivalent to $(\sigma_{k_1}^2 + \dots + \sigma_{k_n}^2) \Delta\vec{u} \Delta\vec{u}^T$, the components of which are independent of θ .

For example, in the case of the motion edge, with the estimated coefficients $\alpha_1, \alpha_3, \beta_1$ and β_3 , let

$$\tilde{M} = \begin{pmatrix} \alpha_1 & \alpha_3 \\ \beta_1 & \beta_3 \end{pmatrix}, \quad \tilde{A} = \tilde{M} \tilde{M}^{*T}. \quad (13)$$

If the estimated flow field, $\vec{u}^s(\vec{x}; \vec{c})$, were on the manifold defined by the feature model, then the singular vector associated with the largest singular value, e_1 , of \tilde{A} should give the direction of the velocity. Thus, an estimate of the velocity change, $\Delta\vec{u}_e$, is obtained by scaling this singular vector by $\sqrt{e_1/(\sigma_1^2 + \sigma_3^2)}$.

Moreover, if the estimated coefficients lie on the manifold, then the rank of \tilde{A} in (13) should be 1. As a consequence, we use the ratio of the singular values, $e_1 \geq e_2$, to determine the quality of the model fit. A measure of the consistency of the estimated coefficients with the model is therefore given by $r = (e_2 + a)/e_1$. Here, r is close to zero when the coefficients satisfy the constraint and large otherwise; a is a small constant that helps to avoid instability when both singular values are extremely small. We use $C_v = \exp[-100r^2]$ as a confidence measure derived from this constraint, an image of which is shown for the disk sequence in Figure 8(b). The values of C_v approach 1 as the rank of \tilde{A} tends to 1.

Direct Estimation of Spatial Orientation. Once the velocity parameters have been estimated, we can use the estimate, $\Delta\vec{u}_e$, along with the matrix \tilde{M} to obtain a set of transformed measurements that, according to the model, depend only on the spatial structure of the motion feature. In particular, given the true velocity change $\Delta\vec{u}$, it is easy to show from (12) that the product $\Delta\vec{u}^T M$ has the form

$$\Delta\vec{u}^T M = |\Delta\vec{u}|^2 (\sigma_{k_1} e^{-i k_1 \theta}, \dots, \sigma_{k_n} e^{-i k_n \theta}). \quad (14)$$

From this it is clear that the orientation θ is available in the phase values of the elements of $\Delta\vec{u}^T M$.

To obtain an orientation estimate θ_e , we form the product $\vec{z} = \Delta\vec{u}_e^T \tilde{M}$, using the estimated velocity change $\Delta\vec{u}_e^T$ and the matrix of estimated coefficients \tilde{M} . To obtain θ_e from \vec{z} we unwrap the phase of each component of \vec{z} according to its corresponding wavenumber, as in (14), and take their average. Figure 8(a) shows the average of the two unwrapped phases for the Disk Sequence. In practice one might use a weighted average to deal with greater noise in higher-order harmonics.

The variance of the unwrapped phases also provides us with another measure of whether the estimated coefficients satisfy the model. In the case of the edge model, where only two harmonics are used, the phase of the third harmonic coefficient should be three times the phase of the first; i.e., $\Delta\phi = \phi_1 - \phi_3/3 = 0$ where ϕ_k is the phase of the component of \vec{z} at wavenumber k . As a simple confidence measure for the quality of the model fit, Figure 8(c) shows $C_\theta = \exp(-\Delta\phi^2/500)$ for the Disk Sequence coefficients. Here, C_θ approaches 1 where the model is a good fit, or when the coefficients are all zero.

4.3 Experiments: Least-Squares Results

The direct method provides estimates of orientation θ_e and the velocity change $\Delta\vec{u}_e$, along with two confidence measures, C_v and C_θ . We find that these estimates of θ and $\Delta\vec{u}$ are usually close to the least-squares estimates. The confidence measures can be used to reject pixels that are clearly far from the feature manifold. For example, Figure 8(d) shows orientation estimates, θ_e , at pixels where $C_v C_\theta > 0.1$.

To obtain optimal estimates of the motion feature parameters we use a gradient descent procedure to find values of θ and $\Delta\vec{u}$ that minimize $E_m(\Delta\vec{u}, \theta)$ in (10), given the initial guess from the direct method. Detection is based on the squared error in the optimal fit divided by the power in the estimated subspace coefficients, i.e., $P = \sum_k (|\alpha_k|^2 + |\beta_k|^2)$. Note that this sum does not include the coefficients for the translational basis flow fields. Then, our confidence measure is given by

$$C = c(P) e^{-E_m/P}. \quad (15)$$

We find that the reliability of the detection improves as P increases. To exploit this, $c(P)$ is a function that tends towards 1 as P increases; for motion edges $c(P) = e^{-40/P}$, and for motion bars $c(P) = e^{-50/P}$. The larger constant in the case of motion bars reflects the greater amount of noise we observe in the estimation of the bar model coefficients. As P decreases, the relative error E/P must decrease for C to remain constant.

Figures 9(a-c) show optimal estimates of θ , Δu and Δv . Notice that the sign of the velocity difference is clearly

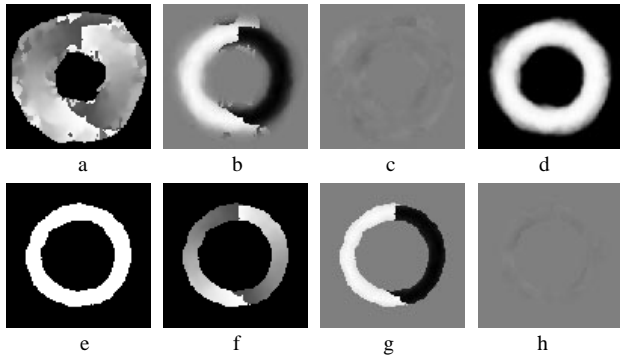


Figure 9: Translating Disk. (a-c) Optimal estimates of θ , Δu , Δv . (d) Confidence measure C , derived from squared error of least-squares fit. (e) Locations where $C > 0.8$ are white. (f-h) Estimates of θ , Δu , Δv where $C > 0.8$.

visible in Figure 9(b) as the disk is moving to the right at 2 pixels/frame. Figure 9(d) shows the confidence measure C , while Figure 9(e) shows locations at which $C > 0.8$. The remaining images in Figure 9 show the least-squares estimates of θ and $\Delta \vec{u}$ at locations where $C > 0.8$. For these pixels, the mean error in θ was 0.12° with a standard deviation of 5.2° (RMS error: 5.2°). The mean error in Δu was -0.21 pixels/frame with a standard deviation of 0.16 (RMS error: 0.27 pixels/frame). Errors in Δv were insignificant by comparison.

To test feature detection and estimation with the motion bar we constructed a synthetic sequence of an annulus (width of 8 pixels) translating across a stationary background to the right at 2 pixels/frame (see Figure 10(a-c)). Figure 10(d) shows C at the least-squares minimum. The remaining images, Figures 10(e-g) show the optimal estimates of θ and $\Delta \vec{u}$ at pixels where $C > 0.7$. The fits with the motion bar model are noisier than those with the edge model, and we therefore use a more liberal threshold to display the results. For these pixels, the mean error in θ was 0.9° with a standard deviation of 9.1° (RMS error: 9.1°). The mean error in Δu was -0.26 pixels/frame with a standard deviation of 0.21 (RMS error: 0.34 pixels/frame).

Note that in these two experiments, although the models for the edge and the bar are straight, we are testing them here with curved edges and bars. This, in part, accounts for the errors we report, but also shows how the models generalize to a relatively wide class of similar feature instances.

Finally, Figures 11 and 12 show motion edges and bars detected in two outdoor sequences. The velocity differences at the tree boundary in the flower garden sequence are as large as 7 pixels/frame. The sign of the velocity change in Figure 11(e) clearly shows the occluding and disoccluding sides of the tree. The orientation estimates along the tree are vertical, as indicated by the grey pixels in 11(d).

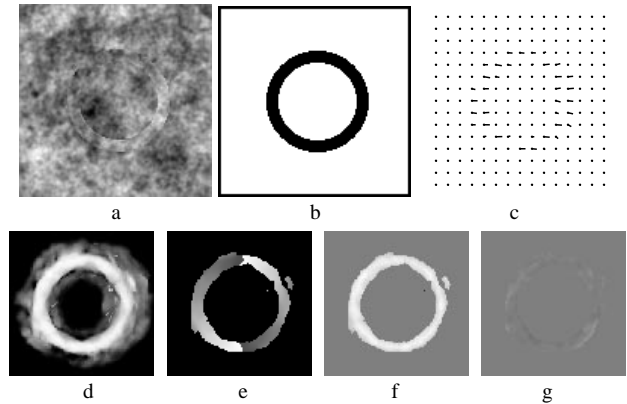


Figure 10: Translating Annulus. (a) Frame 1 of sequence. (b) Mask showing location of annulus. (c) Recovered flow. (d) Confidence measure C from least-squares minimization. (e-g) Optimal estimates of θ , Δu and Δv where $C > 0.7$.

The results on the handrail sequence are noisier, but where the change in velocity between the rails and the hill behind them is sufficiently large, the rails are detected well. The rails are moving upwards faster than the background, the motion of which is visible in Figure 12(e,f). Unlike results in previous figures, orientation in Figure 12(d) is represented between 0 and 2π , so that horizontal is shown in grey.

5 Discussion

We have shown how motion features such as occlusion boundaries and moving bars can be modeled as a linear combination of basis flow fields. Our approach extends the notion of a steerable basis set from static images to motion data. Moreover, we show how the coefficients of the feature models can be recovered directly from the image in the same fashion as affine motion coefficients are computed. Finally, we have shown how to reliably detect the presence of a motion feature from the linear coefficients and how to recover the feature orientation and the relative velocities of the surfaces. This work extends regression-based optical flow methods to cope with more complex features and helps bring to light the relationships between static image features and motion features.

In current work we are combining moving edges and bars into a single basis set and are extending the detection method to determine which features are present. We also plan to explore how variations in scale affect detection and how to extend our model to cope with features at multiple scales. Currently, the coefficients of each image region are estimated independently and it would be interesting to explore the regularization of neighboring coefficients to reduce noise and enforce continuity along contours. We could also classify motion edges as occluding or disoccluding, but

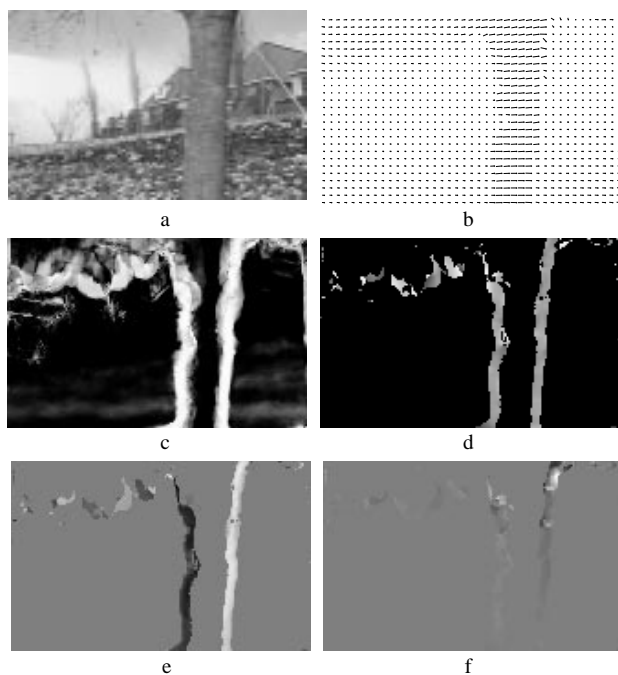


Figure 11: Flower Garden Sequence. (a) Frame 1 of sequence. (b) Recovered flow. (c) Confidence measure C . (d-f) Optimal estimates of θ , Δu and Δv where $C > 0.8$.

with two frames we cannot determine the relative depth order of the surfaces. The method, however, could be applied to stereo data where the relative depth ordering could be determined. Finally, we have not attempted to accurately locate the position of an edge or bar; this should be relatively straightforward.

References

- [1] J. R. Bergen, P. Anandan, K. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. *Proc. ECCV*, pp. 237–252, 1992.
- [2] M. J. Black and P. Anandan. Constraints for the early detection of discontinuity from motion. *Proc. AAAI*, pp. 1060–1066, 1990.
- [3] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *CVIU*, 63:75–104, 1996.
- [4] M. J. Black, Y. Yacoob, A. D. Jepson, and D. J. Fleet. Learning parameterized models of image motion. *Proc. CVPR*, pp. 561–567, 1997.
- [5] C. Bregler and S. M. Omohundro. Nonlinear manifold learning for visual speech recognition. *Proc. ICCV*, pp. 494–499, 1995.

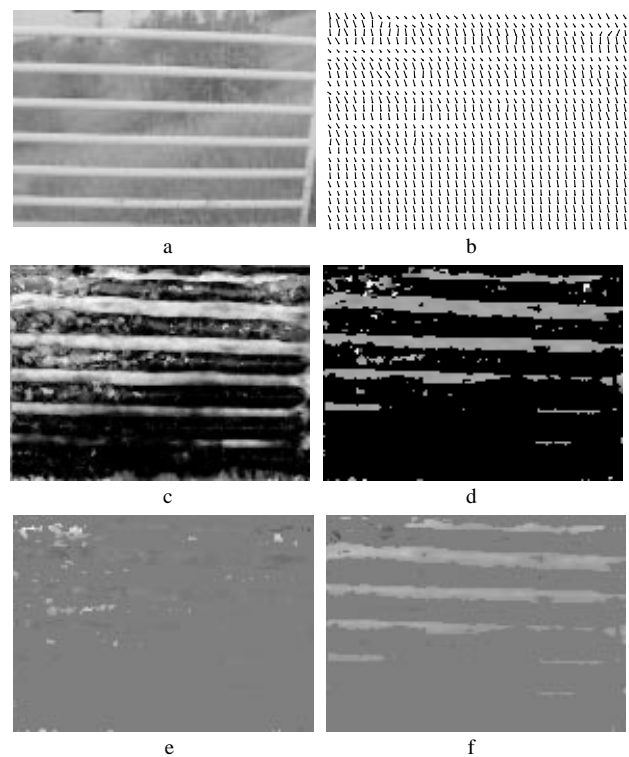


Figure 12: Handrail Sequence. (a) Frame 1 of sequence. (b) Recovered flow. (c) Confidence measure C . (d-f) Optimal estimates of θ , Δu and Δv where $C > 0.7$.

- [6] G. T. Chou. A model of figure-ground segregation from kinetic occlusion. *Proc. ICCV*, pp. 1050–1057, 1995.
- [7] D. J. Fleet and A. D. Jepson. Computation of component image velocity from local phase information. *IJCV*, 5:77–104, 1990.
- [8] D. J. Fleet and K. Langley. Computational analysis of non-fourier motion. *Vision Research*, 22:3057–3079, 1994.
- [9] W. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. PAMI*, 13:891–906, 1991.
- [10] S. K. Nayar, S. Baker, and H. Murase. Parametric feature detection. *Proc. CVPR*, pp. 471–477, 1996.
- [11] S. A. Niyogi. Detecting kinetic occlusion. *Proc. ICCV*, pp. 1044–1049, 1995.
- [12] P. Perona. Deformable kernels for early vision. *IEEE Trans. PAMI*, 17:488–499, 1995.
- [13] J. L. Potter. Scene segmentation using motion information. *IEEE Trans. SMC*, 5:390–394, 1980.
- [14] A. Spoerri and S. Ullman. The early detection of motion boundaries. *Proc. ICCV*, pp. 209–218, 1987.