

# Local Grouping for Optical Flow

Xiaofeng Ren  
Toyota Technological Institute at Chicago  
1427 E. 60th Street, Chicago, IL 60637  
xren@tti-c.org

## Abstract

Optical flow estimation requires spatial integration, which essentially poses a grouping question: what points belong to the same motion and what do not. Classical local approaches to optical flow, such as Lucas-Kanade, use isotropic neighborhoods and have considerable difficulty near motion boundaries. In this work we utilize image-based grouping to facilitate spatial- and scale-adaptive integration. We define soft spatial support using pairwise affinities computed through intervening contour. We sample images at edges and corners, and iteratively estimate affine motion at sample points. Figure-ground organization further improves grouping and flow estimation near boundaries. We show that affinity-based spatial integration enables reliable flow estimation and avoids erroneous motion propagation from and/or across object boundaries. We demonstrate our approach on the Middlebury flow dataset.

## 1. Introduction

The computation of optical flow requires spatial integration, because local signals are noisy and suffer from the well-known *aperture problem* [16]. How to integrate motion spatially, referred to as the *generalized aperture problem* [17], is highly non-trivial: it essentially comes to a grouping question, what pixels belong to a single motion and what do not.

Classical local approaches to optical flow, such as Lucas-Kanade [22], ignores grouping and uses isotropic neighborhoods. They face a scale dilemma: a small neighborhood may not contain enough information, and a large one may straddle across object/motion boundaries.

To accommodate grouping, many approaches compute segmentations and use resulting segments as spatial support (e.g. [9, 43, 35]). Segmentation, however, is never perfect, and making a hard decision at an early stage could be dangerous. These approaches also face a scale dilemma: small segments again may not contain enough motion, and seg-

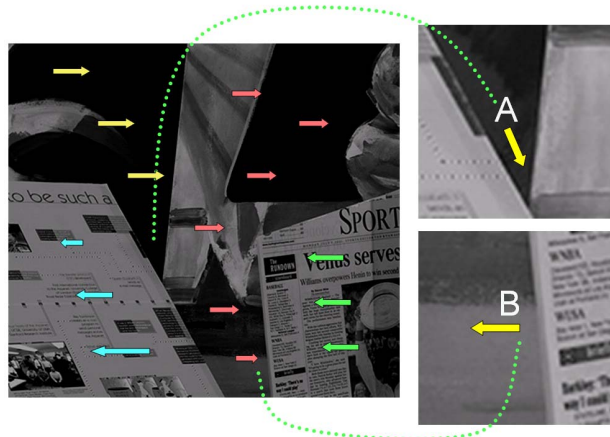


Figure 1. Left: the Venus image from the Middlebury flow dataset, consisting of four motion layers. Right: examples of erroneous spatial integration. Flow estimated at Point A is typically downward, because two boundaries around Point A move in opposite directions and the T-junction “moves” down. Motion cues around Point B are weak, and the strong motion boundary nearby, which actually “belongs” to the right-hand side, causes its flow estimation to go leftward. We solve these issues by applying image-based grouping and figure-ground organization.

mentation at a large scale is unreliable.

In this work, we develop a grouping-based approach to optical flow without making hard decisions. We apply the *Probability-of-Boundary* operator [24] to compute a boundary probability/contrast map. We use an asymmetric *intervening contour* [23] scheme on the boundary map to define pairwise affinities between locations. These affinities define a soft spatial support independently for each location. We use affinity values as weighting to estimate an affine motion at each location.

To facilitate computation, we sample images at corners and edges, estimate flow at sparse samples only, and then interpolate them back to dense flow. Affinity-based soft grouping defines spatial- and scale-adaptive support for motion integration and enables accurate flow estimation under challenging circumstances (such as the examples in Fig-

ure 1). We show that we significantly improve state-of-the-art results on the Middlebury flow dataset [5].

## 2. Related Work

There are in general two perspectives to flow integration, i.e. *local* and *global*. Classical local approaches, such as Lucas-Kanade [22], use rectangular or Gaussian windows. Windows are adaptive in scale in [18] but remain isotropic. Pyramid-based Lucas-Kanade is capable of extending motion estimates from corners to edges and region interior, and is still among the most popular flow and feature tracking techniques today [4, 5].

A global approach to optical flow, such as Horn and Schunck [16], jointly optimizes a flow field using both brightness constancy and flow smoothness. Black and Anandan [10] introduces a robust estimation framework, allowing the recovery of piecewise smooth flow fields and motion discontinuities.

The CLG algorithm [12] combines local and global approaches by smoothing motion signals before feeding into a global scheme. Local smoothing offers robustness as in Lucas-Kanade. This local integration is nevertheless still isotropic, and has difficulty with the challenging examples in Figure 1 (see [5]). More recent extensions have explored occlusion detection [1, 41], real-time performance [11], or learning spatial priors [29].

A different approach to motion analysis is to group or segment based on motion, explaining it in layers. Mixture models have been popular, using affine motion for planar surfaces [38] or smoothness inside layers [39], and a number of methods have been proposed to estimate model order [3, 40]. Recent approaches have explored the combination of motion and color [19, 43]. The theme of these approaches is mostly on segmentation and semantic interpretation, not on accurate flow estimation.

Several lines of work have been developed to accommodate image-based grouping into optical flow. From a local perspective, many algorithms segment images into regions (using either brightness [9], color [14] or boundary contrast [35]) and compute a single motion for each region. The quality of segmentation is critical for these approaches.

From a global perspective, *anisotropic diffusion* uses grouping information in a way by encouraging flow discontinuities at high-gradient locations [25, 2]. Gradients, especially computed at a small scale, may not correspond well with object boundaries [24]. Empirically, there has not been enough evidence to support this anisotropy [6].

Another related line of work that uses grouping is *bilateral filtering* [36], where smoothing is based on both spatial proximity and brightness/color similarity. It has been applied as a post-processing step to correct errors near motion boundaries [41]. Such post-processing would have difficulty if initial flow estimates are qualitatively wrong.

Image grouping or segmentation is a vast field of its own and has seen a lot of progress in recent years (e.g. [23, 37]). Many discriminative approaches to grouping are based on the concept of pairwise similarity or *affinity* between points [31, 13]. Recent works define pairwise affinity through *intervening contour* [20, 23, 42], which measures boundary energy along straight line paths between points. State-of-the-art boundary detection combines brightness, color and texture contrasts [24].

## 3. Grouping for Optical Flow

Spatial integration for optical flow is essentially a question of grouping: points that belong to a single moving object (or object part) should share their motion information for robustness, and points that belong to different objects should have no influence on one another. One may define grouping as pairwise affinities [31]: for every pair of points in the image, an affinity value indicates how likely they are grouped together.

Let us consider a local point of view for motion integration, where we want to estimate motion/flow at one particular point **A**. Classical Lucas-Kanade uses an isotropic neighborhood around **A**. This is a very conservative view of grouping, using proximity only: the affinity of **A** to another point is 1 if it is with a small distance threshold, 0 otherwise. On the other hand, a segmentation-based approach (e.g. [9]) takes a very aggressive view of grouping, computing a hard segmentation, and the affinity of **A** to another point is 1 if they belong to the same segment, 0 otherwise.

In this work we show that such a hard decision is not necessary. We estimate a (soft) pairwise affinity  $g(A, B)$  from **A** to another point **B**, and use this affinity as a weighting of how much information we should include from **B** when computing the flow at **A**.

Figure 2 shows a summary of our approach. We start by computing boundary probability/contrast. The *Probability-of-Boundary* operator [24] (Pb) combines local brightness, color and texture contrasts and has been shown to outperform gradient-based approaches at boundary detection. Texture analysis is especially useful for optical flow: unlike traditional anisotropic diffusion, it would allow spatial integration across small-scale textures.

The boundary contrast map, as shown in Figure 2(b), enables us to define affinity between any pair of points through *intervening contour* [23]: if two points are separated by strong boundaries, they should be weakly associated; otherwise, we are probably in a uniform region and the two points should be strongly associated.

### 3.1. Sampling corners and edges

We have defined a soft grouping using pairwise affinity between points through boundary estimation. It would be

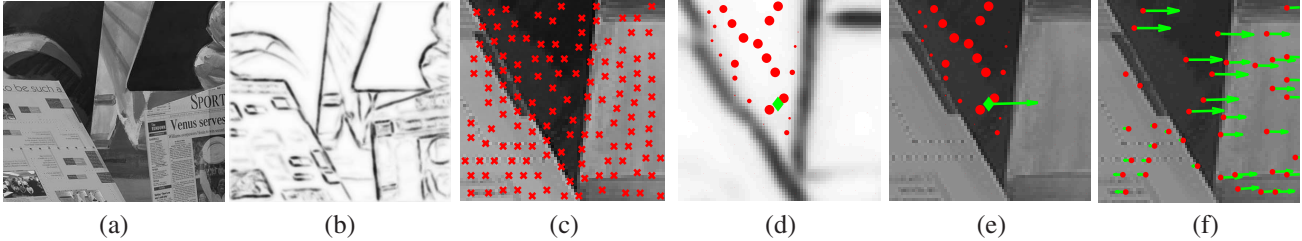


Figure 2. A summary of our grouping-based optical flow approach: for a pair of images (a), we apply the *Probability-of-Boundary* operator [24] to compute a soft boundary map (b). The boundary map enables us to define pairwise affinities through *intervening contour* [23]. To compute all-pair affinities would be costly and unnecessary; instead we sample the image at corners and edges (c). To estimate flow at a sample point, e.g. the green-diamond location in (d), we estimate its grouping connections toward all other points. In (d) we visualize (some) affinities using disk size, with large disks plotted at strongly connected points. These affinities define a (soft) support for spatial integration of flow, avoiding connections across object boundaries. We estimate an affine motion using these affinities, obtaining a correct flow estimate (e). Flow estimates for other points in the area are shown in (f).

too costly to compute, or just to store, affinities between all pairs of pixels; it would also be unnecessary to use all pixels, as motion varies slowly except at object boundaries. We facilitate the computation by sampling an image at corners and edges, working with sparse samples only, and projecting the flow to the pixel-grid at the end.

A classical sparse flow approach, such as Lucas-Kanade, focuses on corners only, where locally there is enough information to determine the flow. Let  $\lambda_1 > \lambda_2$  be the two eigenvalues of the second moment matrix. To select “good features” or corners, one typically uses  $\lambda_2$ , the smaller eigenvalue [33]. This criterion largely ignores edges.

There is of course a lot of motion information at a strong edge: flow in the gradient direction is highly constrained. For our purpose, where we want to extract all motion information and to faithfully represent motion, edges are almost as important as corners. Therefore we use a modified criterion of “goodness”, i.e. the product

$$\lambda_1 \cdot (\lambda_2 + \epsilon_s)$$

where  $\epsilon_s$  is a small constant. This criterion still prefers corners, but will also include strong edges, where  $\lambda_1$  is large and  $\lambda_2$  is small. We sample sequentially with this criterion, enforcing a minimum distance between samples. An example of the sampling is shown in Figure 3. We choose a low threshold so as to cover most of the image. The sampling is almost uniform but it respects corners and edges.

A related sampling strategy is adopted in [30], but with a different focus: there they obtain motion on the pixel-grid first and then use it to track sample points through frames; while we use sample points to estimate the motion itself.

### 3.2. Pairwise affinity using intervening contour

*Intervening contour*, introduced in [20], computes pairwise affinities from a boundary map by connecting two points with a straight line path and measuring boundary energy or “resistance” along the path.

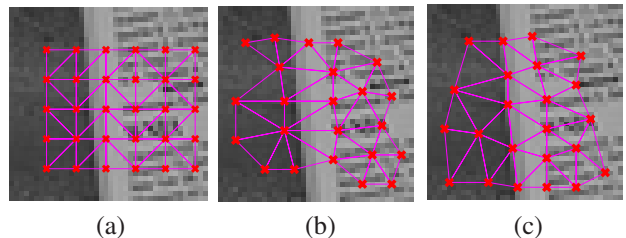


Figure 3. We want a spatial sampling that captures most motion information and faithfully represents underlying image structure and motion. (a) Uniform sampling is unaware of image structures. (b) Classical feature selection criterion favors corners and ignores edges. (c) Our sampling strategy uses both eigenvalues of the second moment matrix and respects corners and edges.

Let  $E(t)$  be the Pb boundary contrast profile along the straight line path between two points  $(i, j)$ . A typical *intervening contour* scheme computes the maximum  $E^* = \max_t E(t)$  as a measure of resistance. One problem with taking the maximum is that it ignores how many boundaries the path crosses. After all, going across a weak boundary is not the same as having to go across four of them. This suggests using a sum  $R = \int |E'|$ .

Under the sum rule, a boundary point usually has a high resistance (hence low affinity) going anywhere, because  $E$  at the starting point is high. To correct this bias against boundary points, we modify the scheme to use “upward” gradient, counting resistance only when climbing up a boundary:  $R = \int \max(E', 0)$ .

Finally, if we travel along a boundary, the contrast will not be constant and we might encounter a lot of gradients. On the other hand,  $E(t)$  is always high on the path. To allow strong connections between points on the same boundary, we balance the gradient  $\max(E', 0)$  with  $\min E$ .

Combining the observations above, we modify the *intervening contour* rule as follows:

$$R = \int \max(E'(t), 0) dt - \min E(t)$$

and compute affinity as  $g = \exp(-c_R R)$ . An example of this affinity is shown in Figure 2(d). We set  $c_R = 10$  and cap affinity values at 1.1 ( $R$  may be negative due to the second term, leading to strong connections along boundaries).

Our definition of intervening contour is asymmetric at boundaries, as illustrated in Figure 4: if **A** is an interior point and **C** is a boundary point nearby, **A** is weakly connected to **C**, but **C** is strongly connected to **A**.

Such behavior is in fact desirable for motion estimation. Because of the aperture problem, points on a boundary alone are not enough to estimate flow, and they need to connect into nearby regions. On the other hand, as we have seen in Figure 1 B, connecting onto a boundary without knowing its figure-ground assignment may lead to large errors. Hence we choose to suppress connections onto boundaries. This conservative strategy downplays the role of contour-based motion analysis (e.g. [21]), a challenging and unsolved problem that needs to be addressed in the future.

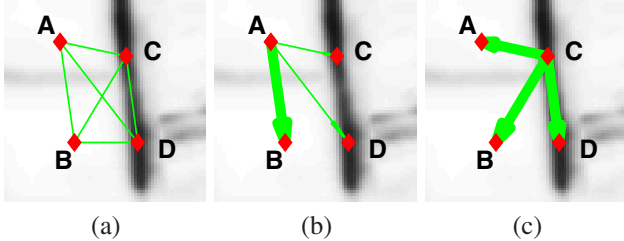


Figure 4. Asymmetric intervening contour: *intervening contour* defines pairwise affinity by measuring how much “resistance” (i.e. boundary energy) one encounters when traveling from one point to another through straight line (a). We measure this resistance using “upward” gradient only, leading to asymmetric affinities: (b) Point **A** is strongly connected with **B** but weakly with **C** and **D**, having to climb onto a strong boundary; (c) Point **C** is strongly connected with all **A**, **B** and **D**, because traveling “downward” on the boundary incurs no cost.

## 4. Semi-Local Optical Flow

In the previous section we have collected a set of sample points  $\{x_i, y_i\}$  at corners and edges, and have defined pairwise affinities  $g(\cdot, \cdot)$  through *intervening contour*. For each point  $i$ , the affinity  $g(i, \cdot)$  defines a soft support for flow integration at  $i$ .

Because the affinity-based support is much larger than the typical scales used in local optical flow methods, we view our approach as *semi-local*. It is spatial-adaptive as *intervening contour* suppresses connections through boundaries. It is also scale-adaptive: the support expands further in a uniform region than in a region of strong contrasts. For a large spatial support, we can no longer assume that motion inside is approximately constant. Instead we will estimate an affine motion, at each point  $i$ , using the affinity  $g(i, \cdot)$  as weighting.

### 4.1. Affinity-based affine flow

Let  $I(x, y, t)$  represent a sequence of images and let  $\mathbf{v} = (v_x, v_y)$  be the flow/motion. The brightness constancy assumption and a differential approximation lead to the classical *optical flow constraint*:

$$I_x v_x + I_y v_y + I_t = 0 \quad (1)$$

The optical flow constraint is defined at a point. Because we are working with sparse samples, every sample point represents a small neighborhood around it. If the minimum distance in sampling is  $d$ , every point has a “zone-of-control” of radius  $d/2$ . As in Lucas-Kanade, we assume that motion is constant in this neighborhood. If we put together the spatial and temporal gradients in vectors  $\mathbf{I}_x$ ,  $\mathbf{I}_y$  and  $\mathbf{I}_t$ , locally the flow needs to satisfy the over-constrained equation  $[\mathbf{I}_x \mathbf{I}_y] \mathbf{v} = -\mathbf{I}_t$ . The least squares solution to the problem is a  $2 \times 2$  system

$$\mathbf{A} \mathbf{v} = \mathbf{b}$$

where  $\mathbf{A} = [\mathbf{I}_x \mathbf{I}_y]^T [\mathbf{I}_x \mathbf{I}_y]$  and  $\mathbf{b} = [\mathbf{I}_x \mathbf{I}_y]^T \mathbf{I}_t$ . This  $2 \times 2$  system summarizes motion cues around each sample point.

The entries in  $\mathbf{A}$  and  $\mathbf{b}$  scale quadratically with image brightness  $I$ . As pointed out in [34], the certainty of local flow scales differently with brightness: it increases first when image contrasts arise above noise level, but quickly saturates. To capture this intuition, we normalize  $\mathbf{A}$  and  $\mathbf{b}$  using  $1/(A_{11} + A_{22} + \epsilon_A)$ .

To estimate the flow at a certain point  $i$ , we assume that the semi-local motion based at  $i$  is affine, and its prediction at a nearby point  $j$  would be  $\mathbf{v}^j = (\alpha_1 x + \alpha_2 y + \alpha_3, \alpha_4 x + \alpha_5 y + \alpha_6)^T$ . We find the best fitting affine motion  $\vec{\alpha}^*$  by minimizing the following weighted error:

$$\sum_j g(i, j) \|\mathbf{A}^j \mathbf{v}^j(\vec{\alpha}) - \mathbf{b}^j\|^2 \quad (2)$$

The flow estimation at  $i$  is then  $\mathbf{v}^i(\vec{\alpha}^*)$ . In practice, we threshold the affinity  $g(i, j)$ , and only include points that have strong connections to  $i$ .

To increase the robustness of the affine estimation, We use two extensions: (1) a robust error norm; and (2) a regularization on the affine parameters. For the robust norm, We use the Lorentzian function as in [10], and solve with iterative least squares. For affine regularization, we put a small constant cost on the four linear parameters in the affine model, hence favoring constant motion over large affine variations.

### 4.2. Iterative and coarse-to-fine estimation

Because our method is based on the linearized optical flow constraint (Eq. 1), conventional wisdoms on differential flow apply: iterative/incremental estimation is needed for accurate flow recovery, and coarse-to-fine search is needed to handle large motions.

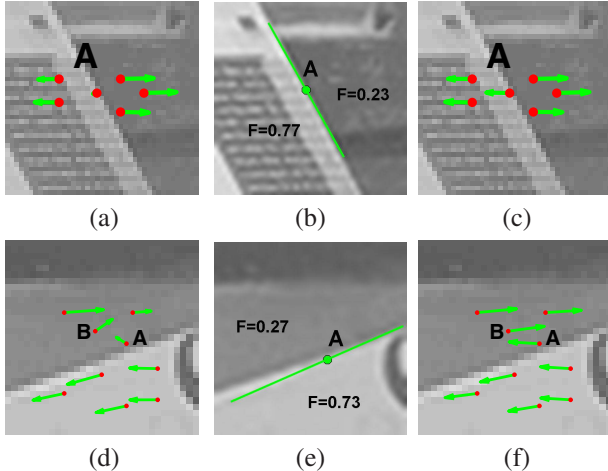


Figure 5. Examples of figure-ground assignment for optical flow: (a) A priori, we do not know which side a boundary location **A** belongs to; our intervening contour scheme allows **A** to connect to both sides, leading to a “tainted” flow estimate. (b) Using local boundary orientation, we fit affine motions to the two sides, and assign **A** to the side that better explains the motion at **A**. In this case, the figureness estimate is 0.77 left vs 0.23 right. This estimate boosts connections to the left side and suppresses those to the right, leading to an accurate estimate (c). A similar example is found in (d-f). There, figure-ground assignment also helps the estimate at an off-boundary location **B**, because the influence from **A** to **B** is reduced.

Suppose we have an initial estimate of flow  $\hat{\mathbf{v}}_j$ , we need to warp the second image to compute incremental temporal differences  $I_t$ . One approach would be to do local warping around each sample point. We instead choose to compute a single global warping. This warping requires a dense flow. We use a simple linear interpolation scheme based on Delaunay triangulation to project the flow on the sample points onto the pixel-grid.

Having computed the temporal differences in **b**, it is easy to incorporate incremental flow into the cost function in Eq. 2: now the data term depends on the difference  $\mathbf{v}_j - \hat{\mathbf{v}}_j$ . We also need an initial estimate on  $\vec{\alpha}$  to start the robust estimation: we use a least squares fit on the current estimates  $\hat{\mathbf{v}}_j$  (i.e. not using the data terms).

### 4.3. Figure/ground assignment of boundaries

When we define pairwise affinity in Section 3.2, we leave open the question of figure-ground assignment at boundaries. Suppose we have obtained a motion estimation  $\hat{\mathbf{v}}_j$ : we can assign a figure/ground label to boundary points based on motion similarity.

At each point  $i$ , the  $Pb$  boundary map provides a local contour orientation (in which contrast is at maximum). Using this orientation we divide sample points into two groups: “left” and “right”. We estimate two affine mo-

tions,  $\vec{\alpha}_L$  and  $\vec{\alpha}_R$ , in the two half-disks, analogous to motion boundary detection (e.g. [8]).  $\vec{\alpha}_L$  would predict a flow  $\mathbf{v}_i^L$  at  $i$ , and  $\vec{\alpha}_R$  would predict a different one  $\mathbf{v}_i^R$ . We compare these two predictions against the original estimate  $\hat{\mathbf{v}}_i$  (which uses points from both sides), by computing the ratio

$$r_F = \|\mathbf{v}_L - \hat{\mathbf{v}}\| / (\|\mathbf{v}_L - \hat{\mathbf{v}}\| + \|\mathbf{v}_R - \hat{\mathbf{v}}\|)$$

To increase robustness, we add a small constant (0.1) to the two differences. Because figure-ground assignment only makes sense at boundaries, we modulate  $r_F$  with the boundary contrast  $E$  to obtain a figure-ground estimate  $F$ :

$$\log [F/(1 - F)] = \log [r_F/(1 - r_F)] \cdot E$$

and multiply the affinity  $g(i, j)$  from  $i$  to  $j$  with  $2F(i)$  and  $\exp[-c_R(F(j) - 0.5)E(j)]$  (the latter term is used to negate the “upward” gradient near point  $j$ ).

### 4.4. From sparse to dense flow

For some applications, a sparse flow suffices. For others, a dense flow is desirable. Because we have a relatively uniform sampling (at both corners and edges), obtaining dense flow is not difficult.

Consider a standard linear interpolation: construct a Delaunay triangulation of the sample points, and for each triangle, use Barycentric coordinates in the interior to average flow at the three vertices. Such a scheme works well in region interior but encounters problems at motion discontinuities. Again this is a figure-ground issue: flow on the boundary should not be extended into the “wrong” side. This problem is only minor, and we use a simple fix by discounting flow on boundaries. In the standard scheme, three vertices on the triangle are treated as equally important; we instead weight the vertices by  $\exp(-c_I E)$ <sup>1</sup>. If a triangle touches a strong contrast boundary, we rely on flow on the interior points to fill it in.

## 5. Experiments

For experimental evaluation, we test our group-based approach on the Middlebury public flow dataset [5]. It is part of a new flow dataset with groundtruth that presents a variety of challenges, including hidden texture, realistic and complex scenes and non-rigid motion. For fair comparison, we take the gray-scale and two-frame version of the dataset as input, and use only the brightness constancy assumption. We also test on the Marble sequence [26], using frame 35 – 36 for evaluation.

We use a sampling distance of 5 pixels, and each sample point averages motion in a  $5 \times 5$  neighborhood. The number of sample points on the images range from 4000 to 6000.

<sup>1</sup>The figure-ground estimates in Section 4.3 may also be used for weighting; in practice it has offered little improvements.

For spatial integration, we set a threshold on affinity at 0.5, and use a minimum number of 10 points and a maximum of 40 points. We work with a pyramid of level 2 and use 5 iterations at the coarse level and 25 at the fine level. Figure-ground estimation is done only twice, at iteration 15 and 20. There are two parameters in the robust affine estimation, a constant in the Lorentzian function, and a constant on the regularization of linear terms. We use the same number for both, and decrease it from  $1e - 5$  to  $1e - 10$  as we iterate. Same parameters are used in all the experiments.

We evaluate results with two standard error measures [6], *average angular error* (AAE), and *average end-point error* (AEE). For comparison, we include results from the classical algorithm of Black and Anandan (1996) [10]<sup>2</sup>, which ranks around the top on the Middlebury dataset [5]. Quantitative comparisons are listed in the table in Figure 7. In addition, we show color-coded results along with groundtruth flow, and visualize angular errors at the end for analysis.

Our experiments show that grouping significantly improves optical flow estimation. We consistently do better, especially on the “hard” examples, such as Dimetrodon (hidden texture) and RubberWhale (non-rigid motion). Qualitatively, we do well in a number of difficult situations, including weak-contrast and untextured regions such as in Venus (the examples in Figure 1) and Dimetrodon, or small regions near motion boundaries such as the corners in Venus and the holes and the rotating wheel in RubberWhale. In many cases, this is because we successfully prevent the propagation of boundary motion into the “wrong” side.

We observe that the improvement is smaller on Hydrangea; there is not much grouping in that example (one object against one background), and the motion of the object is not affine. We still do better than Black & Anandan near the object boundary and improve the score. Similarly, the Marble example is highly textured with little static boundary contrast; nevertheless our result is comparable to what has been reported in the literature (e.g. [27, 1]).

It is interesting to look at the visualization of error in Figure 7 and see where we do poorly. A revealing example can be found in RubberWhale (on the right center): if grouping fails (and we know grouping is not perfect), it will hurt flow estimation and cause large errors. Figure 6 explores the problem in detail. A similar failure can be found in the Venus image (near bottom center). While this calls for better grouping cues than *Pb*, it also suggests that static image grouping should be combined with motion or “common fate” to improve affinity estimation.

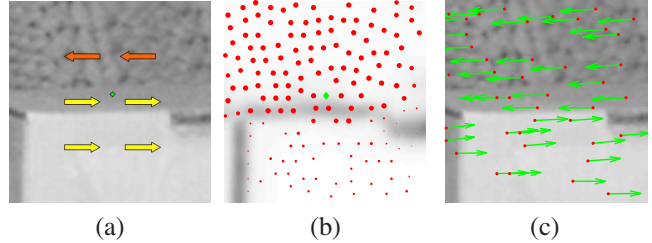


Figure 6. An example where incorrect grouping hurts optical flow: (a) a zoom-in on the RubberWhale image, where the grey band in the middle moves rightward. (b) The boundary contrast from the gray region to the white region below is much stronger than that to the textured region above; hence the green-diamond location is incorrectly grouped together with the textured region, causing its flow estimate to go leftward (c).

## 6. Discussions

In this work we have incorporated image-based grouping into optical flow. We apply a local boundary operator and an asymmetric *intervening contour* scheme to compute affinities between points. Pairwise affinity defines local spatial- and scale-adaptive support for motion integration, and allows accurate recovery of flow near motion boundaries and in weak-contrast regions. Quantitative evaluations on challenging datasets show that grouping significantly improves flow estimation over state-of-the-art results.

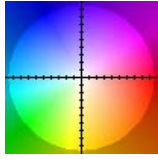
Local motion integration at a large and adaptive scale incurs a high computational cost, and our current approach is not nearly as efficient as existing methods that are close to real-time (it takes minutes per image). Nevertheless, we have shown that sparse sampling makes such complex local computations feasible while still capturing most motion information and faithfully representing flow fields. Such a sampling strategy opens up many possibilities for accurate flow computation that could go beyond popular PDE-based methods.

Grouping in our current approach is solely based on brightness and texture contrasts, because we want to demonstrate how static image analysis at the mid-level can help motion estimation. As we have observed, such grouping is never perfect and may cause problems where it fails. It would need to be combined with “common fate”, or motion-based grouping [7, 38, 32], to avoid making gross errors. On the other hand, the figure-ground analysis in our work is solely based on motion. Recent works on figure-ground [28, 15] have revealed that there is rich information available in a single image as well. It would be interesting to combine the strengths of both.

## References

- [1] L. Alvarez, R. Deriche, T. Papadopoulos, and J. Sanchez. Symmetrical dense optical flow estimation with occlusions detection. *Int'l. J. Comp. Vision*, 75(3):371–385, 2007.

<sup>2</sup>We thank the authors for providing their implementation. Default parameters are used in these experiments.



	Dimetrodon		Hydrangea		RubberWhale		Venus		Marble	
	AAE	AEE	AAE	AEE	AAE	AEE	AAE	AEE	AAE	AEE
Our approach	<b>3.34</b>	0.17	<b>2.77</b>	0.26	<b>5.32</b>	0.17	<b>3.93</b>	0.26	<b>4.88</b>	0.16
Black & Anandan	7.77	0.39	3.04	0.32	8.14	0.27	7.30	0.54	5.76	0.19



Dimetrodon



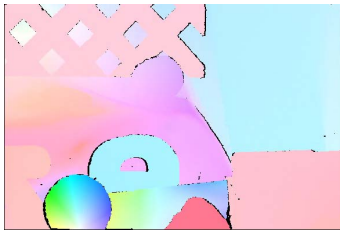
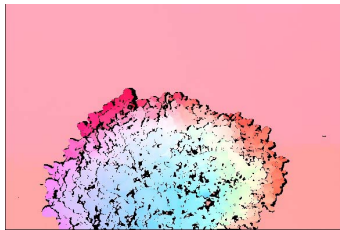
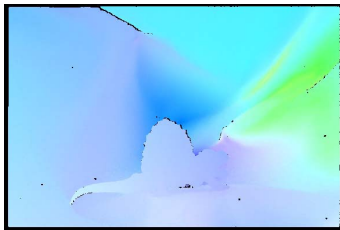
Hydrangea



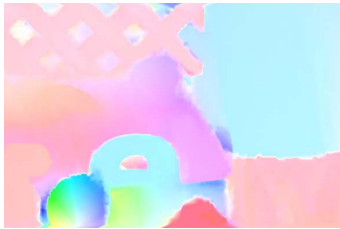
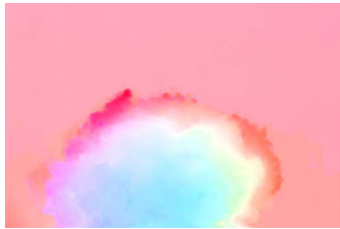
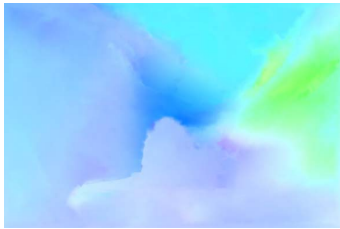
RubberWhale



Venus



color-coded groundtruth flow



color-coded results using grouping



angular error visualization

Figure 7. Experimental results: quantitative evaluations are listed in the table, using both angular and end-point error measures. Actual flows are visualized using the color coding shown on the top-left. We visualize both the groundtruth and our results, as well as the angular errors (capped at  $100^\circ$ ). We do well in difficult situations, such as the corners and weak-contrast regions in Venus and the small holes and rotating wheel in RubberWhale. Large errors in RubberWhale and Venus are due to incorrect grouping. In Dimetrodon, the lack of contrast at motion discontinuities causes over-smoothing. Many errors in RubberWhale are caused by shadow movements. Errors near boundaries in Venus are partly due to localization in sampling and partly imperfect interpolation.

[2] L. Alvarez, J. Weickert, and J. Sanchez. Reliable estimation of dense optical flow fields with large displacements. *Int'l.*

*J. Comp. Vision*, 39(1):41–56, 2000.

[3] S. Ayer and H. Sawhney. Layered representation of motion

- video using robust maximum-likelihood estimation of mixture models and mdl encoding. In *ICCV*, pages 777–784, 1995.
- [4] S. Baker and L. Matthews. Lucas-kanade 20 years on: a unifying framework. *Int'l. J. Comp. Vision*, 56(3):221–255, 2004.
- [5] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski. A database and evaluation methodology for optical flow. In *ICCV*, 2007.
- [6] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *Int'l. J. Comp. Vision*, 12(1):43–77, 1994.
- [7] M. Black. Combining intensity and motion for incremental segmentation and tracking over long image sequences. In *ECCV*, pages 485–493, 1992.
- [8] M. Black and D. Fleet. Probabilistic detection and tracking of motion boundaries. *Int'l. J. Comp. Vision*, 38(3):231–245, 2000.
- [9] M. Black and A. Jepson. Estimating optical flow in segmented images using variable-order parametric models with local deformations. *IEEE Trans. PAMI*, 18(10):972–986, 1996.
- [10] M. J. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, 1996.
- [11] A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnörr. A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. *Int'l. J. Comp. Vision*, 70(3):257–277, 2006.
- [12] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *Int'l. J. Comp. Vision*, 61(3):211–231, 2005.
- [13] C. Fowlkes, D. Martin, and J. Malik. Learning affinity functions for image segmentation: combining patch-based and gradient-based approaches. In *CVPR*, pages II:54–61, 2003.
- [14] B. Heisele, U. Kressel, and W. Ritter. Tracking non-rigid moving objects based on color cluster flow. In *CVPR*, pages 257–260, 1997.
- [15] D. Hoiem, A. Efros, and M. Hebert. Recovering occlusion boundaries from a single image. In *ICCV*, 2007.
- [16] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 2003.
- [17] A. Jepson and M. Black. Mixture models for optical flow computation. In *CVPR*, pages 760–761, 1993.
- [18] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: theory and experiment. *IEEE Trans. PAMI*, 16(9):920–932, 1994.
- [19] S. Khan and M. Shah. Object based segmentation of video using color, motion and spatial information. In *CVPR*, pages 746–51, 2001.
- [20] T. Leung and J. Malik. Contour continuity in region-based image segmentation. In *ECCV*, pages I:544–59, 1998.
- [21] C. Liu, W. Freeman, and E. Adelson. Analysis of contour motions. In *Advances in Neural Information Processing Systems 19*, 2006.
- [22] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of IJCAI*, pages 674–679, 1981.
- [23] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *Int'l. J. Comp. Vision*, 43(1):7–27, June 2001.
- [24] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color and texture cues. *IEEE Trans. PAMI*, 26(5):530–549, 2004.
- [25] H. Nagel and H. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from images sequences. *IEEE Trans. PAMI*, 8:565–593, 1986.
- [26] M. Otte and H. Nagel. Optical flow estimation: advances and comparisons. In *ECCV*, pages 51–70, 1994.
- [27] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *Int'l. J. Comp. Vision*, 67(2):141–158, 2006.
- [28] X. Ren, C. Fowlkes, and J. Malik. Figure/ground assignment in natural images. In *ECCV*, volume 2, pages 614–627, 2006.
- [29] S. Roth and M. Black. On the spatial statistics of optical flow. In *ICCV*, pages I:42–49, 2005.
- [30] P. Sand and S. Teller. Particle video: Long-range motion estimation using point trajectories. In *CVPR*, pages II:2195–2202, 2006.
- [31] J. Shi and J. Malik. Normalized cuts and image segmentation. In *CVPR*, pages 731–7, 1997.
- [32] J. Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *ICCV*, pages 1154–1160, 1998.
- [33] J. Shi and C. Tomasi. Good features to track. In *CVPR*, pages 593–600, 1994.
- [34] E. Simoncelli, E. Adelson, and D. Heeger. Probability distributions of optical flow. In *CVPR*, pages 310–5, 1991.
- [35] A. Stein, D. Hoiem, and M. Hebert. Learning to find object boundaries using motion cues. In *ICCV*, 2007.
- [36] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV*, pages 839–846, 1998.
- [37] Z. Tu, X. Chen, A. Yuille, and S. Zhu. Image parsing: Unifying segmentation, detection, and recognition. *Int'l. J. Comp. Vision*, 63(2):113–140, 2005.
- [38] J. Wang and E. Adelson. Representing moving images with layers. *IEEE Trans. Im. Proc.*, 3:625–638, 1994.
- [39] Y. Weiss. Smoothness in layers: Motion segmentation using nonparametric mixture estimation. In *CVPR*, pages 520–526, 1997.
- [40] Y. Weiss and E. Adelson. A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models. In *CVPR*, pages 321–6, 1996.
- [41] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isnardi. Bilateral filtering-based optical flow estimation with occlusion detection. In *ECCV*, pages I:211–224, 2006.
- [42] S. Yu. Segmentation induced by scale invariance. In *CVPR*, pages I:445–451, 2005.
- [43] L. Zitnick, N. Jovic, and S. B. Kang. Consistent segmentation for optical flow estimation. In *ICCV*, pages II:1308–15, 2005.