

# Accurate Computation of Optical Flow by Using Layered Motion Representations

Steve Hsu      P. Anandan

David Sarnoff Research Center  
Princeton, NJ 08543-5300, U.S.A.

Shmuel Peleg  
Institute of Computer Science  
The Hebrew University  
91904 Jerusalem, ISRAEL

## Abstract

*This paper presents a framework combining two prevailing approaches to motion analysis: optical flow which describes motion at each point, and methods that define global motions for larger regions. Image motion is represented by layers, image regions whose coherent motion can be approximated by some parametric motion model. The motion at every point is obtained by the parametric motion estimate of the entire layer, corrected by a residual flow field which captures the difference between the real image motion and the layer's motion model. The new approach is able to construct accurate flow fields in the presence of multiple motions, motion boundaries, and transparent motions.*

## 1 Introduction

This paper describes an approach for accurately computing optical flow by combining layered representation of global parametric motion with local residual estimates. In general, optical flow estimation requires the use of some type of spatial coherence constraint in order to obtain estimates in regions of low texture or aperture effect. The standard formulation of this constraint in the form of a smoothness assumption is not adequate to handle dynamic scenes whose projection on the image plane include multiple moving regions or multiple surfaces. Recently there has been a trend towards using global parametric motion models (e.g., see [2]) in order to obtain accurate estimates over an entire region.

An important and necessary extension of parametric motion models is the use of multiple motion estimation leading to a *layered representation* [10, 13, 7]. This approach represents the scene with multiple surfaces or layers, each moving with a single global parametric transformation. However, there are two difficulties with this approach. First, the global parametric models, while being very effective and efficient for obtaining good overall alignments of the layer, do not capture the detailed variation of flow from pixel to pixel. Second, the separation of images into layers is a difficult problem, and while a number of different solutions have been proposed for it, the fundamental issues involved in solving this problem are still unclear.

The approach described in this paper closes the circle on this line of work, by taking the problem back

to accurate optical flow computation. In this regard, the layered parametric motion representation can be viewed as a generalization of the spatial coherence assumption. By combining the layered motion fields together with pointwise residuals, we expect to obtain the benefit (in terms of the accuracy, stability, and efficiency) of layered parametric motion models while being able to account for local variation of flow.

## 2 Estimation of layered representations and flow

The basic problem of optical flow estimation based on parametric motion of layers plus residuals can be described as follows: Given some image frames determine an optical flow field based on the decomposition of the image into a set of layers, each of which is described by a parametric motion model, such as the affine model  $(u, v) = (p_1 + p_2x + p_3y, p_4 + p_5x + p_6y)$ . The total motion of each pixel is expressed as the combination of the layer's motion at that pixel together with a residual flow vector. Each pixel derives its flow vector as a single layer plus residual, while each layer has a set of pixels whose information contributes to the estimation of the motion parameters of the layer (called the *support set*), although it is important to note that it is *not* required that the support sets of different layers be mutually exclusive. (Hence, the support set and the classification map are not inverses of each other.)

The overall requirement is that the flow field describe the image motion at all the pixels as accurately as possible. We impose the following constraints on the problem: (i) the matched pixels between the images should be "similar" according to some similarity metric, (ii) the residual motion of each pixel relative to the layer motion should small as possible, (iii) the number of layers should be as small as possible, and (iv) the overall residual measure (appropriately defined) of the support of each layer relative to the layer's motion should be as small as possible. In this paper, we do not develop a complete and rigorous mathematical formulation of each of these constraints; rather, our aim is to identify the primary constraints and explore strategies for formulating them and implementing them.

As expected, the basic method is to start with crude

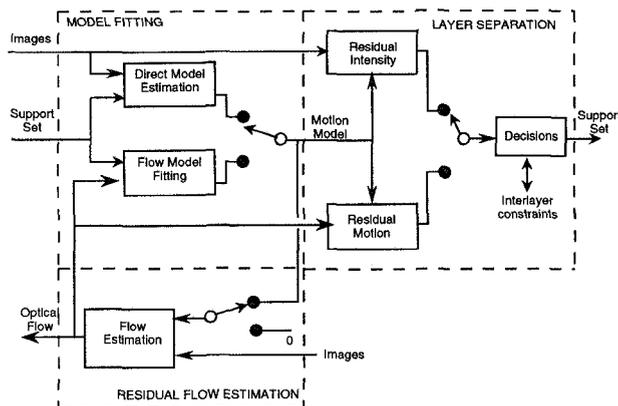


Figure 1:  
Framework for the estimation of layered motions.

estimates of layer models or support sets and iterate among flow computation, motion model estimation, and segmentation. As shown in Figure 1, this framework is non-committal with respect to choices made at the various decision points. Those choices lead to particular algorithms, which resemble existing algorithms in the literature (e.g., [10, 13, 7]). Framework components are now examined.

**Parametric Motion Models:** The motion modeling module estimates a parametric model that fits the coherent component of motion of any set of points given to it. Choices include fitting a regression model to a motion field computed by the flow estimation module [13] and direct parameter estimation from image gradients [10, 3, 9]. Unless segmentation is already correct, a set of pixels being fitted may contain multiple motions besides a dominant coherent component. Flow regression demands techniques like M-estimation [11] to avoid outlier effects. Hierarchical gradient-based algorithms developed for parametric motion model estimation [2] usually tend to lock onto the dominant component [6].

**Layer Separation:** The segmentation module determines the support sets of one or more layers given their motion models. The approach is to base the decision on a local measure of fit between each model and the local information over an *elemental area* that could be single pixels or small regions. Choices considered for the measure of fit include: *residual flow* with respect to the layer motion, the *parametric error*, i.e. the difference between the model and parameters obtained by a local fit to flow around the pixel, and *residual intensity* between images after compensating for the layer's global motion [12]. These measures can be integrated over a neighborhood to promote spatial coherence and reduce noise sensitivity. Spatiotemporal coherence [13, 4, 10] and mutual exclusion could also be used to make layers correspond more closely to physical scene surfaces.

**Flow Estimation:** The flow estimation module is responsible for estimating an image-wide motion

field close to any specified parametric motion model. This is performed by registering the images by that model, computing the residual flow using any of the standard flow estimation algorithms, then composing the result with the model. To feed the motion modeling or segmentation modules, the given model can be, depending on the control strategy, either (i) fixed as 0, in which case flow need only be computed once—prior to framework iterations [13]—or else (ii) set to the current motion model of each successive layer.

Flow estimates are usually most accurate when the given model is close to actual motion, since small residual flows can be more accurately estimated: there is less need to invoke large scale spatial smoothness during estimation, which tends to reduce accuracy and spatial resolution of the results. Therefore, to construct an accurate flow field for a multiple-motion scene, all the flow fields from (ii) should be multiplexed on a pixel-wise basis according to the segmentation mask.

### Control Strategy

Motion modeling, segmentation, and optical flow estimation are the top level computational modules of the algorithm framework. The control strategy determines when layers are added and updated. Three paradigms include:

*Sequential*—layers are successively added to the representation, initializing the support set to the whole image for the first layer and previously unclaimed pixels for subsequent layers [5, 3, 10].

*Parallel*—multiple layer motion models and supports are refined concurrently, as in K-means clustering. Each layer initially has either a random model or random support set, e.g. a small patch of connected pixels, likely to be part of a coherent motion.

*Hierarchical*—split existing classes when homogeneity tests are not satisfied [8, 9].

## 3 Results

Two distinct algorithms consistent with the foregoing framework were implemented. The two major choice points in the framework are the control strategy and the motion model estimation technique. In order to cover these alternatives, one algorithm used a parallel control strategy with both direct and flow-based model estimation, and another one used a sequential control strategy with direct model estimation. The first of these, which was inspired by [13], is farther along in its development and was used to produce the experimental results included in this section.

This algorithm updates a fixed number of layers in parallel and consists of two phases, first with blocks of pixels as *elements* and then with individual pixels. During phase 1, each layer support starts out as a single randomly chosen small block and, via framework iterations, grows to include all consistently moving blocks. This phase is restarted several times to seek lower error solutions, then phase 2 starts out with the best motion models of phase 1. In phase 1, a hierarchical gradient-based algorithm [1] is initially used to estimate optical flow, starting from a null model, and affine motion models are regressed to

flow by M-estimation in each layer at each iteration. In phase 2, model parameters are estimated by a direct method. In both phases, support sets are updated by assigning each element to the layer minimizing measures of residual flow (with respect to the initial flow) and residual intensity integrated over elements, yielding nonoverlapping layers. (For comparison, the clustering step of [13] is equivalent to phase 1 using parametric error.) At the end, optical flows are computed starting from each layer model and multiplexed.

The characteristics of optical flow estimation via layered representations can be illustrated using the sequence shown in Figure 2a. In this scene, there are three principal moving regions: the background translates mainly to the right due to camera motion, the right boxer moves slightly to the right, and the left boxer moves down.

The parallel algorithm is used to extract a layered representation, with the number of layers fixed at four and the phase 1 block size set to  $20 \times 20$ . The layer classification map is indicated by shades of gray in Figure 2b. Besides extracting the three primary moving objects, the algorithm extracts a separate layer for the right boxer's arm, which moves slightly with respect to his body.

The frame difference after motion compensation by each layer's parametric motion model only is shown in Figure 3a. In part, these residuals are caused by insufficiency of an affine model to fully account for motion in the extracted layers, due to parallax motion and nonrigid objects.

Finally, optical flow is computed within each layer, initialized with the corresponding affine motion model. As shown in Figure 3b, the resulting motion-compensated difference energy is much smaller than with affine registration alone. The most significant improvement in optical flow, shown in Figure 4, is that sharp discontinuities of flow at motion boundaries are well preserved. For comparison, optical flow estimation without layers, i.e. the flow starting from a null model, yields the flow fields shown in Figure 5. Since the same hierarchical optical flow algorithm is used in all the cases, the improvement can be attributed to the use of layers.

## 4 Summary

This paper described a framework which is an alternative to the prevailing approaches to motion analysis. Global parametric motion models are used to segment the image into layers. This is interleaved with optical flow estimation which refines the layered representation's models to allow for local flow variation. Experimental results based on a parallel layer extraction strategy show that this approach extends the ability of standard optical flow estimators to deal with multiple motions and motion boundaries.

## References

[1] J.R. Bergen and E.H. Adelson. Hierarchical, computationally efficient motion estimation algorithm. *J. Opt. Soc. Am. A.*, 4:35, 1987.

- [2] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *European Conference on Computer Vision*, pages 237-252, Santa Margarita Ligure, May 1992.
- [3] J.R. Bergen, P.J. Burt, K. Hanna, R. Hingorani, P. Jeanne, and S. Peleg. Dynamic multiple-motion computation. In Y.A. Feldman and A. Bruckstein, editors, *Artificial Intelligence and Computer Vision: Proceedings of the Israeli Conference*, pages 147-156. Elsevier, 1991.
- [4] M.J. Black and P. Anandan. A model for the detection of motion over time. In *International Conference on Computer Vision*, pages 33-37, Osaka, Japan, December 1990.
- [5] P.J. Burt, J.R. Bergen, R. Hingorani, R. Kolczynski, W.A. Lee, A. Leung, J. Lubin, and H. Shvaytser. Object tracking with a moving camera, an application of dynamic motion analysis. In *IEEE Workshop on Visual Motion*, pages 2-12, Irvine, CA, March 1989.
- [6] P.J. Burt, R. Hingorani, and R.J. Kolczynski. Mechanisms for isolating component patterns in the sequential analysis of multiple motion. In *IEEE Workshop on Visual Motion*, pages 187-193, Princeton, NJ, October 1991.
- [7] T. Darrell and A. Pentland. Robust estimation of a multi-layered motion representation. In *IEEE Workshop on Visual Motion*, pages 173-178, Princeton, New Jersey, October 1991.
- [8] N. Diehl. Object-oriented motion estimation and segmentation in image sequences. *Signal Processing: Image Commun.*, 3:23-56, 1991.
- [9] M. Hoetter and R. Thoma. Image segmentation based on object-oriented mapping parameter estimation. *Signal Processing*, 15(3):315-334, October 1988.
- [10] M. Irani and S. Peleg. Computing occluding and transparent motions. *International Journal of Computer Vision*, accepted for publication, 1993.
- [11] P. Meer, D. Mintz, A. Rosenfeld, and D.Y. Kim. Robust regression methods for computer vision: A review. *International Journal of Computer Vision*, 6:59-70, 1991.
- [12] S. Peleg and H. Rom. Motion based segmentation. In *International Conference on Pattern Recognition*, volume 1, pages 109-113, Atlantic City, June 1990.
- [13] J. Wang and E. Adelson. Layered representation for motion analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 361-366, New York, June 1993.

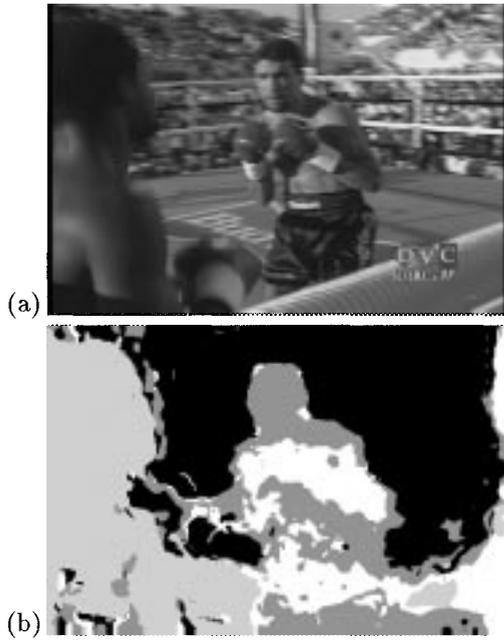


Figure 2:  
 (a) One frame of a 3-frame sequence. There are 3 main moving objects in this scene: the background and boxing ring, the left boxer, and the right boxer.  
 (b) Classification map of extracted layered representation.

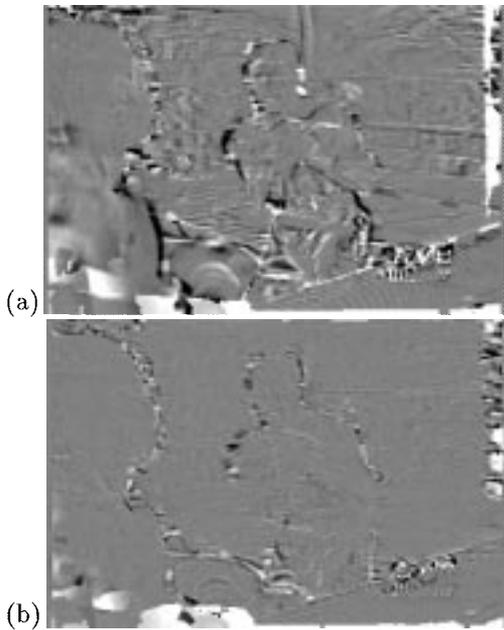


Figure 3:  
 (a) Brightness difference after registration using only affine motion model for each layer.  
 (b) Reduced difference using flow field of layered algorithm, which corrects affine model by a residual flow.

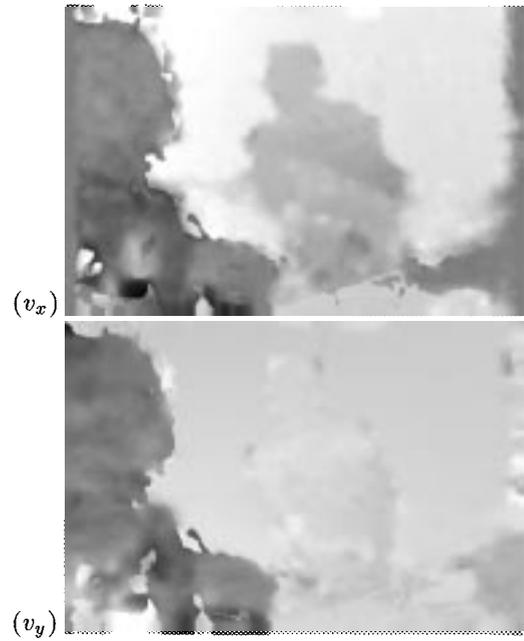


Figure 4:  
 Velocity components from layered flow estimation algorithm, dark=leftward or downward motion. Sharp discontinuities of flow are preserved at motion boundaries.

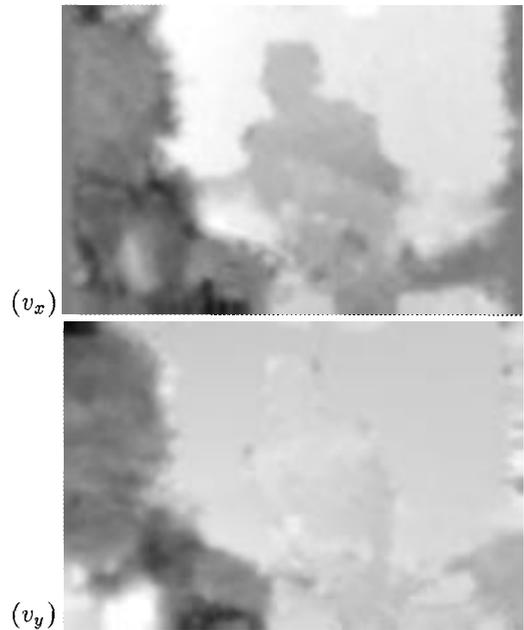


Figure 5:  
 Velocity components from standard optical flow estimation, without using layers.