M.I.T. Media Laboratory Vision and Modeling Group Technical Report No. 198

On the use of "Nulling" Filters to Separate Transparent Motions

Trevor Darrell and Eero Simoncelli
trevor@media.mit.edu eero@media.mit.edu

Vision and Modeling Group
The Media Laboratory
Massachusetts Institute of Technology
20 Ames Street
Cambridge MA, 02139

ABSTRACT

Transparent motions can be isolated in the Fourier domain, using derivative prefilters that selectively null the spatio-temporal energy consistent with the gradient constraint imposed by each component velocity. We use these "nulling" prefilters to robustly compute the support a particular velocity has at a point despite other velocities that may also be present at that point. Images are decomposed into a set of global constraints that account for a maximal amount of the motion information in the sequence. Adopting global motion hypotheses and a selection mechanism based on finding a parsimonious subset of these hypotheses, we show results separating motion stimuli into their constituent layers, even when those layers are transparently combined in the stimuli.

1 Introduction

The human visual system can easily distinguish multiple motions that are transparently combined in an image sequence. A model for the perception of transparent motion must address two questions: 1) what local motion measurements are made? and 2) how are these local estimates used to group coherently moving regions of the scene?

Several current computational approaches provide interesting insights into these issues. The algorithm of Shizawa and Mase [13, 14] directly computes two velocity vectors for each location in the image, but does not address the problem of perceptual grouping of coherently moving regions of the scene. The algorithms of Bergen et. al. [3] and Irani and Peleg [9] compute global affine optical flow fields, but use local measurements that are only capable of determining a single velocity estimate at each point.

We have adopted a model that combines the best of both approaches, using a global constraint mechanism with arbitrary spatial grouping, and local measurements that are robust in the presence of transparent motion. Our grouping mechanism is based on the idea of a "layered representation" proposed by Adelson [2, 1], in which each object or process in a scene is represented by a data structure describing aspects of the image attributable to that object.

In [7] we used the layered representation idea for the processing of motion information, allowing for arbitrary occlusion using layers with explicit regions of support for each moving object in the scene. While this work also applied to some forms of motion transparency, such as transparent random-dot displays ([8]), the local measurements used did not allow for true motion transparency. Here we remedy this deficit, incorporating a model of local motion measurements that accounts for motion transparency using the principle of superposition [13] together with a probabilistic model of velocity computation [15].

The grouping mechanism adopted in [7] assumed a purely spatial notion of support and thus could not explicitly represent transparent phenomena, since the support was constrained to be non-overlapping in the final solution. This paper extends the definition of support from an exclusively spatial notion, to include the spatio-temporal energy domain. The key insight is that when processing transparent motion displays, the support of a motion hypotheses should exist over both a region of space *and* velocity, so that it can be isolated both spatially and in terms of local velocity. With this formulation, two layers can have spatially overlapping support, but not be "conflicting" since they are distinct in the spatio-temporal energy domain.

2 Integration of Motion Information

The integration of motion information over a scene is is an essential part of any motion perception mechanism for two reasons. First, typical scenes have many regions for which the velocity is underconstrained (i.e., the intensity function is either one-dimensional or zero-dimensional). The motion in these regions must be disambiguated by combining constraints spatially and/or temporally. Second, combining constraints produces estimates that are more robust to noise that may corrupt the imaging or measurement processes.

If the motion in a scene is smooth, and there are no discontinuities in the optic flow, then this integration may be performed using a regularization process. When there are discontinuities, a grouping mechanism must determine which portions of the image are moving coherently, and thus should be integrated together. A common approach to motion grouping has been to use an edge field in combination with a velocity field, and use a smoothing mechanism that does not integrate information across edge boundaries [4].

This type of "line-process" approach works well when the scene contains only simple occlusion, in which each object projects to a relatively large, connected region in the image. But in more complicated situations, such as when the projected regions of an object are very small and disjoint (i.e. random dot displays) and/or overlapping with other objects (i.e., when looking through a window with reflections), this approach cannot adequately model or recover the stimulus. To accommodate these types of phenomena, one needs a representation that allows for multiple velocities in a single neighborhood, and for the integration of information across disjoint regions of an image.

Instead of a line-process, we have advocated the use of a set of layered "support processes" [5, 6]. Based on the notion of support found in the robust estimation literature [10], the support for a given global motion is computed as the conditional probability of the velocity field implied by the motion given the observed image sequence. Typically, this reduces to a weighted (normalized) residual error computation.

Our approach to computing a layered representation involves a generate-and-test strategy to produce multiple layer hypotheses, followed by a pruning optimization to select a parsimonious set of these hypotheses. We assume a global model of motion in the scene, described by a parametrized flow field. Further, we assume that the space of flow fields is sufficiently smooth that a relatively coarse sampling of the flow parameter(s) will produce a set of vector fields that can be combined to reasonably approximate the actual motions in the scene.

In the next section, we will develop in detail the mechanism for computing the support of a given flow field in a particular image sequence. Following that we will review the clustering mechanism that selects an optimal set of candidate layers, and present the extension to handle the case of transparency.

3 Local Measurements of Motion Information

For a global motion hypothesis in the form of a velocity field, we wish to determine the set of points in the scene that have a motion that is consistent with the velocity field. If the scene contains only a single motion at every point, this is a straightforward task using the well-known gradient-based models of velocity computation [12].

3.1 The gradient constraint

Given a particular velocity field $v(x, y)$ and an image sequence $I(x, y, t)$, we wish to compute the support field $s(x, y)$ indicating those regions of the sequence with motion matching $v(x, y)$ (at a particular time t_0). According to the gradient constraint, the spatial and temporal image derivatives at a point obey the following equation:

$$v_x(x,y)\frac{\partial I(x,y,t)}{\partial x} \;+\; v_y(x,y)\frac{\partial I(x,y,t)}{\partial y} \;=\; -\frac{\partial I(x,y,t)}{\partial t}. \tag{1}$$

Using an operator notation,

$$D(v) = \begin{bmatrix} v_x \\ v_y \\ 1 \end{bmatrix} \cdot \nabla = (v_x\frac{\partial}{\partial x} \;+\; v_y\frac{\partial}{\partial y} \;+\; \frac{\partial}{\partial t}), \tag{2}$$

we can express eq. (1) as

$$D(v(x,y))I(x,y,t) \;=\; 0. \tag{3}$$

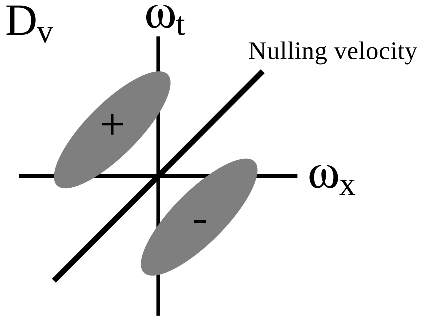

Figure 1: Idealized depiction of the spectrum of a directional derivative operator. The operator "nulls" energy along the line specified by the constraint velocity v, and passes any other energy. Thus the operator will have zero output, $D(v)I = 0$, if all of the energy lies on the constraint line.

Taken in the Fourier domain, the gradient constraint implies that all energy associated with a velocity v lies on a plane in space-time [17]. From eqs. (2,3) we can see $D(v)$ is a linear operator, and that it must have zero response for spatio-temporal frequencies on the constraint plane. We thus call $D(v)$ a "nulling" filter, as it removes any energy along the constraint plane from the image sequence. Figure 1 shows a plot of the spectral response of this filter for a 1D velocity; in this case, all energy is constrained to lie along a line in space-time.

3.2 Computing velocity support

We define the support for a velocity at a point as the conditional probability that a velocity is present given the observed image derivatives. Following [15], we introduce a noise model into the gradient constraint equation:

$$D(v(x,y) + n_1(x,y))I(x,y,t) \;=\; n_2(x,y) \qquad n_i = N(0,\sigma_i), \tag{4}$$

where the noise term n_1 describes errors in the planarity assumption, and n_2 accounts for any remaining errors in the temporal derivative measurement.

Assuming there is only a single motion in the neighborhood, the conditional probability of a velocity v given the image gradient ∇I is given by

$$-\log P(v|\nabla I) \propto \frac{||D(v)I||^2}{\sigma_1||\nabla I||^2 + \sigma_2} + \frac{||v||^2}{\sigma_v}, \tag{5}$$

where we assume a prior distribution of velocities of the form $P(v) \propto N(0,\sigma_v)$. In this case our support function is simply the conditional probability:

$$s(x,y) \;=\; P(v(x,y) \mid \nabla I(x,y,t)). \tag{6}$$

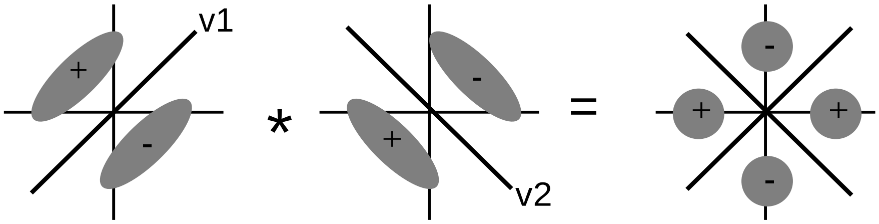

Figure 2: Idealized spectrum of a second-order derivative operator, constructed by composing two first-order directional derivatives. The operator "nulls" energy along the two lines specified by the constraint velocities v and v'. Thus the operator will have zero output, $D_2(v, v') = 0$, if all of the image energy lies on the constrained lines.

3.3 Additive models of motion transparency

When there is motion transparency in a scene, more than one constraint must be specified at each point in the image. To compute the support of a velocity field in the presence of transparency, we need to rely on local measurements that are robust in the presence of multiple motions at a given point. To derive these measurements, we need a local model of multiple (transparent) motion. For image sequences that have been additively combined, we can simply chain the constraint operators of the constituent velocities:

$$D(v_1(x,y))D(v_2(x,y))I(x,y,t) \;=\; 0 \tag{7}$$

In the spatio-temporal domain, this corresponds to the dual constraint that all of the energy in the signal lie on *either* of the planes specified by the two velocities.

Shizawa and Mase [13] introduced this formulation and proceeded to develop methods for the analytic estimation of two velocities at each point given the local image derivative information. Our focus, however, is on the computation of support, i.e. the likelihood of a velocity given the image information.

Constructing the dual-motion constraint operator, we have

$$\begin{aligned} D_2(v, v') = D(v)D(v') &= (v_x \frac{\partial}{\partial x} \;+\; v_y \frac{\partial}{\partial y} \;+\; \frac{\partial}{\partial t})(v'_x \frac{\partial}{\partial x} \;+\; v'_y \frac{\partial}{\partial y} \;+\; \frac{\partial}{\partial t}) \\ &= (v_x v'_x)\frac{\partial^2}{\partial x^2} \;+\; (v_y v'_y)\frac{\partial^2}{\partial y^2} \;+\; (v_x v'_y + v_y v'_x)\frac{\partial^2}{\partial xy} \;+ \\ &\quad (v_x + v'_x)\frac{\partial^2}{\partial xt} \;+\; (v_y + v'_y)\frac{\partial^2}{\partial yt} \;+\; \frac{\partial^2}{\partial t^2} \end{aligned} \tag{8}$$

and in this notation the dual motion constraint is simply

$$D_2(v_1(x,y), v_2(x,y))I(x,y,t) \;=\; 0 \tag{9}$$

This operator "nulls" two constraint planes (or lines, in the 1D case). Figure 2 shows a spectral plot of D_2 for two velocities.

3.4 Computing transparent support with "nulling" prefilters

If one of the motions in a sequence is known, perhaps due to known background motion, we can use the constraint operator as a "nulling prefilter" to remove the energy corresponding to that motion. This operation is similar in spirit to the predictive cancellation of Bergen et. al [3], however we perform the computation locally rather than globally.

We want to find the conditional probability of v given the observed image gradient *and* a known transparent velocity v', $P(v|\nabla I, v')$. To compute this, we first construct an intermediate sequence,

$$I' = D(v')I, \tag{10}$$

that removes any contribution of v' to the spatio-temporal spectrum of I. Neglecting any overlap in the energy distributions of v and v', we approximate

$$P(v|\nabla I, v') \approx P(v|\nabla I'). \tag{11}$$

We then compute $P(v|\nabla I')$, using eq. 5:

$$\begin{aligned} -\log P(v|\nabla I') &\propto \frac{||D(v)I'||^2}{\sigma_1||\nabla I'||^2 + \sigma_2} + \frac{||v||^2}{\sigma_v} \\ &\propto \frac{||D(v)(D(v')I)||^2}{\sigma_1||\nabla(D(v')I)||^2 + \sigma_2} + \frac{||v||^2}{\sigma_v}. \end{aligned} \tag{12}$$

Typically we don't know any of the motions in a sequence *a priori*. However, if we assume that there are only a small number of motions at a given point, [1] then we can quantize and enumerate the possible conflicting motions at a point, and selectively discount them using appropriately tuned "nulling" pre-filters.

For example, if two motions are possible at a single point, then to test the support for a velocity v we evaluate $P(v|\nabla I, v')$ over the range of possible v'. Because we are computing the conditional probability with nulling filters, we define the support to be the maximum value of this conditional probability over the range of possible conflicting motions, v'. This can be justified by the observation that the choice of an incorrect nulling filter (one corresponding to a velocity that does is not in the image at that point) will always yield a lower likelihood than the correct nulling filter. The dual-motion transparent support function is thus:

$$s(x,y) = \max_{v'} P(\ v(x,y) \mid \nabla I(x,y,t),\ v'\ ). \tag{13}$$

For three motions, we could similarly construct $P(v|\nabla I, v', v'')$:

$$P(v|\nabla I, v', v'') \propto \frac{||D(v)(D(v')(D(v'')I))||^2}{\sigma_1||\nabla(D(v')(D(v'')I))||^2 + \sigma_2} + \frac{||v||^2}{\sigma_v}, \tag{14}$$

for which the support function would be:

$$s(x,y) = \max_{v',v''} P(\ v(x,y) \mid \nabla I(x,y,t),\ v', v''\ ). \tag{15}$$

[1] Mulligan [11] has shown that most human observers cannot discriminate more than 2 local motions at a single point.

Higher-order support functions are possible, but at the expense of considerably increasing computational complexity. In practice we have limited our implementation to assume only two possible motions at a single point (however many different motions may be present in the overall scene.)

4 Selecting a set of Global Velocity Fields

A key issue in the recovery of a multi-layer representation of an image is the question of how many layers to use in describing that image. One approach, presented by [9], is to assume a spatially dominant "background" whose parameters can be estimated based on the entire image data, since the outlier contamination from the foreground will be relatively small. The support of the background is then computed using the weighted residual error computation given above. The background estimate can be further refined using an iterative robust estimation technique, re-estimating the parameters based only on the newly supported points. The motion of a foreground object can be similarly computed by initializing its support to be the complement of the background support. With multiple objects, this approach fails when two of the objects exist at the same scale. (In this case the percentage of outliers – the foreground support – will exceed the breakdown point of the robust estimation method [10].)

4.1 Hypothesize and select methods

An alternative approach is to test many different motion hypotheses for a given image sequence and decide which are worth retaining. With this approach, one has to specify a method of generating the initial set of hypothetical motion fields, and a selection mechanism that acts upon them.

Wang et. al. [16] have developed a layer description algorithm that fits motion parameters to initial patches of the image, and then computes the support associated with the estimated parameters. They have a computationally efficient mechanism that iteratively selects the layer with the largest support among uncovered portions of the image, and then marks the pixels associated with that layer as covered.

In [7], we presented a method that generated initial hypotheses either by sampling the parameter space, or by sampling the image domain (into small patches, for example) and fitting parameters to each sample. If one has knowledge about the spatial properties of the objects in a scene, e.g. that they have relatively large unobscured subregions, then this latter approach has advantages. However when this is not the case, for example with transparent phenomena, assumptions on the spatial geometry of the support fields are untenable. For scenes with transparent phenomena, direct initialization of hypotheses in the parameter space is thus appropriate. To find a subset of layers from the initial hypothesis set, we have adopted a selection mechanism that attempts to choose the smallest set of layers whose support covers as much of the scene as possible.

4.1.1 Subset selection method

Given a set of velocity fields and an image sequence, we compute the corresponding support each velocity field receives from the image using eq. 13. This gives us an initial hypothesis set $\mathcal{H}$:

$$\mathcal{H} = \{l_0, l_1, ..., l_M\} \tag{16}$$

$$l_i = \{v_i(x,y), s_i(x,y)\} \tag{17}$$

where $v_i(x,y)$ and $s_i(x,y)$ are the velocity field and support map (respectively) of the ith layer.

We attempt to find a subset $\mathcal{L} \subset \mathcal{H}$ that maximizes the amount of support covered by the layers in $\mathcal{L}$ minus a penalty term on the cardinality of $\mathcal{L}$. We maximize a criterion function:

$$S(\mathcal{L}) = C(\mathcal{L}) - \alpha||\mathcal{L}|| \tag{18}$$

where $C(\mathcal{L})$ is the amount of support covered by the given subset, and α is the penalty term for adding a new layer to the representation.

We thus need an expression for the amount of support covered by a given subset $\mathcal{L}$. If there happens to be no overlapping support among the elements of $\mathcal{L}$ (a rather unlikely proposition), then this is quite trivial:

$$C(\mathcal{L}) = \sum_{\{i|l_i \in \mathcal{L}\}} \left(\sum_x \sum_y s_i(x,y) \right). \tag{19}$$

When support does overlap, we do not want to count twice the support for the same velocity at the same point. We therefore introduce a weighting term, defined as the cosine of the angle between the two velocities, to determine how much they conflict.

Somewhat counter-intuitively, we do *not* count a supported point pixel as being covered if that point has conflicting support from more than one velocity hypothesis in $\mathcal{L}$. The optimization we perform incrementally updates a confidence value for each candidate layer based on the number of points it covers. Since it is difficult to attribute a point with conflicting support to a single hypothesis, we adopt a "least commitment" strategy and allocate it to neither. We have found that the alternative approach of normalizing the contribution of support point by the number of layers with conflicting support does not lead to a criterion function that can be successfully maximized using a gradient descent method.

We compute the set of points that have conflicting support by the following product:

$$c_i(x,y) = \prod_{\{k|k \neq i, l_k \in \mathcal{L}\}} (1 - \phi_{ki}(x,y) s_k(x,y)) \tag{20}$$

where ϕ_{ki} measures whether two velocities are "conflicting", and is defined to be the cosine of the angle between the normal vectors of velocity planes corresponding to v_k and v_i:

$$\phi_{ki}(x,y) = \frac{v_k(x,y) \cdot v_i(x,y) + 1}{\sqrt{||v_k(x,y)||^2 + 1}\sqrt{||v_i(x,y)||^2 + 1}} \tag{21}$$

This quantity $c_i(x, y)$ is used to indicate when there is conflicting support for a point; if more than one hypothesis in $\mathcal{L}$ shares the same spatio-temporal support at a point (x, y) then $c_i(x, y)$ will equal 0 for those hypotheses. Using this to discount conflicting support, the amount of support covered by $\mathcal{L}$ is

$$C(\mathcal{L}) = \sum_{\{i|l_i \in \mathcal{L}\}} \left(\sum_x \sum_y c_i(x, y) s_i(x, y) \right) \tag{22}$$

4.1.2 Numerical solution

To maximize S numerically, we represent $\mathcal{L}$ by enumerating those elements of $\mathcal{H}$ that are in $\mathcal{L}$ using a vector a, and then performing gradient descent on a . The sign of each a_i indicates whether a layer hypothesis $l_i \in \mathcal{H}$ is an element of the layer subset $\mathcal{L} \subset \mathcal{H}$: a positive value means $l_i \in \mathcal{L}$, and a negative value means $l_i \notin \mathcal{L}$. Initially a_i is set to zero for each description.

For a subset represented by a, the amount of covered support is

$$C(a) = \sum_i \sigma(a_i) \left(\sum_x \sum_y c_i(x, y) s_i(x, y) \right) \tag{23}$$

with

$$c_i(x, y) = \prod_{k \neq i} (1 - \sigma(a_k) \phi_{ki}(x, y) s_k(x, y)) \tag{24}$$

where $\sigma()$ transforms the a_i values into a multiplicative weight between 0 and 1. Ideally, $\sigma()$ would be the unit step function centered at the origin. However, this hard non-linearity makes it very difficult to find a maxima of S using traditional methods. Instead, we use the "softer" sigmoid function,

$$\sigma(x) = \frac{1}{1 + e^{-kx}} \tag{25}$$

Combining eqs. 18, 23, we have

$$S(a) = \sum_i \sigma(a_i) \left(\sum_x \sum_y c_i(x, y) s_i(x, y) - \alpha \right) \tag{26}$$

Using the shorthand notation $A_i = \sigma(a_i)$, we update the values of a_i with the following update rule:

$$\frac{da_i}{dt} = \frac{1}{k} \frac{dS(a)}{dA_i} = \frac{1}{k} \sum_x \sum_y c_i(x, y) s_i(x, y) - \alpha \tag{27}$$

where k is an integration constant. This rule increments a_i whenever the velocity layer has enough "unconflicted" support to offset the overhead cost penalty α. The convergence of the a_i values to a local maxima of $S(a)$ is shown in Appendix A.

5 An Example Recovering the Support of Transparent Motions

5.1 Velocity model

In our implementation, we adopt a simple yet non-trivial model of velocity fields in the scene. We use a linear combination of horizontal and vertical shifts together with a looming field:

$$V(x,y) = a\begin{bmatrix} 1 \\ 0 \end{bmatrix} + b\begin{bmatrix} 0 \\ 1 \end{bmatrix} + c\begin{bmatrix} x-d \\ y-e \end{bmatrix} \tag{28}$$

where a, b, c, d, e are the parameters of the model. Other authors have experimented with higher-order approaches, such as affine models (see [9, 16]).

In previously reported work [7], we demonstrated a system that used this velocity model, a simple version of the the support computation described in eq. 6, and a selection method equivalent to the one presented above without the model of non-conflicting velocities (thus it could not handle pure transparency). This system could recover the segmentation of multiple moving objects in a scene, as long as they obeyed the parametric model given above.

We have extended this implementation to use the transparent support mechanisms presented above. Support is computed using a fixed set of single-stage prefilters, as in eq 13. For the following examples we used a set of eight prefilters, with velocities $\{(-1,-1),\ (-1,0),\ (-1,1),\ (0,-1),\ (0,1),\ (1,-1),\ (1,0),\ (1,1)\}$. Since we have not yet implemented a coarse-to-fine strategy in our support computation, we assume the range of motions in the sequences to be processed are small, on the order of one or two pixels per frame.

5.2 Hypothesis initialization

Similarly, we adopted an initial set of velocity field hypotheses corresponding to 8 planar shifts:

$$\{(a,b,c)\} = \{(-1,-1,0),\ (-1,0,0),\ (-1,1,0),\ (0,-1,0),\ (0,1,0),\\ (1,-1,0),\ (1,0,0),\ (1,1,0)\}$$

plus a full field looming hypothesis $\{(0,0,1)\}$ with fixed offsets at the center of the image $(d = 64, e = 64)$. [2]

[2] For more complicated scenes than those used in the examples below a set of initial hypotheses that more finely samples our field model may be needed, but empirically we have found that a small number of hypotheses, typically less than the 64 our implementation can manage, will usually suffice to find support fields for an image sequence. Additionally, the examples presented here do not demonstrate the recovery of a transparently looming object, since although it is a valid instance of our motion model we did not have a sequence with transparent looming motion available as of the writing of this paper. We expect to report results on such scenes shortly. Results on sequences with non-transparent looming are reported in [7].

5.3 Results

We ran our system on several example images. We constructed sequences that contain two additively combined subimages moving over each other in different directions: $v_1 = (0.8, -0.8)$, $v_2 = (0.0, 0.8)$. In the sequence used for Figure 3, each subimage is a patch of bandpassed random noise. Figure 3(a) shows the first frame of the sequence used, and Figure 3(b) shows the support fields for the selected velocity fields. Figure 4 shows similar results, using two well-known face images.

Having a support function that is robust to a second transparent motion is useful both for cases of pure transparency, and to compute support near regions of occlusion (points of an image whose distance to an occluding boundary is within the filter radius of the derivative operator can be functionally considered to be transparent with respect to the multiple motion model described above). Thus we can recover more accurate support maps on images that only contain occlusion than we could using the original support function. Figure 5 shows an example sequence with a person moving behind a stationary plant. The support field computed using the non-transparent support function (eq. 6) is shown in Figure 5(b), as was reported in [7]. The support field computed using the transparent support function, eq. 13, captures the true area of the person more accurately, and is shown in Figure 5(c).

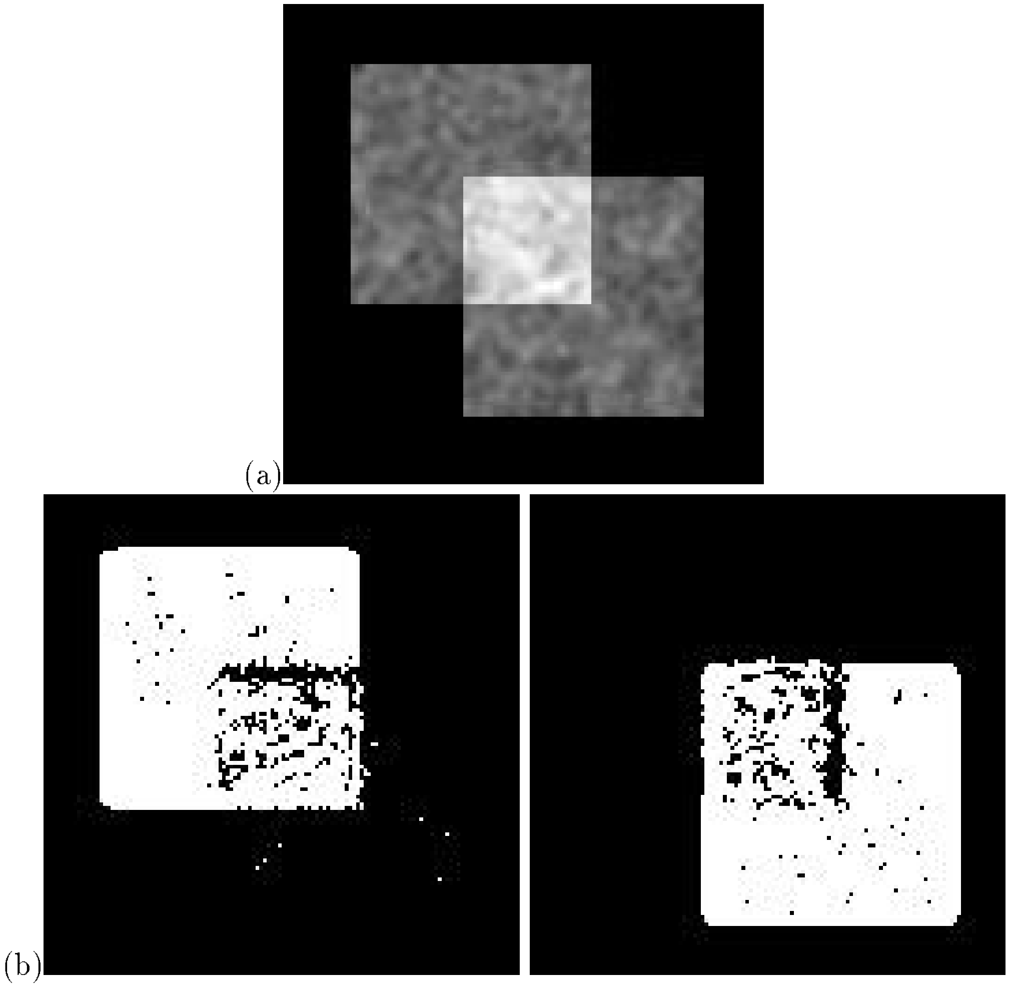

Figure 3: (a) image from sequence with transparently combined moving patches. (b) support fields found by selection mechanism.

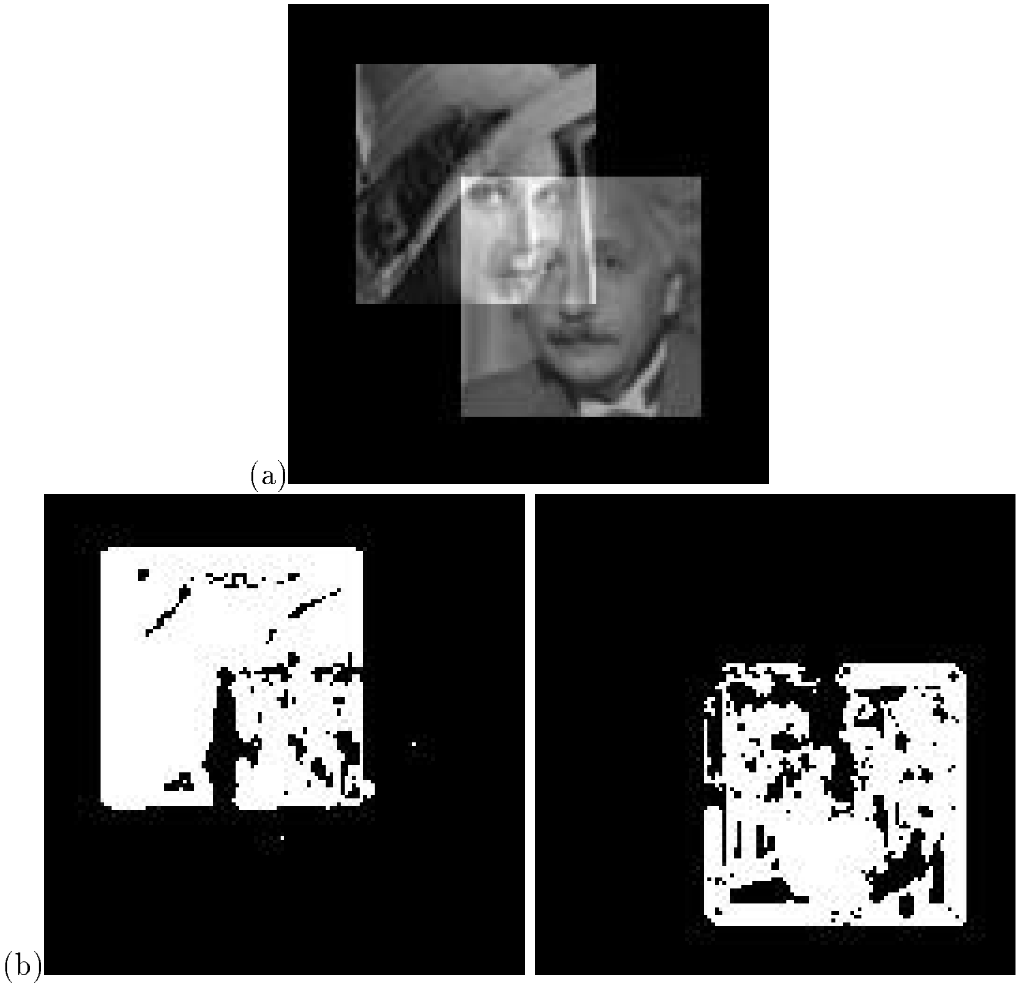

Figure 4: (a) image from sequence with transparently combined moving face images. (b) support fields found by selection mechanism.

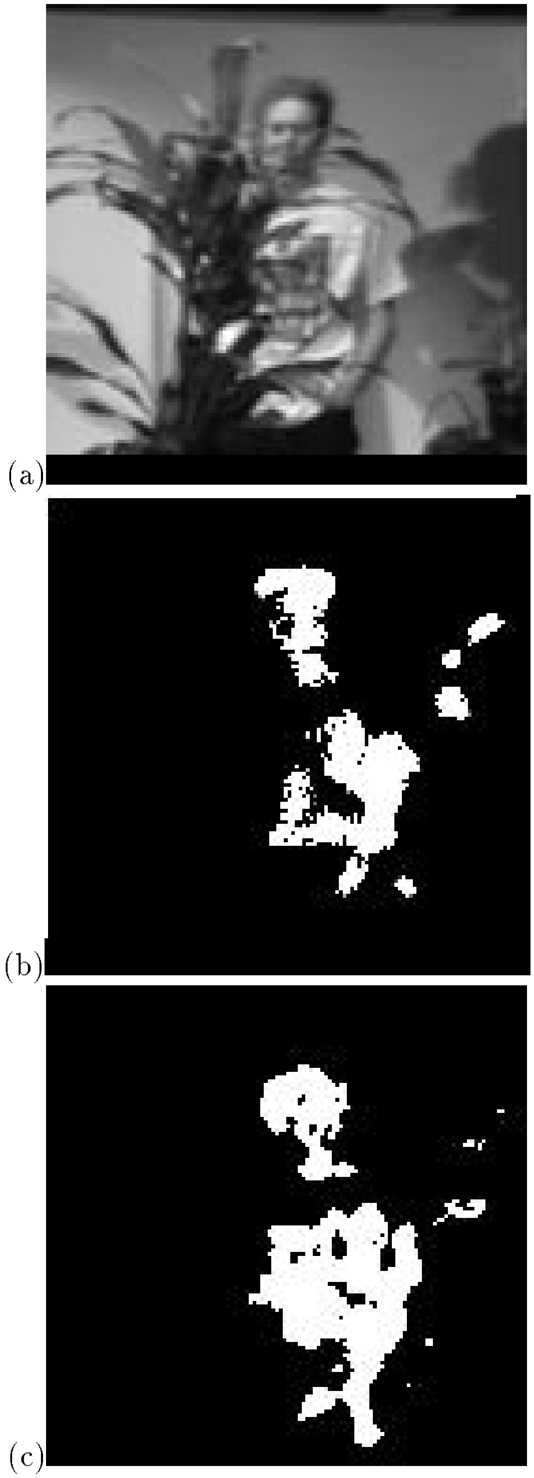

Figure 5: (a) image from sequence with person moving behind a plant. (b) support for person computed using non-transparent support function. (c) support for person computed using support function that is robust to transparent motion.

6 Acknowledgements

The authors would like to thank Ted Adelson for introducing us to the notion of "layers" and its power as an image representation for transparency, and Alex Pentland for providing several simplifying insights that helped us develop the selection method.

7 Summary and Conclusion

We have presented a new approach to the separation of transparent motions, using derivative prefilters to selectively null the spatio-temporal energy due to component velocities. In this paper we focus on the computation of support for a single velocity hypotheses in a manner that is robust to other transparent motions, rather than attempt to analytically estimate all the motions in a local patch. Using global motion hypotheses and a selection mechanism based on finding a parsimonious subset of these hypotheses, we are able to successfully decompose motion stimuli that contain additively combined transparent layers.

References

[1] E. H. Adelson. Layered representations for motion sequences. Technical Report TR-181, Vision and Modeling Group Technical Report, MIT Media Lab, December 1991.

[2] E. H. Adelson and P. Anandan. Ordinal characteristics of transparency. Technical Report TR-150, Vision and Modeling Group Technical Report, MIT Media Lab, July 1990.

[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*, pages 147–156. Elsevier Science Publishers B.V., 1991.

[4] M. Black and P. Anandan. Constraints for the early detection of discontinuity from motion. In *Proc Eighth National Conf on Artificial Intelligence*, pages 1060–1066, July 1990.

[5] T. Darrell and A. P. Pentland. Segmentation by minimal description. In *Proceedings Thrid Intl. Conference on Computer Vision*, Osaka, Japan, December 1990.

[6] T. Darrell and A. P. Pentland. On the representation of occluded shapes. In *Proceedings IEEE Conference on Computer Vision*, Maui, Hawaii, June 1991.

[7] T. Darrell and A. P. Pentland. Robust estimation of a multi-layer motion representation. In *Proceedings IEEE Workshop on Visual Motion*, Princeton, October 1991.

[8] M. Husain, S. Treue, and R. Andersen. Surface interpolation in three-dimensional structure-from-motion perception. *Neural Computation*, 1:324–333, 1989.

[9] M. Irani and S. Peleg. Image sequence enhancement using multiple motions analysis. Technical Report 91-15, Department of Computer Science Technical Report, The Hebrew University of Jerusalem, Israel, December 1991.

[10] P. Meer. Robust regression methods for computer vision: A review. *Intl. J. Computer Vision*, 6:60–70, 1991.

[11] J. Mulligan. Motion transparency is restricted to two planes. *Investigative Opthalmology and Visual Science Supplement (ARVO)*, 33:1049, 1992.

[12] H. H. Nagel. On the estimation of optical flow: relations between different approaches and some new results. *Artificial Intelligence*, 33:299–324, 1987.

[13] M. Shizawa and K. Mase. Simultaneous multiple optical flow estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, Atlantic City, June 1990.

[14] M. Shizawa and K. Mase. A unified computational theory for motion transparency and motion boundaries based on eigenenergy analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, Maui, June 1991.

[15] E. P. Simoncelli and E. H. Adelson. Probability distributions of optical flow. In *IEEE Conference on Computer Vision and Pattern Recognition*, Mauii, Hawaii, June 1991.

[16] J. Y. A. Wang and E. Adelson. Layered representations for image sequence coding. Submitted to *Fourth International Conference on Computer Vision*.

[17] A. B. Watson and A. J. Ahumada. A look at motion in the frequency domain. In J. K. Tsotsos, editor, *Motion: Perception and representation*, pages 1–10. 1983.

A. Convergence of selection method

Using eq. (27), $S(a)$ will converge to a local maxima. We can see this by noting that

$$\frac{dS(a)}{dt} = \sum_i \frac{dS(a)}{dA_i}\frac{dA_i}{dt}. \tag{29}$$

From 27 we have

$$\frac{dS(a)}{dA_i} = k\frac{da_i}{dt} \tag{30}$$

and thus

$$\frac{dS(a)}{dt} = \sum_i k\frac{da_i}{dt}\frac{dA_i}{dt} \tag{31}$$

But $a_i = \sigma^{-1}(A_i)$, so

$$\frac{da_i}{dt} = \frac{d\sigma^{-1}(A_i)}{dt} = \frac{d\sigma^{-1}(A_i)}{dA_i}\frac{dA_i}{dt} \tag{32}$$

and so

$$\frac{dS(a)}{dt} = \sum_i k\frac{d\sigma^{-1}(A_i)}{dA_i}\left(\frac{dA_i}{dt}\right)^2 \tag{33}$$

Since $\sigma^{-1}()$ is monotonically increasing,

$$k\frac{d\sigma^{-1}(A_i)}{dA_i}\left(\frac{dA_i}{dt}\right)^2 \geq 0 \quad \forall i \tag{34}$$

and thus $\frac{dS(a)}{dt} \geq 0$, and $S(a)$ will never decrease under eq. (27). Since the values of A_i are bounded, $S(a)$ is bounded, and thus must converge.