A Parallel Motion Algorithm
Consistent with Psychophysics and Physiology

Heinrich H. Bülthoff
Center for Biolog. Inform. Processing
Massachusetts Institute of Technology

James J. Little
Artificial Intelligence Laboratory
Massachusetts Institute of Technology

Tomaso Poggio
Artificial Intelligence Laboratory
Massachusetts Institute of Technology

Abstract
The precise measurement of the 2D field of velocities – which is
the projection of the true 3D velocity field – from time-varying
2-dimensional images is in general impossible. It is, however,
possible to compute suitable “optical flows” that are qualitatively
similar to the velocity field in most cases. We describe a simple,
parallel algorithm that successfully computes an optical flow from
sequences of real images, is consistent with human psychophysics
and suggests a plausible physiological model.

Regularizing optical flow computation leads to a formulation
which minimizes matching error and, at the same time, maximizes
smoothness of the optical flow. We develop an approximation to
the full regularization computation in which corresponding points
are found by comparing local patches of the images. Selection
among competing matches is performed using a winner-take-all
scheme. The algorithm is independent of the types of features
used for matching.

Experiments with natural images show that the scheme is ef-
eective and robust against noise. The algorithm shows several
of the same “illusions” that humans perceive. A natural physi-
ological implementation of the model is consistent with data from
cortical areas V1 and MT.

1 Introduction
If an observer moves in a three-dimensional world, a flow of ap-
parent motion (optical flow) occurs on his retina that is useful
for several different tasks such as images segmentation, structure
from motion and navigation.

The velocity field of object moving in space is a 3-D vector field
W(x, y, z). When projected into the retinal (image) coordinate
system of an observer, this field generates the projected velocity
field W'(x, y). What is observed on the retina is the image bright-
ness change Ei(x, y), Ei, 6(x, y). The optical flow, V(x, y), is a
time-varying vector field describing image brightness change. In
general, V(x, y) and W'(x, y) are not the same.

In the Vision Machine project [PLG88], our goal is to devise
robust methods for computing early vision modules and to inte-
grate these modules. The flexible, robust behavior of the human
visual system is in large part due to integration of many early
vision modules. The output of this integration stage is a map of
the physical discontinuities in the scene, in the image coordinate
system. An important input to this integration stage is the opti-
cal flow and its discontinuities, which provide important cues for
separating figure from ground, as has been demonstrated for the
visual system of vertebrates as well invertebrates [RPH83].

2 The Parallel Motion Algorithm
We describe a new parallel algorithm (Fig. 1) for computing opti-
cal flow in near real-time for natural images (256 x 256 pixels)
on the Connection Machine system [Hil85, LBC87]. We first out-
line some of the assumptions we use, and then detail the par-
ticular constraints applied to the optical flow field. Next, we
describe how we solve the “aperture problem”, some constraints
from “short-range motion”, and finally describe the details of the
algorithm and its application to real images and stimuli which
induce visual illusions.

2.1 Assumptions
The algorithm relies on the following assumption about the imag-
ing conditions: the time δt between images is small, on the order
of one video frame time (1/30th second). For this time scale,
our algorithm assumes that image displacements are small with
respect to the image size, within a range (δx, δy), but δ can be
larger than 1 or 2 pixels. Unlike long-range motion schemes, such
as minimal mapping [Ull79], the displacements are assumed to be
relatively small compared to the size of the objects in the scene.
This restricted motion situation is called short-range motion. In
this framework, correspondence of image elements is a problem,
so the computation resembles binocular stereo.

During the time between frames, the appearance of a moving
object can change due to its own motion, camera motion, light
source motion, or all three, among other effects [VP87]. However,
when the local brightness variation in the surface albedo is suffi-
ciently large, the errors introduced by these effects are relatively
small.

2.2 Constraints on Motion
We use the following constraints to identify the correct optical
flow: uniqueness, each image point has a unique velocity, and
continuity, surfaces are locally smooth. Physical constraints on
motion limit the spatial variation of the optical flow field. First,
the projected velocity field of a planar patch under arbitrary,
rigid 3-D motion is quadratic in (x, y) [Wax87]. A constraint
that is true under more restrictive conditions is: the projected
velocity field of a planar patch, translating parallel to the image
plane, is constant. This is true in limited cases but it may be
a satisfactory local approximation in many cases [LV89]. Our
algorithm assumes local constancy of the flow, i.e., the variation
of the flow is small with respect to its magnitude. Experiments
with support determined by linear variation have shown relatively
little improvement, and this only at discontinuities.

2.3 The Aperture Problem
In most previous approaches, the extraction of motion data from
varying image brightnesses is complicated by the so-called apa-
ture and correspondence problems [MIL81]. The correspondence
problem arises if motion detection is based on image features
that have to be identified in subsequent images. If the problem
is avoided by continuously registering image brightness changes
not necessarily corresponding to features, the motion signal obtained can still be ambiguous due to the aperture problem. In the formulation of Marr and Ullman [MU81], only the velocity normal to the brightness contour can be recovered; they termed this the aperture problem because a moving edge, seen through a circular aperture, seems to be moving normal to itself, while the transverse component of the velocity is not perceived.

There are, however, two types of aperture problems: The strong aperture problem is an instance of the correspondence problem of finding corresponding features in subsequent images. This problem arises only under the unnatural situation of, for example, a straight black line (no features) with infinite extension (no end markers) or straight line segments seen through a small aperture (end markers not visible). Most papers refer to this situation as the aperture problem. The aperture problem can however be solved under the weaker condition of curved line segments or space-varying brightness along the lines (weak aperture problem). The curvature of lines as a feature for matching can be used only by non-local mechanisms.

When the image within the aperture is not simply a portion of a line, i.e., when the matrix of second partial derivatives of the image, the image Hessian ($E_{ij}$), is not zero, then the velocity can be correctly identified [BSE88]. For example, if the aperture contains a corner, the constraints for each edge form a line in velocity space, and the correct velocity lies on their intersection. The aperture problem is an instance of the correspondence problem; if the window used in matching is large, i.e., the features themselves have large spatial extent, it is likely that there should be some variation in orientation of the included contour and then the aperture problem does not occur. This is the idea behind tracking image corners to find displacements [Nag83].

In the optical flow algorithm of Horn and Schunck [HS81], at each point, there is one equation in two unknowns:

$$u \frac{dE}{dz} + v \frac{dE}{dy} + \frac{dE}{dt} = 0$$

(1)

This definition of the optical flow suffers from the strong aperture problem. In order to make the optical flow computation well-posed, we regularize [BPT87] the solution, adding constraint to the computation, for example, by choosing the smoothest optical flow field fitting the data. This leads to formulations of optical flow which apply the regularizing constraint to compute the smoothest velocity field which matches the data: Horn and Schunck [HS81] developed an area-based formulation for instantaneous motion, and Hildreth [Hil84] described a contour-based method for the movement of zero-crossings.

2.4 Short-Range Motion

When the projected motion of objects is small relative to the image size, we can restrict the search for corresponding points
to small regions in the image. We look for a discrete motion
displacement \( V(z, y) = (u(z, y), v(x, y)) \in (\pm \delta, \pm \delta) \) to minimize:

\[
\int \phi(E_t(z,y), E_{t+\Delta t}(x + u\Delta t, y + v\Delta t)) + \\
\lambda(u_x^2 + u_y^2 + v_x^2 + v_y^2) \, dz \, dy
\]

(2)

where \( \phi \) is a comparison function which measures the pointwise match between two images. Note, this is the same smoothness measure used by Horn and Schunck [HS81]. Even with the small motion assumption, the complexity of this procedure is high; at each point, the number of possible displacements is \((2\delta + 1)^2\). The number of possible vector fields for an \( N = n \times n \) image is:

\[(2\delta + 1)^2N\]  

(3)

Of course, many of these vector fields are far from smooth.

2.5 Local Planarity

We approximate Eq. 2, using the constraint of piecewise planarity. Recall that the projected motion of a planar patch orthogonal to the viewing direction is constant over the projected area of the patch. Choose a patch diameter \( \sigma \), dependent on distance to objects in the scene and their expected size in the image. The magnitude of \( \delta \) depends on the expected velocities of objects in the scene, their distances from the camera, and the time separation \( \Delta t \) between frames. The number of fields examined, using the assumption of piecewise planarity, is enormously reduced:

\[(2\delta + 1)^2\]  

(4)

The approximation, in each overlapping patch \( P(x, y) \) of diameter \( \sigma \), minimizes:

\[
\sum_P \phi(E_t(x,y), E_{t+\Delta t}(x + u\Delta t, y + v\Delta t))
\]

(5)

\( \phi \) is a comparison function, for example, the squared difference of the two images.

2.6 Description of the Algorithm

The algorithm can be described as follows (Fig. 1). Consider a set of layers of processors holding the results of evaluating a comparison function \( \phi \) at different displacements of one image relative to the next, each layer corresponding to a different displacement (displacements can be different in magnitude and direction). Each displacement in turn corresponds to discrete velocity. Each processor collects a vote from processors in a small circular neighborhood, \( P(x, y) \) (typically 11 by 11 pixels), in the same layer: a large vote is expected if the motion field is locally constant (local frontoparallel motion) at that velocity. Next, at each \((z, y)\) position, the processor that has the maximum vote across all layers is selected (non-linear suppression or winner-take-all step). The corresponding \( v(z, y) \) is taken as the velocity of the point \((z, y)\). We identify a displacement only at those locations \((z, y)\) at which the maximum vote is unique; ties result in no motion decision. This scheme is similar to a Connection Machine implementation of the Marr-Poggio cooperative stereo algorithm by Drumheller and Poggio [DP86].

2.7 The algorithm works on real images

The algorithm has been implemented as part of the Vision Machine system [PLG86]. It has been shown to be successful both for synthetic (Fig. 2) and natural images (Fig. 3) using several different types of features or measurements on the brightness data, including edges (both zero-crossings of the Laplacian of Gaussian and Canny's method) (Fig. 4), or brightness data appropriately filtered (for instance, through a center-surround operator) (Fig. 5). Because the optical flow is computed from quantities integrated over patches, the results are robust against the effects of uncorrelated noise.

3 Psychophysics

3.1 Barberpole Illusion

The algorithm is consistent with several perceptual effects [BLP87]. For instance, the algorithm suffers from the "Barberpole Illusion" in the same way humans do. Figure 6 shows that, while the true velocity vectors for a rotating barberpole are strictly horizontal, our algorithm computes a vertical velocity field which is consistent with the illusion. This is also a plausible physical interpretation of the optical flow field in the image. Therefore it is not surprising that other motion algorithms are also consistent with perception [Hil84].
3.2 Non-Rigidity Illusion

We could also successfully simulate a psychophysical observation described recently by Nakayama and Silverman [NS88]. In their paper they addressed the question whether, for solving the aperture problem, early motion measurements are integrated over areas (as suggested by Horn and Schunck [HS81] and our voting scheme) or integrated along contours (as proposed by Hildreth [Hil84]). Their study uses a simple distorted line (sigmoid) moving up and down on an otherwise untextured background (Fig. 7). When the distortion is small (the line is almost straight), the central diagonal section appears to move in an oblique direction, while the straight parts of the line move up and down. This leads to a non-rigid perception of the entire figure. The figure can be made to move more rigidly both perceptually and in the algorithm by (a) increasing the curvature of the sigmoid part of the line or (b) by introducing additional features, on or off the line, that are unambiguously moving up and down. Nakayama and Silverman measured the perceived rigidity as a function of the distance between the features and the inflection point of the sigmoidal line. This was done for both cases where the features were on the line and the case where they were off the line. There results provide a striking though partial confirmation of Hildreth’s scheme of integrating motion constraints along contours since the effect of features on the contour has a much stronger effect than off the contour. But since there is a significant effect also for features off the contour line some integration over areas must occur. Our algorithm agrees with the psychophysical observations (Fig. 8 and Fig. 9). Note, however, that the algorithm may be in this case more robust than human

1Increasing the curvature is equivalent to weakening the weak aperture problem; see also Barberpole Illusion.
perception, since it shows the illusion on non-rigidity only for a small range of parameter values, corresponding to a low-precision - or noisy - summation.

3.3 Motion Capture Illusion
Figure 10 shows the perceptual illusion termed "Motion Capture" [RI85]. If a moving sine-wave grating is superimposed on a pair of alternating and uncorrelated random-dot patterns, most of the dots in the display appear to move as a uniform sheet in synchrony with the sine wave grating. The algorithm shows the same effect (Fig. 11), in this simple case, without using any of the "tricks" suggested by Ramachandran[RA86].

3.4 Plaid Motion
Motion of a "plaid" pattern (Fig. 12) generated as the superposition of two sinusoidal gratings with orthogonal orientation is also correctly recovered by the algorithm (Fig. 13), in a manner consistent with perception[AM82].

3.5 Wallach’s Aperture Illusion
Wallach showed already in 1935[Wal76] that the perceived direction of a line moving behind an aperture is strongly influenced by the shape of the aperture. If the aperture is horizontal an oblique line seems to move horizontally as long as it intersects the horizontal boundaries of the window. The same line physically moving in the same direction as before is perceived as moving vertically for a vertical aperture. We tested this illusion with a modification of Wallach’s experiment. Instead of using a single line as the stimulus we used an oblique oriented sine-wave grating in order to get a denser motion field. Our algorithm is consistent with Wallach’s description. It finds a vertical motion field if the grating moves horizontally behind a vertical slit (Fig. 14a) and vice versa, a horizontal motion field when the slit is horizontal (Fig. 14b).
Figure 11: The algorithm produces an output consistent with the perception of "motion capture". The algorithm shows motion capture simply because the motion at each pixel is estimated from the motion of neighboring pixels.

Figure 12: A "plaid" pattern generated as the superposition of two orthogonal gratings moving rigidly downwards and to the right.

4 Physiology

It is natural to ask whether some version of this algorithm may have a natural implementation in terms of cortical physiology. There are psychophysical data suggesting that motion computation is done in two stages, with the first stage computing the perpendicular components of motion and the second stage combining these measurements over an extended area into a coherent motion pattern [AM82]. There is also physiological evidence for a two-stage motion computation in primate visual cortex. Movshon et al. [MAGN85] found that motion sensitive neurons in area V1 could only compute the component of motion in the direction perpendicular to the orientation of image features. These neurons only responded to one component of two superimposed sine wave gratings moving in different directions (same stimulus as in Adelson and Movshon [AM82]). In area MT, however, cells have been found that appear to respond to the direction of motion of the combined pattern (pattern cells). These psychophysical and physiological observations suggest a physiological implementation of the parallel network of Fig. 1. The first step of the algorithm – the matching stage – may be implemented by cortical motion detectors such as the direction selective cortical cells in area V1. The underlying circuitry is an open question, though it could be similar to the one proposed by Poggio and Poggio for disparity sensitive cells and based on silent inhibition [PP84]. The voting stage may be carried on by a set of neurons with a large receptive field that collects information from direction selective cells in V1 and may correspond to the voting neighborhood of Fig. 1. These neurons are tentatively identified with neurons in MT. The last step – the winner-take-all step – could be implemented by inhibitory connections between neurons in MT tuned to different velocities but with the same receptive field location. Our algorithm has a much simpler and more natural physiological implementations than other motion schemes recently discussed [HKLM88, YG88]. The algorithm can be implemented in a simple, feed-forward network with no iteration (relaxation) or feedback mechanisms.

5 Discussion

The algorithm described here suffers from the "aperture problem" only in a mild form. More precisely, it will find a unique optical flow field except in pathological cases (a very long straight contour with no visible terminations or features). The comparison stage is equivalent to patchwise cross-correlation, which exploits local constancy of the optical flow. As matching features, we have used zero-crossings, the Laplacian of Gaussian filtered images, their sign, and smoothed brightness values, with similar results. It is interesting, but not surprising, that methods superficially so different (edge-based and brightness-based) give such similar results, since there are theoretical arguments that support, for instance, the equivalence of cross-correlating the sign bit of the Laplacian filtered image and the Laplacian filtered image itself [LBP87, LBP88].

The algorithm suggests a general point that is of some interest given the several recent papers in the area: phenomena such as motion capture are to be expected by any algorithm that integrates information about motion over local spatial neighborhoods, as standard regularization algorithms do,
such as Horn and Schunck's and a recent scheme of Yuille and Graywacke [YG88].

5.1 Against the Utilitarian Theory of Perception

Different explanations for the motion capture illusion have been proposed by several authors. We would like to comment especially on Ramachandran's utilitarian theory of perception in which he claims that the visual system has developed during evolution a "bag of tricks" for short cuts in visual processing [RA86]. One of such short cuts is visualized by Ramachandran using the example of a leopard leaping from a branch of one tree to a branch of another.

By extracting salient features, such as clusters of dots (the leopard pattern), rather than individual dots, from a complex image and then searching for just the salient features in successive images, Ramachandran proposes that the matching process in motion detection should run much faster because the number of potential matches are reduced. This seems a very plausible argument because motion detection should run fast especially for the dangerous example. This example is also dangerous in another sense because it neglects the fact that biological visual processing runs in parallel rather than serially. The computation time in our algorithm is independent of the number of potential matches because the search process runs in parallel for all image points. The common mistake of thinking that computation time scales with the "size of the problem" reflects our usual way of thinking in serial programming languages.

6 Conclusion

There are several advantages of the method outlined here over differential approaches to motion detection:

- it accommodates longer-range motions
- noise is reduced by the large support regions
- it does not rely on numerical precision of derivatives
- it is therefore more robust

The method uniformly integrates many varying image transformations, and leads to dense optical flow fields. These dense optical flow fields remove the necessity of interpolating or smoothing the output field.

The algorithm by itself is clearly not a full theory of motion perception, which is likely in any case to require more than one single module. In the Vision Machine system the motion algorithm is used in conjunction with later stages that find motion discontinuities, interpolate and smooth the optical flow, and integrate it with other visual modalities [PGL88].

Finally we would like to mention that we are currently working towards a VLSI implementation of the algorithm, using charge-coupled-device (CCD) technology [HLL+88].

7 Acknowledgements

This report describes research done within the Center for Biological Information Processing, in the Department of Brain and Cognitive Sciences, and at the Artificial Intelligence Laboratory. This research is sponsored by a grant from the Office of Naval Research (ONR), Cognitive and Neural Sciences Division; by the Artificial Intelligence Center of Hughes Aircraft Corporation; by the Alfred P. Sloan Foundation; by the National Science Foundation; by the Artificial Intelligence Center of Hughes Aircraft Corporation (S1-801534-2); and by the NATO Scientific Affairs Division (O403/87). Support for the A. I. Laboratory's artificial intelligence research is provided by the Advanced Research Projects Agency of the Department of Defense under Army contract DACA76-85-C-0010, and in part by ONR contract N00014-85-K-0124. Tomaso Poggio is supported by the Uncas and Helen Whitaker Chair at the Massachusetts Institute of Technology, Whitaker College.

References


Werner Reichardt, Tomaso Poggio, and Klaus Hausen. Figure-ground discrimination by relative movement in the visual system of the fly. part II: Towards the neural circuity. *Biological Cybernetics*, 46:1–30, 1983.


