M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 315
Submitted: ICCV '95

Perceptually organized EM: A framework for motion segmentation that combines information about form and motion

Yair Weiss and Edward. H. Adelson
Dept. of Brain and Cognitive Sciences and the Media Lab
MIT E15-384, Cambridge, MA 02139, USA
{yweiss,adelson}@media.mit.edu

Abstract

Recent progress in motion analysis has been achieved with systems that estimate global parameterized motion by integrating multiple constraints. The success of these approaches depends critically on the ability to segment constraints derived from different motions. Hence the problems of motion estimation and segmentation are tightly coupled. We believe it is impossible to solve these problems solely in the motion domain, and that mechanisms of spatial form analysis must be incorporated into the motion estimation procedure.

We present a new framework which allows the incorporation of form information in a graceful manner. It combines concepts from perceptual organization with the powerful optimization technique of EM. We show that the algorithm is guaranteed to decrease a cost function at every iteration, and that in the absence of form information the cost function reduces to the one minimized by EM. We demonstrate that the approach can achieve good motion estimation and segmentation with challenging motion sequences.

Recent progress in motion analysis has been achieved with systems that estimate global parameterized motion [Black and Jepson, 1994, Wang and Adelson, 1994, Hsu et al., 1994, Bergen et al., 1992] These methods have advantages over local optic flow in that they overcome the local ill-posedness of the motion estimation problem by integrating multiple constraints. The sucess of these approaches, however, depends critically on the ability to segment constraints derived from different motions. Hence the problems of motion estimation and segmentation have become tightly coupled.

The joint solution of these problems remains difficult, even for scenes that are very simple. Consider, for example, the scene shown in fig 1(a) (see also [Bergen et al., 1990]). Two bars of different grey shades are moving, one to the left and one to the right. We will consider how several kinds of motion analyses treat this input.

First, the output of a standard least-squares optic flow routine is shown in fig. 1(b), as an arrow plot; the x and y components of velocity are shown in fig. 1(c) and (d) (velocities below some threshold confidence are set to zero, the algorithm is an implementation of Lucas and Kanade (1981) modified according to [Simoncelli et al., 1991]).

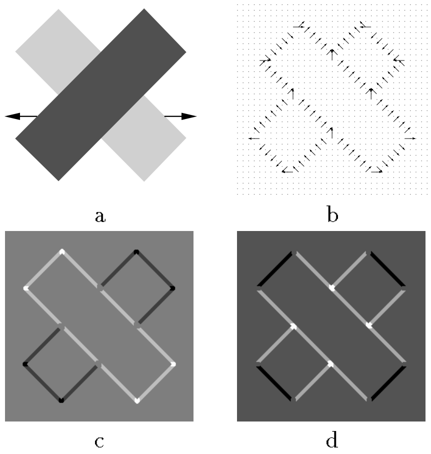

Figure 1: **a** A simple image sequence which causes problems for traditional motion estimation algorithms. **b** Least squares optical flow shown as an arrow plot **c** Least squares optical flow horizontal component. **d** Least squares optical flow vertical component.

Although this sequence is a synthetic one, it illustrates problems that occur frequently in analyzing real sequences.

1. The flow is underconstrained in regions containing extended contours.
2. The T-junctions that occur where one contour crosses the other form spurious features that move with spurious upward velocities; moreover these features are assigned high confidence by standard techniques because they have "good" 2-D structure (the local estimation is overconstrained).
3. The interiors of the bars, being textureless, have no motion information, although one would like them to be filled with the motion assigned their contour. But simply propagating the motion away from the contour will spread it into the exterior as well as the interior. Even propagation along contours is problematic since the spurious T-junction motions will be propagated along with the correct corner motions.
4. The flow field cannot explicitly convey the fact that

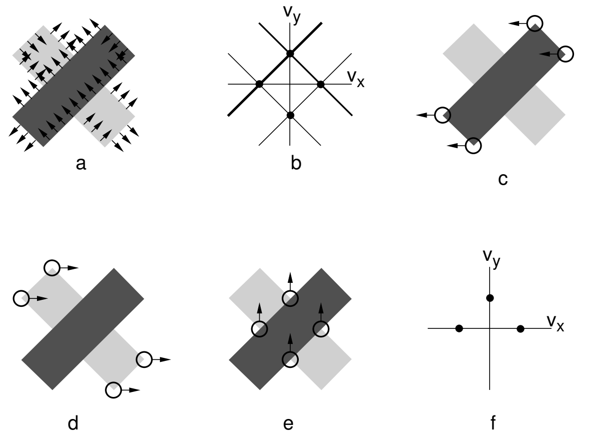

Figure 2: (a) Normal flow (b) Constraint lines in velocity space (c)-(e) Unambiguous feature motion

the two halves of the occluded bar are moving together in "common fate." Indeed there is no information about grouping and segmentation in the flow field representation.

The shortcomings of local analysis can be ameliorated by accumulating constraints over larger regions as in several recent approaches (e.g. [Black and Anandan, 1993]) but figure 2 shows that difficulties persist. The normal flows along the ambiguous contours are shown in fig. 2(a). The constraint lines may be accumulated into velocity space as shown in fig. 2(b). There are four constraint lines, and their thickness corresponds to the number of votes. Clearly there are four major motion candidates, two of which are correct (leftward and rightward), and two of which are incorrect (upward and downward). The spurious upward motion has more votes than any other motion.

As an alternative, one might ignore the ambiguous contour motions and consider only the unambiguous motions of the features. These are regions where the local regression matrix is non-singular. As shown in fig. 2(c), (d), and (e), four such points move to the left, four move to the right, and four move upward. The upward motions are spurious but there is no way to know this by looking at the local regression matrix. In velocity space the feature motions support three of the four motions that were supported by the normal motions including the spurious upward motion.

Another global approach is the iterative nulling technique used by Bergen et al. (1990). In this approach the entire image is warped by a parameterized flow field in an attempt to null one of the motions; the procedure finds the dominant motion, removes it, and proceeds to the next. Success can be verified by aligning two frames and subtracting; the region undergoing the motion should be zeroed out.

Fig. 3 shows the results of nulling with the four candidate motions we have described above. The leftward and rightward motions, in fig. 3(a) and (b), successfully null much of the image. However, the upward motion in fig. 3(c) is even more successful. (The downward motion in fig. 3(d) is less successful). The upward motion finds a large spurious object – the X – and tries to null it as a whole, as if it were

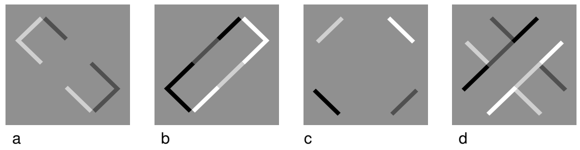

Figure 3: Results of nulling with the four candidate motions described above **a.** leftward **b.** rightward **. c** upward **d.** downward. The upward motion nulls the most pixels.

rigidly translating. The model fails to fully explain the motion of either bar, but the model doesn't know about bars. It only knows about pixels, and it nulls more pixels than any other single motion.

The problems with motion analysis may be summarized as follows. Standard optic flow techniques move from one local representation (e.g. local gradients) to another local representation (a flow field), and thus are limited in their ability to integrate information across an image. Recent techniques use more sophisticated representations and allow more powerful integration of information. However they still fall short of what is needed. A successful approach will need to deal with such issues as occlusions, segmentation, contour ownership, and grouping. That is to say, it is impossible to analyze motion without simultaneously analyzing static form. Issues in perceptual organization are likely to be critical for further progress in motion processing.

Our goal is to introduce a new framework that will make use of recent advances in motion analysis and optimization, and will also allow us to incorporate form information in a graceful manner. The optimization is based on Expectation Maximization (EM) [Dempster et al., 1977], and we combine it with concepts from perceptual organization (PO). We call the new approach POEM. Since current understanding of PO is rapdily evolving, we have designed the POEM to be flexible enough to take advantage of various new PO algorithms as they become available.

1 The Algorithm

1.1 Mixture Models and EM

We begin by reviewing the EM algorithm for mixture estimation briefly.

Mixture estimation refers to the estimation of parameters given data that was generated by multiple processes. In other words, assuming there are K models with parameters θ_k, we measure observations $\{O(r)\}_r$ and estimate (1) the model parameters θ_k and (2) the conditional probability of each process generating each data point, which we will denote, following [Jordan and Jacobs, 1994], by $g_k(r)$. In the case of image segmentation, the $O(r)$ are a set of measurements obtained over the pixel array and $g_k(r)$ offer a soft segmentation assigning each pixel to one or more process.

EM treats mixture estimation as a special case of estimation with incomplete data. The underlying model is that the complete data includes not only $O(r)$ (the "visible data"), but also the "hidden data", labels $L(r)$ specifying which process generated the data (L(r) is a binary vector such that $L_k(r) = 1$ iff process k generated the data at r).

The assumption is that if $L(r)$ were known, the estimation of θ_k would be simple.

The EM algorithm calls for replacing $L(r)$ at each iteration with its conditional expectation (this is the estimation, or E step) based on the current parameter estimates. Since the labels are assumed to be binary vectors, this expectation is merely the calculation of $g_k(r)$:

$$E(L_k(r)|O;\theta) = P(L_k(r)=1|O;\theta) \quad (1)$$
$$= g_k(r) \quad (2)$$

The maximization, or M step uses the current expectation of $L(r)$ to maximize the likelihood of the parameters (since it treats $L(r)$ as known, this step is assumed to be simple), and the algorithm is iterated until convergence. Dempster et al. have shown that each iteration is guaranteed to increase the likelihood of the estimates of θ_k.

As noted by Redner and Walker (84) the attractiveness of the EM algorithm for mixture estimation derives not only from its convergence properties but also from the fact that it often reduces to decoupled intuitive steps. For example if we assume that $\{O(r)\}$ is an IID sample of data generated by adding Gaussian noise of variance σ^2 to the model predictions, the algorithm reduces to the following simple forms:

The E step:

$$g_k(r) = \frac{P(O(r) \cap L_k(r)=1)}{P(O(r))} \quad (3)$$

$$= \frac{\pi_k e^{-D_k^2(r)/\sigma^2}}{\sum_j \pi_j e^{-D_j^2(r)/\sigma^2}} \quad (4)$$

and the M step:

$$\theta_k = \arg\min_\theta \sum_r g_k(r) D^2(r;\theta_k) \quad (5)$$

where we have denoted by $D_k(r) = D(r;\theta_k)$ the deviation of the data at location r from the prediction of model k, and π_k the prior probability of process k.

Note that for equal priors, equation 4 can be rewritten:

$$g(r) = \mathrm{softmin}(D_1^2(r)/\sigma^2, D_2^2(r)/\sigma^2 \ldots) \quad (6)$$

Therefore the algorithm can be characterized as assigning each observation to the process which explains it the best (minimizes deviation from prediction), and then updating model parameters based on these assignments. The parameter σ specifies the "softness" of the partition: for infinitely small σ the softmin function approaches the min function and each data point can be assigned to exactly one process, while for larger σ a datapoint can be assigned to multiple processes. In this case the assignments, $g_k(r)$ serve as weights to the parameter update stage.

These two decoupled steps bear an interesting resemblance to a recently proposed framework for motion segmentation [Hsu et al., 1994] where it was observed that many motion segmentation algorithms can be characterized as iterating the two steps of *segmentation* and *motion modeling*. In the segmentation step elemental areas are assigned to models by measuring deviation from prediction, and in motion modeling motion is estimated using the assignment of the elemental areas.

Thus many of the existing algorithms can be incorporated into the EM framework and viewed as maximum likelihood estimators. Choices of $D(r)$ in the motion domain can be, for example:

- Optic flow constraint [Horn and Schunck, 1981, Simoncelli et al., 1991, Otte and Nagel, 1994]

$$D_k^2(r) = \sum_s \alpha_{rs}(\frac{\partial I}{\partial r} v_k(r) + \frac{\partial I}{\partial t})^2 \quad (7)$$

  where the sum is taken over a window (α_{rs}) centered on location r and the velocity $v_k(r)$ is the predicted velocity at location r of model k.

- Angular deviation from normal velocity [Jepson and Black, 1993]:

$$D_k(r) = \frac{\nabla I(r) v_k(r)}{\|\nabla I(r)\| \|v_k(r)\|} \quad (8)$$

  (here $v_k(r)$ is assumed to be a three-vector in space time)

- Deviation from constant intensity [Lucas and Kanade, 1981]

$$D_k^2(r) = \sum_s \alpha_{rs}(I(s+v_k(r), t+\delta t) - I(s,t)).^2 \quad (9)$$

  Again the sum is taken over a local window around r.

The choice of deviation constrains the M step, of course. The first choice (eq. 7) has the advantage that it is quadratic in $v(r)$ and hence for any motion model where the velocity field is assumed to be a sum of a small number of basis functions (e.g. global translation, affine motion, or rigid planar motion [Adiv, 1985]) the M step involves solving a low rank linear system (see appendix). Jepson and Black (1993) obtained a closed form M step using the second choice for global translation models. Finally the last choice does not have a closed form M step but the M step can be performed through successive linearizations [Lucas and Kanade, 1981]. This final example illustrates the advantage of the EM algorithm: since it decouples the motion estimation and segmentation problems, algorithms developed for estimating a single motion can be directly applied to the estimation of multiple motions in the M step; the only modification needed is to weigh the constraints according to $g_k(r)$.

In motion segmentation it is also desirable to include an outlier model, since observations obtained at occlusion boundaries, for example, will not be well explained by any model. This is done by including an extra model, whose parameters are fixed and whose deviation from prediction is a constant: $D_{K+1}(r) = T$. Thus locations for which deviations from all models are above T will not have significant weight in the parameter update stage. This is equivalent to using a robust measure of deviation D [Black and Rangarajan, 1994]. Thus approaches which recursively estimate the dominant motion [Black and Anandan, 1993, Irani and Peleg, 1992, Bergen et al., 1990] can also be incorporated into the EM framework where the mixture is assumed to be composed of one model ($K=1$) and an outlier model.

To summarize, the EM algorithm defined by equations 4 and 5, which we will refer to hereafter as the classical EM algorithm, is an attractive framework for unifying many motion estimation and segmentation algorithms. However, as we show below, it is based on an assumption of independence which is neither plausible nor helpful in the case of motion segmentation.

The E step in equation 4 depends on two assumptions of independence. The first is that the Gaussian noise added in data generation is independent across space. The second, and much less reasonable assumption is that the "hidden" label variables are also independent. As pointed out by Baum (1977) , assuming a simple Markov dependence between the hidden variables complicates the E step significantly. In one dimension, an efficient forward-backward procedure known as the Baum-Welch algorithm can be used. Although this algorithm has proven tremendously successful in applications such as speech recognition (cf [Rabiner, 1989]) it can not be applied to cases such as image segmentation when the labels are most naturally viewed as lying on a two-dimensional grid. Suggestions for estimating $g_k(r)$ in the multidimensional case include Monte Carlo methods or approximating it by a simpler function [Comer and Delp, 1994, Zhang et al., 1994].

Although the assumption of $L(r)$ independence simplifies the calculations, it is obviously misplaced in the case of motion segmentation. It amounts to the assumption that knowing the membership of a particular location yields no information on the membership of all other locations in the image. In image formation, this is rarely the case: e.g. neighboring points with the same intensity are likely to be from the same object. In the next section we describe a modification of the EM algorithm which can take advantage of such information. For convenience, we will assume equal priors ($\pi_k = 1/K$) and unit variance in what follows.

1.2 The Perceptually Organized EM algorithm

The POEM algorithm receives as input not only the measured observations $O(r)$ but also a measure of expected grouping $w(r,s)$ which is assumed to be extracted by static form analysis. $w(r,s)$ can be any real valued symmetric function $w(r,s) = w(s,r)$ where positive values indicate that location r and location s are likely to belong to the same process, $w(r,s) = 0$ indicates that no information on the membership of r is gained by knowing membership of s and $w(r,s) < 0$ indicates that the two locations are unlikely to belong to the same process. As a simple example one could express the fact that neighboring measurements are likely to belong together by using the function $w(r,s) = e^{\|r-s\|^2}$.

The perceptually organized estimation (or POE) step calls for collecting "votes",$V_k(r)$ from other locations before updating $g_k(r)$:

$$V_k(r) = \sum_s g_k(s) w(r,s) \tag{10}$$

This vote is then combined with the local deviation from model predictions to yield:

$$g(r) = \mathrm{softmin}(D_1(r) - \eta V_1(r), D_2(r) - \eta V_2(r), ...) \tag{11}$$

The M step remains as before.

Before deriving the modified algorithm, we point out a heuristic justification for equation 11. Note that this equation can be rewritten:

$$g_k(r) = \frac{\hat{\pi}_k e^{-D_k(r)}}{\sum_j \hat{\pi}_j e^{-D_j(r)}} \tag{12}$$

With:

$$\hat{\pi}_k(r) = \frac{e^{V_k(r)}}{\sum_j e^{V_j(r)}} \Rightarrow \tag{13}$$

$$\hat{\pi}(r) = \mathrm{softmax}(\eta V_1(r), \eta V_2(r) ...) \tag{14}$$

Thus before taking into account the data at location r we calculate an estimated prior probability, $\hat{\pi}_k$ of that observation being generated by process k based on the votes of other location. The parameter η determines the "softness" of the voting combination: for infinitely large η, if one model receives more votes than any of the others then its prior will be one and all others zero. For zero η all processes will have equal priors regardless of the voting results. These calculated priors are then combined with the data in determining $g_k(r)$ just as in equation 4.

1.3 Derivation of Algorithm

To derive the POEM algorithm we make use of the recent results regarding the equivalence between mixture estimation and optimization based on statistical physics [Yuille et al., 1994, Neal and Hinton, 1993]. As shown by these authors, the classical EM algorithm described above can equivalently be thought of as minimizing the following effective energy:

$$E_{eff}(\theta, g; O) = \sum_{r,k} g_k(r) D_k^2(r) + \sigma^2 \sum_{r,k} g_k(r) \log g_k(r) \tag{15}$$

subject to the constraint that $g_k(r)$ sum to one for all r. Minimizing this effective energy with respect to $g(r)$ while holding θ constant gives the E step, while minimizing with respect to θ while holding $g(r)$ constant gives the M step. (The proof of this statement is given in the appendix). Note that this energy function seems like a reasonable thing to minimize regardless of the maximum likelihood justification for the EM algorithm. The first term penalizes for deviation from model prediction, but the penalty is "gated" by $g_k(r)$: if datapoint r was not generated by model k than there is no penalty for deviation there. The second term penalizes for the entropy of the distributions $g(r)$, i.e. it prefers soft partitions over those where each datapoint is assigned to exactly one process. In this formulation, the parameter σ corresponds to the temperature in statistical physics. Indeed, Yuille et. al (1994) have observed that gradually decreasing σ improves performance.

To obtain the POEM algorithm we add an extra term to the effective energy:

$$\begin{aligned} E(\theta, g; O) &= \sum_{r,k} g_k(r) D_k^2(r) + \sigma^2 \sum_{r,k} g_k(r) \log g_k(r) \\ &\quad - \eta \sum_{r,k} \sum_{s \neq r} w(r,s) g_k(r) g_k(s) \end{aligned} \tag{16}$$

The extra term rewards coherence of the gating parameters for those locations which are likely to belong to the same group. Minimizing this energy with respect to $g(r)$ while holding all other parameters fixed gives the POE step, and minimizing with respect to θ_k while holding all other parameters constant gives the M step. Thus the POEM algorithm is guaranteed to decrease the effective energy in equation 16 at each iteration (see appendix).

For fixed θ, O the efficient energy 16 is similar to the Ising energy of a magnetic material. In fact for $K = 1$ the POE step (eq. 11) exactly describes the dynamics of a deterministic Hopfield network [Hopfield, 1984] which has also been suggested for segmentation use [Geiger and Girosi, 1991, Poggio et al., 1985, Geiger and Yuille, 1991]. To avoid confusion we note that in the above references the Ising potential enforces coherence of the "line processes", which are either present or absent at each location. Here, the Ising term enforces coherence of the regions of grouping. Darrell and Pentland have noted the insufficiency of line processes based approaches to dealing with scenes involving complicated occlusions [Darrell and Pentland, 1991].

2 Implementation

In this section we give examples of the POEM algorithm where the static form constraints given as input are simply based on local intensity and distance in the image (see also [Murray and Buxton, 1987]). Although we will show in the next section that these static form constraints are too simplistic to deal with some image sequences, they serve as a simple illustration of the algorithm's properties.

We use $w(r, s)$ which decreases as a function of the spatial distance $r - s$ and as a function of the intensity difference $I(r) - I(s)$:

$$w(r, s) = \exp\left(-\frac{\|r - s\|}{\sigma_1^2} - \frac{\|I(r) - I(s)\|}{\sigma_2^2}\right) \qquad (17)$$

If we set σ_1 such that votes only occur for neighboring pixels, this is one of the anisotropic diffusion kernels studied by Perona and Malik (1989). This voting function causes the membership probabilities $g_k(r)$ to diffuse nonisotropically in the image, respecting intensity boundaries. Since motion measurements are in fact obtained over a local spatiotemporal window, the intensity associated with each measurement was taken to be the intensity of the window's center. The motion models were assumed to be affine, and the deviation from model prediction was measured by the optic flow constraint 7. Thus the M step involved solving a rank 6 linear system.

Figure 4 shows the two frames given as input to the algorithm. A hand is rotating in front of a static checkerboard. The result of a standard optical flow algorithm is shown in the bottom of figure 4. The flow is noisy but the leftward motion of the top of the hand can be discerned. Figure 5 shows the results of the classic EM algorithm on the sequence. The two global motions, shown at the top, are correctly estimated; one corresponds to the static checkerboard the other to the rotating hand. However, the segmentation, shown at the bottom is poor. Since the output of the algorithm is a probabilistic segmentation we show the segmentation of the first process (the checkerboard) in fig. 5c by weighing each pixel by the probability of belonging to process one $g_1(r)$. We have clipped pixels for which $g_1(r) < g_2(r)$. It can be seen that the untextured regions

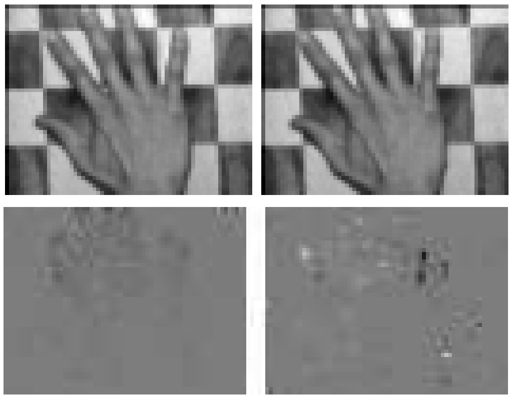

Figure 4: **Top** Two frames given as input to the algorithm. A hand is slowly rotating in front of a static checkerboard **Bottom** Standard optic flow analysis on the two frames (left: horizontal flow, right: vertical flow).

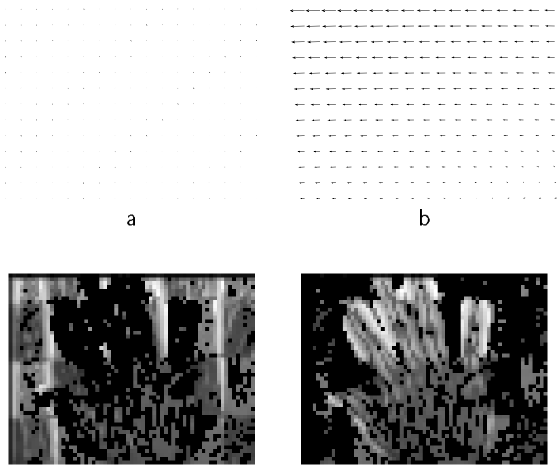

Figure 5: Results with classical EM algorithm. **a.** The flow field for the first process. **b.** Flow field for the second process. **c.** Pixels for which $g_1(r) > g_2(r)$ weighted by $g_1(r)$ **d.** Pixels for which $g_2(r) > g_1(r)$ weighted by $g_2(r)$.

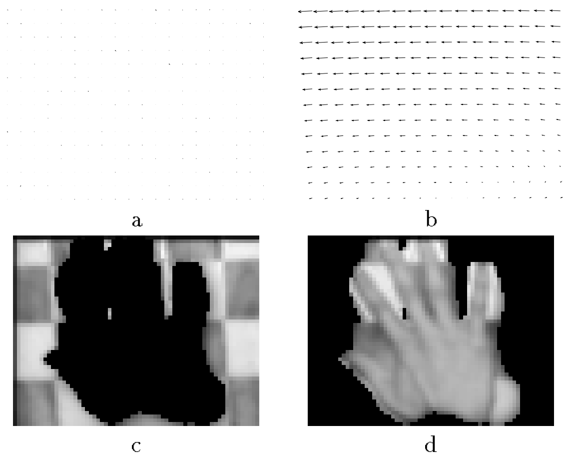

Figure 6: Results with POEM algorithm incorporating only proximity and intensity constraints. **a.** The flow field for the first process. **b.** Flow field for the second process. **c.** Pixels for which $g_1(r) > g_2(r)$ weighted by $g_1(r)$ **d.** Pixels for which $g_2(r) > g_1(r)$ weighted by $g_2(r)$.

are almost randomly assigned to one of the two processes, while the vertical edges of the checkerboard and the hand are assigned with high probability to their respective processes. However, the horizontal edges in the checkerboard are gray: this is a result of the fact that the two motion fields have identical horizontal components, a classic aperture problem effect.

Figure 6 shows the results of the POEM algorithm on the sequence. All parameters are identical to those used in the previous experiment except that the parameter η in equation 11 is nonzero. The flow fields obtained (shown at the top) are almost identical although the flow of the hand has a stronger rotational component. The segmentation is shown at the bottom with the same display format as before. No morphological post-processing was done on the output. The hand and the checkerboard are clearly segmented. The segmentation is of course not perfect: The untextured region between the thumb and the next finger, for example, is filled in incorrectly. This is due to the fact that there is neither a strong intensity gradient nor a conflict in motion information in that region. The space between the two rightmost fingers is not filled in, due to the vertical edge there, whose motion information conflicts with that of the hand. Even a small vertical edge, such as the two fragments between the second and third and third and fourth fingers, are not grouped with the hand, while regions in which there is only horizontal edges are filled in.

In both the classic and POEM runs, convergence was rapid and 10-15 iterations were more than sufficient.

3 Limitations of low-level form constraints

While the voting function used in the previous section improves performance on most sequences, it can not solve image sequences such as the crossed bars depicted in the introduction. The reason is that grouping based on common intensity does not determine contour ownership - the problem of which contour belongs to which region.

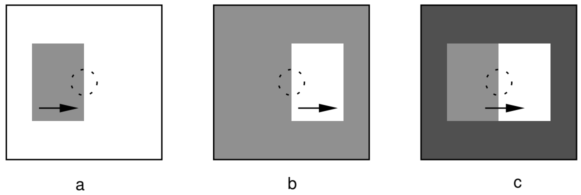

Figure 7: The importance of contour ownership for motion estimation.

Consider the images of figure 7. Suppose that one has correctly estimated the rightward motion of the contour within the region indicated by the dashed circle. The same contour motion could be caused by any of the three configurations shown in fig. 7(a), (b), or (c). In fig. 7(a), a gray rectangle moves against a stationary white background. The contour is owned by the gray region, so the contour's motion should be propagated into the gray rectangle and not into the white background . In fig. 7(b) a white rectangle moves against a stationary gray background. The same contour motion should now be propagated into the white region because the contour ownership is reversed. In fig. 7, a white/gray object moves as a whole against a stationary black background. In this case the contour is owned by both regions, since it is due to a surface marking rather than to occlusion.

We have implemented the POEM algorithm with a rudimentary form of contour ownership. There exist more sophisticated schemes for dealing with perceptual organization [Nitzberg et al., 1993, Sajda and Finkel, 1994, Williams, 1990], but our concern here was to see how well POEM would work when the PO part was quite simple. First, given a static image, we segment it into regions of smooth intensity using a static POEM system. The models are quadratic intensity patches; deviation from prediction is intensity difference, and the spatial weighting function is a Guassian. Next, we run a 3 by 3 window over the segmented image and windows with three labels are marked as containing T-junctions. We collect the number of times any given segment occludes another in the T-junctions and calculate the relative depth of any two segments. Finally we use the segmentation to identify the motion measurements which are contour measurements (those whose local window contain exactly two patches).

Having calculated contour ownership we use the same voting function as in equation 17 but set the "intensity", $I(r)$, of contour measurement according to their ownership. Thus measurements for which ownership is certain have an intensity equal to that of the foreground patch, and measurements for which ownership is uncertain have an intensity which is the average of the two patches and hence will vote equally in both directions. The POEM algorithm is identical to that described in the previous section.

Results of the classical EM algorithm on the sequence discussed in the introduction (fig. 1) are shown in figure 8. The segmentation $(g_1(r))$ is shown on the left, and the estimated optic flow on the right (these flows are generated by taking $v_1(r)$ in regions where $g_1(r) > g_2(r)$ and vice versa, regions where $g_1(r) = g_2(r)$ default to zero velocity). Both the correct interpretation and the spurious interpretation

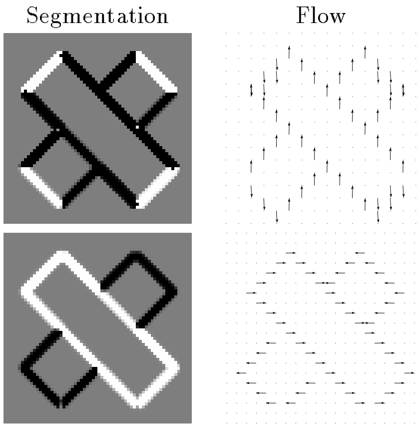

Figure 8: The two interpretations arrived at by the classical EM algorithm randonly on alternate runs. **Top:** The incorrect interpretation. Segmentation on the left and motion field on the right. With the exception of the corners, this interpretation explains all the data. **Bottom:** The correct interpretation. In both cases all untextured regions are ambiguous.

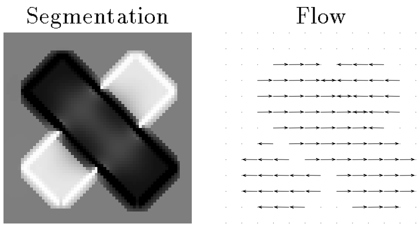

Figure 9: The interpretation arrived at by the modified EM algorithm. The untextured regions are correctly filled in.

are obtained randomly on alternate runs. In both cases, the interior of the bars do not move with the contours.

The results of the POEM algorithm without the use of contour ownership is identical to the bottom of figure 8. Since contour measurements vote for neighboring contour measurements, the correct interpretation is favored. However, since the contour measurements do not vote for either the interior or the exterior of the bars, only the contours are estimated as moving. Compare this with the results of the POEM algorithm where contour ownership is used, which is shown in figure 9 with the same format as figure 8. The correct interpretation is reached and the interiors of the bars do move with the contours while the exterior background does not.

Finally, figure 10 shows another pair of images where contour ownership is essential. A cup occluding a paper punch is captured by a translating camera. The results of classical EM is shown in the top of figure 11 and the results with POEM are shown on the bottom. When contour ownership is used, the motion of the interiors of the objects are correctly estimated. In both cases, the contour between the table and the background is grouped with the paper punch due to common fate.

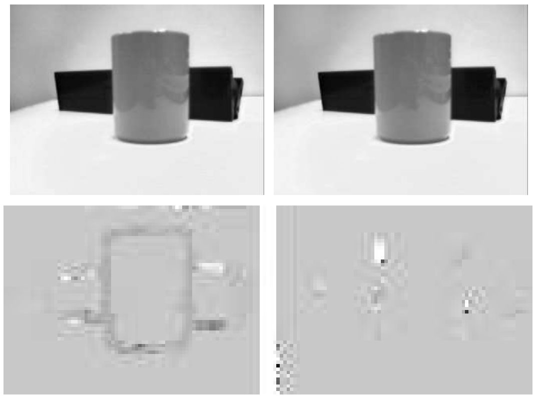

Figure 10: **Top:** Two frames from a sequence in which contour ownership is essential. A cup occluding a paper punch is captured by a translating camera. **Bottom:** Optic flow on these two frames (left- horizonal flow, right - vertical flow).

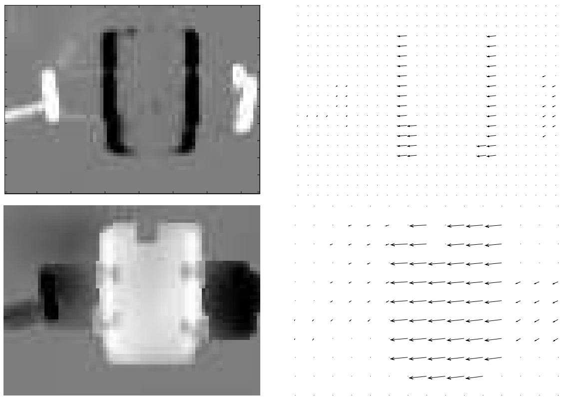

Figure 11: **Top:** The interpretation arrived at by the classical EM algorithm segmentation on the left and motion field on the right. The motions are correct but the interiors of the objects are not moving with their contours. **Bottom:** The interpretation arrrived at by POEM using contour ownership. The interiors of the object move with the contours.

4 Discussion

As various researchers have remarked, for motion analysis to succeed, it must be able to deal with multiple motions. We have argued here that static form constraints are essential in the multiple motion case, and have introduced a framework, perceptually organized EM, whereby these constraints may be utilized by the motion analysis system. The POEM algorithm gives an efficient and intuitive algorithm that unifies many existing algorithms for segmentation and points out their relationship to Hidden Markov Models and approaches based on statistical physics. Our initial experiments demonstrate that even rudimentary perceptual organization cues improve performance significantly on challenging synthetic and real image sequences.

References

[Adiv, 1985] Adiv, G. (1985). Determining three-dimensional motion and structure from optical flow generated by several moving objects. *IEEE Trans. PAMI*, 7(4):384–401.

[Baum, 1977] Baum, L. E. (1977). A comment on Dempster et al.'s EM algorithm. *J. R. Statist. Soc. B*, 39:28–29.

[Bergen et al., 1992] Bergen, J., Anandan, P., Hana, K., and al, R. H. (1992). Hierarchial model-based motion estimation. In *Proc. Second European Conf. on Comput. Vision*, pages 237–252, Santa Margherita Ligure, Italy.

[Bergen et al., 1990] Bergen, J., Burt, P., Hingorini, R., and Peleg, S. (1990). Computing two motions from three frames. In *Proc. Third Int'l Conf. Comput. Vision*, pages 27–32, Osaka, Japan.

[Black and Anandan, 1993] Black, M. J. and Anandan, P. (1993). The robust estimation of multiple motions: affine and piecewise smooth fields. Technical Report spl-93-092, Xerox PARC.

[Black and Jepson, 1994] Black, M. J. and Jepson, A. (1994). Estimating multiple independent motions in segmented images using parametric models with local deformations. In *IEEE Workshop on Motion of Non-Rigid and Articulated Objects*. (in press).

[Black and Rangarajan, 1994] Black, M. J. and Rangarajan, A. (1994). The outlier process: Unifying line processes and robust statistics. In *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, pages 15–22, Seattle, Washington.

[Comer and Delp, 1994] Comer, M. and Delp, E. (1994). Parameter estimation and segmenation of noisy or textured images using the EM algorithm and MPM estimation. In *Proceedings of ICIP*, pages 650–653, Austin, Texas.

[Darrell and Pentland, 1991] Darrell, T. and Pentland, A. (1991). Robust estimation of a multi-layered motion representation. In *Proc. IEEE Workshop on Visual Motion*, pages 173–178, Princeton, New Jersey.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. R. Statist. Soc. B*, 39:1–38.

[Geiger and Girosi, 1991] Geiger, D. and Girosi, F. (1991). Parallel and deterministic algorithms from MRFs: surface reconstruction. *IEEE Trans. PAMI*, 13(5):401–412.

[Geiger and Yuille, 1991] Geiger, D. and Yuille, A. (1991). A common framework for image segmentation. *Int'l J. Comput. Vision*, 6(3):227–243.

[Hopfield, 1984] Hopfield, J. (1984). Neurons with graded responses have collective computational properties like those of two-state neurons. In *Proceedings of the National Academy of Sciences*, volume 81, pages 3088–3092.

[Horn and Schunck, 1981] Horn, B. K. P. and Schunck, B. G. (1981). Determining optical flow. *Artif. Intell.*, 17(1–3):185–203.

[Hsu et al., 1994] Hsu, S., Anandan, P., and Peleg, S. (1994). Accurate computation of optical flow by using layered motion representation. In *Proc. 12th Int'l Conf. Pattern Recog.*

[Irani and Peleg, 1992] Irani, M. and Peleg, S. (1992). Image sequence enhancement using multiple motions analysis. In *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, pages 216–221, Champaign, Illinois.

[Jepson and Black, 1993] Jepson, A. and Black, M. J. (1993). Mixture models for optical flow computation. In *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, pages 760–761, New York.

[Jordan and Jacobs, 1994] Jordan, M. I. and Jacobs, R. A. (1994). Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214.

[Lucas and Kanade, 1981] Lucas, B. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *Image Understanding Workshop*, pages 121–130.

[Murray and Buxton, 1987] Murray, D. and Buxton, B. (1987). scene segmentation from visual motion using global optimization. *IEEE Trans. PAMI*, 9(2):220–228.

[Neal and Hinton, 1993] Neal, R. and Hinton, G. (1993). A new view of the EM algorithm that justifies incremental and other variants. *Biometrika*. submitted.

[Nitzberg et al., 1993] Nitzberg, M., Mumford, D., and Shiota, S. (1993). *Lecture Notes in Computer Science: Filtering, Segmentation and Depth*, volume 622. Springer-Verlag.

[Otte and Nagel, 1994] Otte, M. and Nagel, H. (1994). Optical flow estimation: advances and comparisons. In *Proc. Fourth European Conference on Computer Vision*, pages 51–60.

[Perona and Malik, 1989] Perona, P. and Malik, J. (1989). Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. PAMI*, 8(5):565–593.

[Poggio et al., 1985] Poggio, T., Torre, V., and Koch, C. (1985). Computational vision and regularization theory. *Nature*, 317:314–319.

[Rabiner, 1989] Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77(2):257–286.

[Radner and Walker, 1994] Radner, R. A. and Walker, H. (1994). Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26:195–239.

[Sajda and Finkel, 1994] Sajda, P. and Finkel, L. H. (1994). Intermediate-level visual representations and the construction of surface perception. *Journal of Cognitive Neuroscience*.

[Simoncelli et al., 1991] Simoncelli, E., Adelson, E., and Heeger, D. (1991). Probability distributions of optical flow. In *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, pages 310–315.

[Wang and Adelson, 1994] Wang, J. Y. A. and Adelson, E. H. (1994). Representing moving images with layers. *IEEE Transactions on Image Processing Special Issue: Image Sequence Compression*, 3(5):625–638.

[Williams, 1990] Williams, L. R. (1990). Perceptual organization of occluding contours. In *Proc. Third Int'l Conf. Comput. Vision*, pages 133–137, Osaka, Japan.

[Yuille et al., 1994] Yuille, A., Stolorz, P., and Ultans, J. (1994). Statistical physics, mixtures of distributions and the EM algorithm. *Neural Computation*, 6:334–340.

[Zhang et al., 1994] Zhang, J., Modestino, W., and Langan, D. (1994). Maximum-likelihood parameter estimation for unsupervised model-based image segmentation. *IEEE Transactions on Image Processing*, 3(4):404–420.

A Proofs

Claim: Let:

$$E_{eff}(\theta, g; O) = \sum_{r,k} g_k(r) D_k^2(r) + \sigma^2 \sum_{r,k} g_k(r) \log g_k(r) \tag{18}$$

Then minimizing this effective energy with respect to $g(r)$ subject to the constraint that $\sum_k g_k(r) = 1$ for all r gives the classical E step, and minimizing with respect to θ gives the classical M step.

Proof: (This was proven by Neal and Hinton (1993) in a more general form, we give an additional proof here for compeleteness). Since θ only figures in the first term, the M step is by definition the minimization of this energy with respect to θ while holding g fixed. To obtain the E step note that ading a Lagrange multiplier to the effective energy and setting the partial derivative with respect to $g_k(r)$ equal to zero gives:

$$\log g_k(r) = -D_k^2(r)/\sigma^2 - \lambda \tag{19}$$

These K equations at each location combined with the constraint that the g_k sum to one give:

$$g_k(r) = \frac{e^{-D_k^2(r)/\sigma^2}}{\sum_j e^{-D_j^2(r)/\sigma^2}} \tag{20}$$

Which is precisely the E step with equal priors. The E step with unequal priors, may be obtained by adding the cross entropy between the gating parameters and the priors to to the effective energy:

$$\begin{aligned} E(\theta, g; O) &= \sum_{r,k} g_k(r) D_k^2(r) + \sigma^2 \sum_{r,k} g_k(r) \log g_k(r) \\ &\quad + \sigma^2 \sum_{r,k} g_k(r) \log \pi_k \end{aligned} \tag{21}$$

Claim: Let:

$$\begin{aligned} E(\theta, g; O) &= \sum_{r,k} g_k(r) D_k^2(r) + \sigma^2 \sum_{r,k} g_k(r) \log g_k(r) \\ &\quad - \eta \sum_{r,k} \sum_{s \neq r} w(r,s) g_k(r) g_k(s) \end{aligned} \tag{22}$$

Then minimizing this effective energy with respect to $g(r)$ subject to the constraint that $\sum_k g_k(r) = 1$ for all r gives the POE step, and minimizing with respect to θ gives the M step.

Proof: The proof is identical to the previous one. Adding a Lagrange multiplier to the effective energy and setting the partial derivative with respect to $g_k(r)$ equal to zero gives:

$$\log g_k(r) = -\frac{D_k^2(r)}{\sigma^2} + \frac{\eta}{\sigma^2} \sum_s w_{rs} g_k(s) - \lambda \tag{23}$$

$$= -\frac{D_k^2(r)}{\sigma^2} + \frac{\eta}{\sigma^2} V_k(r) - \lambda \tag{24}$$

These K equations at each location combined with the constraint that the g_k sum to one give:

$$g_k(r) = \frac{e^{(-D_k^2(r) + V_k(r))/\sigma^2}}{\sum_j e^{(-D_j^2(r) + V_j(r))/\sigma^2}} \tag{25}$$

Which is precisely the POE step.

Claim: The POEM algorithm decreases the effective energy at each iteration.

Proof: This follows trivially from the fact that at each stage we are performing a minimization. Formally let $\tilde{\theta}, \tilde{g}$ be the current estimates of the parameters. And let:

$$g(r) = \arg\min_{g} E(\tilde{\theta}, g; O) \tag{26}$$

then by definition, $E(\tilde{\theta}, g; O) \leq E(\tilde{\theta}, \tilde{g}; O)$. The proof for the M step is identical.

Claim: Let:

$$D_k^2(r) = \sum_s \alpha_{rs}(\frac{\partial I}{\partial r} v_k(r) + \frac{\partial I}{\partial t})^2 \tag{27}$$

and let $v_k(r)$ be expressible as the sum of N basis functions, then the M step involves solving an N by N system of linear equations.

Proof: The velocity at any location can be written $v_k(r) = \Psi(r)\theta(r)$ where $\Psi(r)$ is a 2 by N matrix which gives the two components of the basis functions at each location. The partial derivative of E_{eff} with respect to $v_k(r)$ gives:

$$\begin{aligned} c@cc\frac{\partial E_{eff}}{\partial v_k(r)} &= g_k(r)\frac{\partial D_k^2(r)}{\partial v_k(r)} && (28) \\ &= g_k(r)(M(r)v(r) + b(r)) && (29) \end{aligned}$$

With:

$$M(r) = \sum_s 2\alpha_{rs} \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} \tag{30}$$

and:

$$b(r) = \sum_s 2\alpha_{rs} \begin{pmatrix} I_x I_t \\ I_y I_t \end{pmatrix} \tag{31}$$

And using the chain rule, gives:

$$\begin{aligned} \frac{\partial E_{eff}}{\partial \theta_k} &= \left(\sum_r g_k(r)(\Psi^t(r)M(r)\Psi(r) \right) \theta \\ &\quad + \left(\sum_r g_k(r)\Psi^t(r)b(r) \right) && (32) \end{aligned}$$

Which gives an N by N linear system of equation for θ_k.