In the Eighth IEEE International Conference on Computer Vision, July 9–12, 2001, Vancouver, Canada.

Region Extraction from Multiple Images

Hiroshi Ishikawa

Courant Institute of Mathematical Sciences
New York University
New York, NY 10012, U.S.A.
http://cs.nyu.edu/ishikawa/

Ian H. Jermyn

Projet Ariana
INRIA, B. P. 93
06902 Sophia Antipolis Cedex, France
http://www-sop.inria.fr/ariana/personnel/Ian.Jermyn/

Abstract

We present a method for region identification in multiple images. A set of regions in different images and the correspondences on their boundaries can be thought of as a boundary in the multi-dimensional space formed by the product of the individual image domains. We minimize an energy functional on the space of such boundaries, thereby identifying simultaneously both the optimal regions in each image and the optimal correspondences on their boundaries. We use a ratio form for the energy functional, thus enabling the global minimization of the energy functional using a polynomial time graph algorithm, among other desirable properties. We choose a simple form for this energy that favours boundaries that lie on high intensity gradients in each image, while encouraging correspondences between boundaries in different images that match intensity values. The latter tendency is weighted by a novel heuristic energy that encourages the boundaries to lie on disparity or optical flow discontinuities, although no dense optical flow or disparity map is computed.

1. Introduction

Sets of images of a scene formed by varying camera parameters such as position, orientation or time of image formation contain more information about the scene than single images. This information, for example the distance to or motion of points in the scene, is contained in the correspondence between the image domains defined by the points in the scene that are visible in more than one image in the set.

If the camera parameters do not vary too much from one image to another, the correspondence will be ‘dense’: most points in each image will correspond to a point in at least one other image. Many approaches to such sets of images concentrate on inferring this dense correspondence from the data. The philosophy and the techniques used in finding a dense correspondence are analogous to those used in models that attempt to partition the image domain in a meaningful way. In both cases the objects studied are fields *on* the image domain. For example, a partition of the image domain can be thought of as a field of labels. Optical flow and disparity are other examples of such fields.

Complementary to these approaches to image understanding is a large body of previous work that attempts to identify significant regions in an image directly, using a model of the nature of the region or its boundary. The objects studied in these methods are typically mappings of some structure *into* the image domain. Examples of this type of approach are template-based methods and active contours. In both cases a model of the type of region (or its boundary) is established, and the region or regions of the image that best conform to that model are sought. In the active contour case, the standard model favours regions with shorter boundaries passing through points in the image domain with higher intensity gradient for example. While there has been a great deal of effort made to utilise multiple image information in the first approach, the body of work that uses such cues in this second approach is less voluminous.

Each approach has its advantages and drawbacks. Field-based approaches provide information about the whole image, whereas region identification methods provide information about the extracted region only. On the other hand, the latter are clearly well suited to situations in which interest centres on the region identified, and where extra information about the rest of the image would be redundant. An example is the approach of a fast-moving object. In such a situation, computing the distance to or motion of every point in the rest of the scene is inefficient. What is needed is both detection and recognition of the object along with the extraction of the pertinent information.

The present paper is a contribution to this second approach. We present a method for region identification in multiple images. A set of regions in different images and the correspondences on their boundaries can be thought of as a boundary in the multi-dimensional space formed by the product of the individual image domains, so that minimizing an energy functional on the space of such boundaries identifies simultaneously both the optimal regions in each image and the optimal correspondences on their boundaries. By using the ratio form of energy functional described in [16], we enable the global minimization of the energy functional in polynomial time using a minimum ratio cycle graph algorithm, as well as a number of other desirable properties. The energy will incorporate information about the boundaries and regions in each image individually, and also about the correspondences between the boundaries in different images, thus allowing us to utilise the extra information inherent in having more than one image. We will focus on the case of motion sequences, but everything goes over to the stereo case (in particular the phrase 'optical flow' will stand in for 'optical flow and disparity' where appropriate), the primary difference being that the epipolar constraint reduces the dimensionality of the space we need to consider.

1.1. Previous Work

The application of active contour models to multiple images is discussed in the original active contour paper [19]. The approach taken to motion sequences is to track a contour over time, a method which has been greatly elaborated and extended in subsequent work, for example using particle filters [14]. While tracking has something in common with the approach we describe here, the philosophy is different: its emphasis is on following an already-identified region. Initialization, or in other words region identification, is a secondary concern. In fact, viewed as a method for the identification of regions in multiple images, tracking is lacking. It is a greedy approach, in that a region in a second image for example is identified given the region in the first. In contrast, the approach we describe here is global: all regions and their correspondences are identified simultaneously. In addition, tracking algorithms usually do not find the global minimum even for the greedy problem.

The approach taken to stereo pairs in [19] is different and similar to that described here. An energy term favouring contours parallel to the image planes (or alternatively contours whose shape is the same in each image) is used to couple the contours in the two images. However no attempt is made to match the data at corresponding points or to discover disparity discontinuities. It would be quite possible for the two contours to lie on unrelated parts of the two images (in the sense that the data on one of the contours would bear no resemblance to that on the other) provided the displacement from one contour to the other was constant along the contour. In addition, while the energy terms described here could be incorporated into a linear functional such as that used in [19], the advantages of equation 2.1, principally the global minimizability of the energy, would then be lost. In [9], stereo pairs of contours are tracked through time. The contours in the two stereo images are constrained to deform in related ways in time or are constrained to be affine transformations of one another for example. This is an interesting way of incorporating shape information if one has stereo motion sequences, but we do not consider this case here.

Field-based methods for the computation of dense optical flow are very numerous. Many methods use some variant of an optimization approach in which the energy functional encourages 'similar' features in the images to correspond, while using a regularization term to smooth out noise and to determine the optical flow in regions where the data is non-committal [1, 4, 5, 6, 7, 11, 13, 15, 17, 18, 21, 28]. As already discussed, such an approach is complementary to that pursued here. The results of dense computations could be used as extra data for the type of model considered here, as is done in [24], but that is what we are trying to avoid. In fact the reverse is also true: the results of the computations described here can be used to fix the optical flow and weaken the smoothing on the extracted boundaries.

There is a significant difference between the methods proposed here and those methods dedicated to computing correspondences between pre-existing contours [12]. While it would be possible to apply a single image version of equation 2.1 to each image and then match the results, this would constitute another greedy approach to the problem. Similar considerations apply to feature-based dense stereo computations (for example the matching of grouped edge maps [3, 8, 22, 25]), where features are extracted first and then matched, rather than the optimal features being found along with their optimal correspondence.

The approach known as 'layers' [10] attempts to avoid the problem of oversmoothing endemic to most of the above methods by assuming that the image domains can be partitioned into regions in each of which an affine or projective (although not always [27]) model of the optical flow applies. The difficulty with such methods, as with all partition methods, is that of model selection: in this case the number of regions and their models. In addition, the EM algorithm does not guarantee global optimization and must be initialized.

An attempt to circumvent these restrictions is described in [26], where, citing the above problems and the problem of initialization in contour models, the normalized cut method developed for single images is applied to stereo and motion. In the case of motion for example, a motion sequence is segmented into spatio-temporal volumes. The drawback of this approach, which is a form of clustering, is that it is not easy to incorporate geometry. It is difficult to use boundary information, and the correspondence is un-

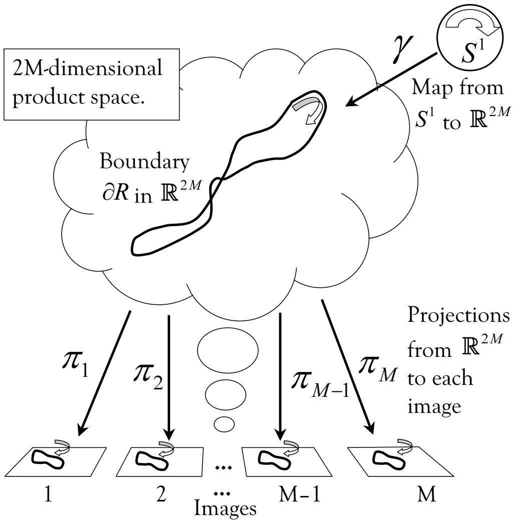
2M-dimensional product space.
γ
S^1
Map from S^1 to $\mathbb{R}^{2M}$
Boundary ∂R in $\mathbb{R}^{2M}$
Projections from $\mathbb{R}^{2M}$ to each image
π_1
π_2
π_{M-1}
π_M
1
2
M-1
M
Images

Figure 1. The geometry of the space of boundaries in multiple images.

known, since the output is a set of pixels with no inherent structure. The model described in the present paper answers the objections raised in [26] to boundary-based methods.

2. Boundaries and Energy Functionals in Multiple Images

In the sequel, we assume that we have M images, indexed by $i \in \{1, .., M\}$. Subscripts on quantities indicate to which image they apply. The intensity functions on the images will be denoted I_i for example. In order to describe regions and boundaries in all images simultaneously, we form the product space of the image domains, which is a domain in $\mathbb{R}^{2M}$. The projections to each image domain will be denoted π_i. By a *boundary* ∂R we mean an oriented closed curve in $\mathbb{R}^n$, which will be represented by a map γ of the circle S^1 into $\mathbb{R}^n$. Several such maps may have the same image ∂R, and hence represent the same geometric information. Our energy functional must and will be invariant to choice of representative.

Given a boundary in $\mathbb{R}^{2M}$ represented by γ, we can project it to each image domain to produce boundaries in each image, $\gamma_i := \pi_i \gamma$. This structure is illustrated in figure 1. The fact that a point (w, x, y, z) lies on the boundary in $\mathbb{R}^4$ for example, says both that the points (w, x) and (y, z) lie on the boundaries in their respective images, and that they correspond. Thus, given an energy functional on the space of boundaries in $\mathbb{R}^{2M}$, we can minimize it and find boundaries in each image simultaneously with their correspondences.

It is necessary to consider such multi-dimensional spaces for we wish to express not only the boundary in each frame but also the correspondences between the boundary and itself in other frames. A boundary moving through time is a function $t \rightsquigarrow \partial R(t)$ living in an infinite-dimensional space. By discretizing time and keeping only a few slices, we arrive at the spaces of boundaries considered here.

The energy functional we will use has the form described in [16]. For the sake of completeness, we review briefly that work here. We start by giving the form of energy functional, and then explain the various quantities involved.

$$E[\partial R] = \frac{N[\partial R]}{D[\partial R]} = \frac{\int_{S^1} dt\, \gamma'(t) \cdot v(\gamma(t))}{\int_{S^1} dt\, |\gamma'(t)|\, g(t)} \qquad (2.1)$$

Here ∂R and γ have already been defined. t is an arbitrary parameterization of S^1. A prime denotes a derivative with respect to t. $\mathbb{R}^{2M}$ has the usual Euclidean metric, denoted $\cdot$. v is a vector field on $\mathbb{R}^{2M}$, and g is a positive function on S^1. Both v and g will typically be derived in some way from the data, or will encode some geometric constraint. Note that N is orientation-dependent, changing sign under a change of orientation of ∂R, whereas D is not. Thus values of E come in positive/negative pairs and minimizing E is the same as maximizing its absolute value.

The functional E has several interesting properties. The first is that for the discrete version of the energy, defined on cycles in directed graphs, there is a polynomial time algorithm, described in section 3, that finds the global minimum for any choice of v and g. We will use this algorithm to find the global minimum of the discrete problem as an approximation to the global minimum of the continuous problem. The second is that the ratio form of the energy ameliorates and in some cases removes the scale dependence inherent in the linear functionals normally used for active contours for example. The third is that, in two dimensions, because the energy is defined on closed curves, we can use Green's theorem to convert integrals over the interior of the curve to integrals over the boundary, and include these in the numerator. Any measure of region optimality that can be expressed as $\int_R f(x, y) \;\; dx\, dy$ can be converted to an integral of the form of the numerator by a simple integration of f. An important example of such a function is $f = 1$, in which case we will favour larger area regions. Many other possibilities have the form of a similarity measure between points in the image and some fixed model. For example, the energy can be made to favour regions of a particular colour, or containing a particular texture.

There are two distinct types of infomation available when we deal with boundaries in multiple images, both of which we will incorporate into the energy 2.1. The first is the information contained about the boundaries, γ_i, in each image individually. Because these boundaries lie in two dimensions, we can include information about the interior regions also, as described above. For example, do the boundaries lie on high intensity gradients, or contain a particular texture? The second is the information contained in the correspondences between the boundaries. For example, are the intensities or intensity gradients similar at corresponding points on the boundaries in each image? We will encode the first type of information in the vector field v on $\mathbb{R}^{2M}$, and the second in the function g. This is appropriate since typically we want to maximize the measure of boundary optimality in each image individually, while minimizing some measure of difference between the boundaries in different images, which is exactly what this assignment of responsibility suggests.

The particular choices that we make in this paper encourage boundaries that lie on points with high intensity gradient normal to the boundary in each image, while also favouring larger areas, and correspondences for which the intensity values at corresponding points are close. In addition we introduce a novel measure that, without computing an optical flow, favours boundaries that lie on discontinuities in the field. We wish to stress however that these are merely the simplest choices, likely to be broadly useful. There are a large number of other possibilities, the choice of which will be determined by the particular application under consideration, or perhaps by higher-level object models. All other choices can be handled in the same way using the same algorithm.

In order to proceed, we must define the vector field v and function g. The choices we make are as follows.

2.1. Numerator

As stated, we will use the numerator of the energy functional to measure the optimality of the projected boundaries *in each image* independently. To do this, we first define a vector field v_i on each image plane. Typically these will all have the same functional form because of symmetry. We then construct a vector field in $\mathbb{R}^{2M}$ by combining the components of these M two-dimensional vector fields into a $2M$-dimensional vector field $v = \langle v_1, v_2, \ldots \rangle$. This combination is done without (potentially present) linear parameters because again symmetry dictates that we treat each image identically. The numerator then becomes

$$\begin{aligned} N[\partial R] &= \int_{S^1} dt\, \gamma'(t) \cdot v(\gamma(t)) \\ &= \sum_i \int_{S^1} dt\, \gamma_i'(t) \cdot v_i(\gamma_i(t)) \end{aligned} \tag{2.2}$$

Thus $N[\partial R]$ is a measure of how optimal a boundary is ∂R when examined in each image independently. In this paper we choose the vector field on each image to be a linear combination of the gradient of the intensity function rotated by $\pi/2$, $(\vec{\nabla}_i I_i)^{\perp}$, and the position vector (x, y) rotated similarly. The latter, when converted to an integral over the region, gives the area of the region. This choice has the effect of favouring boundaries that pass through points in each image with high intensity gradient oriented normal to the boundary and with larger enclosed area. There is a constant of proportionality α between these two terms, which we discuss in section 4.

2.2. Denominator

In [16], a trivial $D[\partial R]$ was used in equation 2.1. Here we exploit the freedom allowed by the energy 2.1, and construct a $D[\partial R]$ that compares the boundaries in the images to each other by comparing intensities at corresponding points. This comparison measure is however weighted by a novel term, likely to be useful beyond the model considered here, that, without computing an optical flow, heuristically favours boundaries that lie on discontinuities in the field.

To proceed, we must therefore describe the positive function g. This function has the following form:

$$g(t) = \frac{g_N(t)}{g_D(t)} \tag{2.3}$$

The value $g_N(t)$ will be a measure of the difference of the intensities at the corresponding points $\gamma_i(t)$ on the boundary in each image. It will be small when the intensities at corresponding points agree. The value $g_D(t)$ is the new weighting term that will tend to be large when the points $\gamma_i(t)$ lie on an optical flow discontinuity. Since g appears in the denominator of E, it will tend to be small for optimal boundaries, thus encouraging boundaries that match similar data and that lie on discontinuities.

Both g_N and g_D compare the corresponding points in consecutive pairs:

$$g_?(t) = \sum_{i=1}^{M-1} g_{i,?}(t) \tag{2.4}$$

where ? stands for N or D, and where $g_{i,?}$ depends on the data at $\gamma_i(t)$ and $\gamma_{i+1}(t)$ only. (Note that consecutive pairwise comparison is not the only possible choice: it is equally easy to compare more than two boundaries at once, to encourage small accelerations for example.) More specifically

$$g_{i,N}(t) = (I_{i+1}(\gamma_{i+1}(t)) - I_i(\gamma_i(t)))^2 \,. \tag{2.5}$$

Note that $g_{i,N}$ is small at a point $t \in S^1$ when the intensities at the projected points $\gamma_i(t)$ and $\gamma_{i+1}(t)$ are similar. Thus the sum of the $g_{i,N}$ encourages good intensity matches between consecutive pairs of corresponding points. To avoid dividing zero by zero, we add a constant β to the value of g_N, which we discuss in section 4.

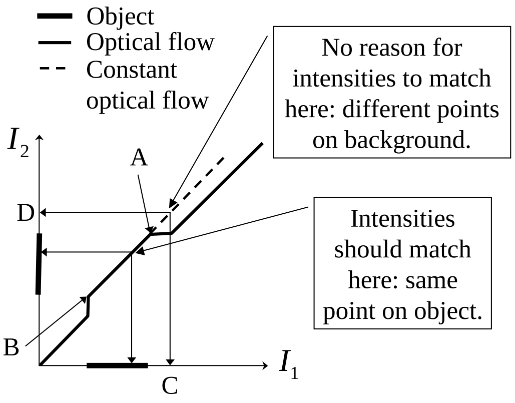
Object
Optical flow
Constant optical flow
I_2
A
D
B
C
I_1
No reason for intensities to match here: different points on background.
Intensities should match here: same point on object.

Figure 2. Two one-dimensional images containing a moving object. The figure illustrates the geometrical meaning of the function g_D.

Before giving the details of the functions $g_{i,D}$, we first give an intuition as to why the form of function we will describe is useful. Consider figure 2. It shows the product space of the image domains of two one-dimensional images, I_1 and I_2. Let us suppose that these images consist of a moving object against a background. The graph shows the true optical flow for such a situation. The role of the boundary in this situation is played by the points A and B in the figure, which project to the boundaries of the object in each image. Take the point A. Now move away from A (the boundary) but in a direction of constant optical flow. (The line of constant optical flow is shown by the diagonal dotted line.) If one moves to the interior of the object, the intensities in each image should match, because they correspond to the same point in the scene. On the other hand, if one moves, still at constant optical flow, to the exterior of the region, there is no longer any reason for the intensities to match because the projected points C and D no longer correspond to the same point in the scene. Thus, if we take the derivative of the intensity difference (squared) normal to the boundary in the direction of constant optical flow, it will tend to be non-zero if the boundary lies on a discontinuity in the true optical flow. This is a heuristic: the intensities outside the boundary may match by chance (if the background is homogeneous for example), but in general this will not be the case. The derivative may also be large in other regions of the plane in figure 2, but in this case the intensities are unlikely to match, so that g_N would be unfavourable.

More precisely: any point in the 4D subspace of $\mathbb{R}^{2M}$ formed by the product of the image planes of the i^{th} and $(i+1)^{\text{th}}$ images defines a two-vector in that space, the difference between the projected points. This in turn defines a plane embedded in the 4D space on which this vector (the optical flow) is constant. The symbol $\vec{\nabla}_{i,2}$ denotes the derivative within this plane. The perpendicular operator $\perp_2$ operates within this plane, rotating any vector in the plane by $\pi/2$. We then define the functions $g_{i,D}$ as follows:

$$g_{i,D}(t) = \left\{ \frac{1}{|\gamma'(t)|}\gamma'(t) \cdot (\vec{\nabla}_{i,2} g_{i,N}(t))^{\perp_2} \right\}^2 . \qquad (2.6)$$

Since $\vec{\nabla}_{i,2} g_{i,N}(t)$ is the gradient of the intensity difference $g_{i,N}$ in the plane of constant optical flow, $g_{i,D}$ is a measure of the component of this gradient normal to the boundary. It is thus large if the intensity difference changes a lot when we take a small step away from the boundary in $\mathbb{R}^{2M}$ in a normal direction, remaining at the same optical flow value. The function $g_{i,D}$ thus captures the intuition explained above.

By summing $g_{i,N}(t)$ and $g_{i,D}(t)$ over consecutive pairs of images and taking their ratio, we therefore have a function $g(t)$ that tends to make the projected points $\gamma_i(t)$ match in intensity and lie on discontinuities in the optical flow. Integrating $g(t)$ over the boundary thus tends to find boundaries with these properties also.

This completes the definition of the energy functional on boundaries in $\mathbb{R}^{2M}$. As we have stressed, with the exception of the above heuristic, which we expect to be useful in any scenario, we have chosen the simplest possible models for the regions and boundaries that we seek. Particular applications can augment or replace these choices for both the numerator and denominator, choosing to look for particular textures for example, or favouring boundaries with high optical flow or small acceleration.

3. Algorithm

In [16], the algorithm that was used to optimize the energy functional was not completely general, solving only a subset of instances of the energy 2.1. The more complex energies we use here necessitate moving beyond this subset, so we take the opportunity to describe a new algorithm, the 'minimum ratio weight cycle' algorithm, that is completely general, in the sense that it applies to any instance of the energy functional in equation 2.1, and is also faster and considerably more space-efficient than the one used in [16].

Consider a directed graph $G = \langle V, E\rangle$ with two edge weights $E \xrightarrow{\lambda} \mathbb{Z}$ and $E \xrightarrow{\tau} \mathbb{Z}^+$. We define $n = \text{card}(V)$, and $m = \text{card}(E)$. Define a weight W on subsets of E by

$$W(C) := \frac{\sum_{e\in C} \lambda(e)}{\sum_{e\in C} \tau(e)} \qquad (3.1)$$

The discrete problem is to find the cycle C^* in G that minimizes W over the space of cycles. We define $W^* = W(C^*)$. The algorithm that solves this problem is called the 'minimum ratio weight cycle' algorithm, described in [20] and generalised and improved in [23]. The algorithm relies on the following, interesting observation. Define a new, parameterized edge weight $E \xrightarrow{w_t} \mathbb{Q} : w_t(e) = \lambda(e) - t\tau(e)$, where $t \in \mathbb{Q}$. We use the same symbol w for the weight of a set of edges (defined by summation). Then the solution, t^*, of the equation $w_t(C^*_t) = 0$, where C^*_t is the minimizing cycle for w_t, is equal to the minimum ratio weight W^* of W, and the minimizing cycle $C^*_{t^*}$ of w_{t^*} is equal to C^*, the minimizing cycle for W. We omit the simple proof.

The problem is now reduced to finding an efficient search strategy for t^*. Although there are a number of approaches, including binary search [20], and a more sophisticated approach using parametric edge weights [23], we find in practice that the fastest method is simply linear search. To see how this works, note that if $t > t^*$, the graph G using the weights w_t will have a negative cycle. We start with a known upper bound t^0 on t^*. We apply a negative cycle detection algorithm (we give no details of this here—we use a variant of a label-correcting algorithm described in [2]) with edge weights w_{t^0}. If we do not find a negative cycle, and the algorithm terminates, there will be a zero weight cycle, and we are done: $t^0 = t^* = W^*$, and the zero weight cycle $C^*_{t^0}$ is the solution C^* required. (Detection of zero weight cycles is easily done. Again we refer the reader to [2] for details.) If a negative cycle C is detected, t^0 is too large. Since $w_{t^0}(C) < 0$, we have that $t^0 > W(C) \geq t^*$. We therefore replace t^0 by $t^1 = W(C)$. The search continues in this fashion until t^* is found. When t^* is found, we have that $W^* = t^*$ and that $C^* = C^*_{t^*}$. C^* is a zero weight cycle and so can easily be detected. Note that the algorithm requires no initialization.

Since the weights λ and τ are integral, t^* is rational, as already stated. This enables a pseudo-polynomial bound to be placed on the search time in the following way. It is easy to see that because of the integrality of the weights, upper and lower bounds on W^* are given by $\pm\lambda_0$, where $\lambda_0 = \max\{\lambda(e) : e \in E\}$. The minimum weight difference between two distinct ratio weights can similarly be proved to be $\frac{1}{\tau_0^2}$, where $\tau_0 = \max\{\tau(e) : e \in E\}$. Thus the maximum number of applications of the negative cycle detection algorithm is $O(\tau_0^2\lambda_0)$. Since the asymptotic complexity of our negative cycle detection algorithm is $O(mn)$, the pseudo-polynomial bound follows. In practice, the negative cycle detection never executes to completion, and the time bound is never saturated. The algorithm requires $O(m)$ space, the bound coming from the negative cycle detection algorithm. Unlike the time bound, this one is saturated.

We construct an instance of this problem starting from the continuous energy as follows. First we define a graph G_i in each image domain. The graphs in each image have as vertex sets the pixels in that image, and as directed edge set the eight neighbours of each pixel. In a manner entirely analogous to the continuous case, we form the product graph $G = \times_{i=1}^{M} G_i$ of the individual image graphs. G is a rectangular lattice embedded in $\mathbb{R}^{2M}$. The edges in G are the product edges, but we eliminate many of them in the following way. First we reduce the overall size of the graph by using only a slice of G around the diagonal. In addition, we reduce the number of edges on a local basis by restricting the magnitude of the "time derivative" of the tangent vector to the boundary. Since we are dealing with a discrete time, this means that we impose a restriction on the difference between the tangent vectors to the boundary at corresponding points in successive frames. Thus if γ'_i and γ'_{i+1} are the projections of the tangent vector to the boundary γ' in two successive frames, we impose that $|\gamma'_{i+1} - \gamma'_i| \leq |\gamma'_i|$. In the discrete case, this becomes a condition on the edges that we allow in the graph. These constraints, which are imposed for reasons of computational efficiency and are not necessary in principle, do however have sensible interpretations. The first says that the inter-frame motion should be less than some threshold. The second amounts to constraining the amount by which the shape of the boundary can change from frame to frame. This is in fact similar in nature to the energy term used in [19] for stereo, where it was imposed as a soft constraint. In the stereo case, the ordering constraint can be used similarly to reduce the number of edges, although the constraint is too stringent as is: it means eliminating all edges that can change the disparity. To alleviate this problem, we added extra edges at each vertex to second neighbours. See figure 3.

For an edge (x, y) in G, we then define the numerator and denominator weights as follows. The vector field v and function g at this edge are defined as the averages of the values at x and y. The tangent vector to the boundary is defined as $y - x$ (recall that x and y are points in $\mathbb{R}^{2M}$). We compute the intensity gradients using a Gaussian derivative spread over a few pixels, and do the same for the gradient of the intensity difference used in the denominator. This gives us all the required data to compute the numerator and denominator of equation 2.1 for each edge. We can then apply the above algorithm to find the minimizing cycle.

4. Demonstrations

In the motion sequences, we applied the model to triples of images in succession, which was the maximum possible within the limits of the memory currently available to us. (The demonstrations were run on a 933MHz Pentium III machine with 1GB of memory.) We iterated the algorithm on each of the examples, after first removing the pixels belonging to the previously identified boundary.

Figure 4 demonstrates the effect of the term g_D. The image sequence consists of the motion of two black circles against a background whose intensity varies linearly with height in the image. At the start of the sequence the left-hand circle is stationary, while the right-hand circles moves up, to the right, and down. Then the right-hand circle is sta-

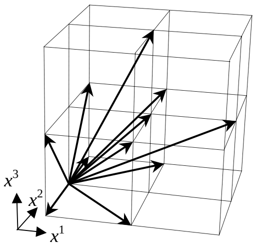

Figure 3. The edge connectivity in the stereo case.

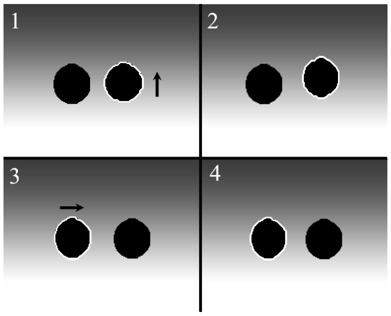

Figure 4. The effect of the term g_D. Four sample frames equally spaced from a sequence consisting of a total of 25 frames.

tionary while the left-hand circle moves to the right. Due to the intensity gradient, g_D is non-zero for the moving circle, while it is zero for the stationary circle. This results in the correct identification of the moving circle over each triple of frames, with the identified boundary switching from the right- to the left-hand circle when the former stops and the latter begins to move. Note that the identification of the right-hand circle occurs in spite of the fact that by moving up in the image this circle acquires a reduced average intensity flow across its boundary, and hence a weaker numerator, $N[\partial R]$. To emphasize this, it is worth noting that when the still image version of the model is applied to these images, it consistently identifies whichever circle is lower in the image.

In figures 5 and 6 are shown the results of the application of the model to several stereo pairs and image sequences. The examples took about 5 to 20 minutes to run on stereo pairs of 135×152, 148×148, and 230×260 pixels and triples of images from motion sequences of 75×50 and 55×70 pixels. The model successfully identifies corresponding regions within each image, and is consistent across consecutive triples of motion sequence images, even though no initialization was used to pass from one triple to the next. Note that although there is a tendency for the boundaries to lie on discontinuities, this tendency can be overruled by other properties of the region model, in this case the tendency to find boundaries lying on high intensity gradients. This is to be expected: a sharp intensity gradient may signal the boundary of an object even though there is no discontinuity present. Both are indicators of such a boundary.

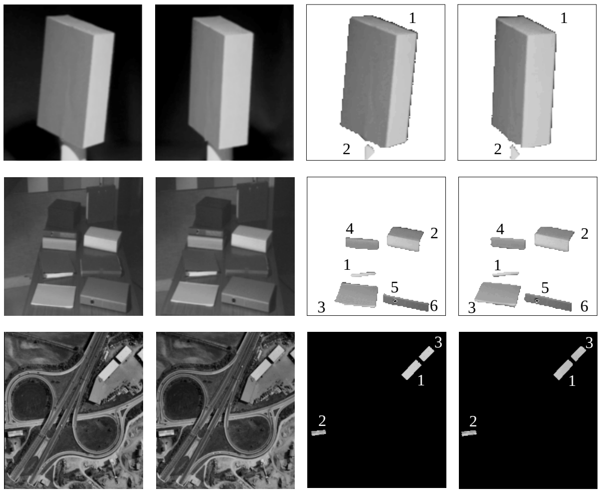

Figure 5. The results, shown on the right, of applying the algorithm to some stereo pairs, shown on the left. The algorithm was iterated on the images, the numbers showing the order in which the regions were found.

5. Conclusions

We have presented a method for the extraction of regions from multiple images simultaneously with their correspondences. This method falls within the category of methods that search for regions in images possessing certain properties directly, without performing dense computations, a particularly relevant example being active contours. The model allows a broad range of possibilities for the description both of regions in the individual images and of their correspondences. For any of these possibilities, the optimal regions and boundary correspondences can be found in polynomial time. We have illustrated the model using the simplest (and hence most likely to be broadly useful) choices for the terms in the energy. In addition, we have introduced an energy term based on a novel heuristic that favours boundaries lying on discontinuities in the disparity or optical flow. This energy should be useful beyond the specific model used in this paper.

The principle drawback of the method lies in the nature of the object models. Object recognition in images is a hard problem, and although equation 2.1 can incorporate a great variety of different information, it is all in the form of summations over the region or boundary. The same criticism applies to active contours and segmentation by partition methods. It is clear that using this type of information

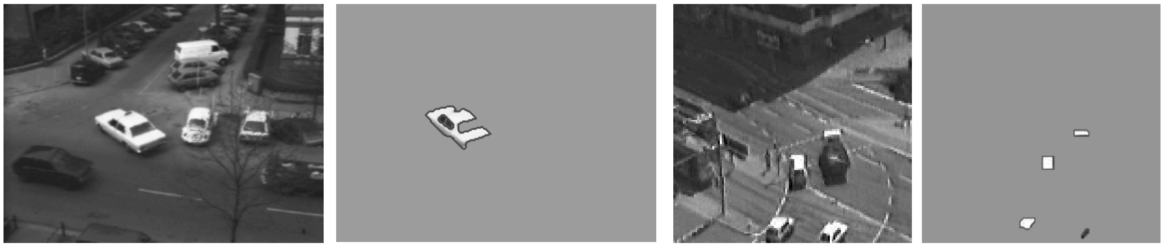

Figure 6. The results of applying the algorithm to some motion sequences.

alone, successful identification of the region occupied by particular objects will remain difficult. The development of higher-order models of regions and boundaries that describe structure rather than average properties is necessary to make more progress.

References

[1] E. H. Adelson and J. Bergen. Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Am.*, 2 (2):284–299, 1985.

[2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms and Applications*, chapter 5, pages 133–165. Prentice Hall, NJ, U.S.A., 1993.

[3] N. Ayache and B. Faverjon. Efficient registration of stereo images by matching graph descriptions of edge segments. *Int'l J. Comp. Vis.*, 1 (2):pp. 107–131, 1987.

[4] H. H. Baker and T. O. Binford. Depth from edge and intensity-based stereo. In *Proc. 7^{th} Int'l J'nt Conf. Artif. Intell.*, pages 631–636, August 1981.

[5] P. N. Belhumeur. A bayesian approach to binocular stereopsis. *Int'l J. Comp. Vis.*, 19(3):237–262, 1996.

[6] M. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proc. IEEE Conf. Comp. Vis. Patt. Rec.*, pages 296–302, 1991.

[7] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *Proc. IEEE Conf. Comp. Vis. Patt. Rec.*, pages 648–655, 1998.

[8] A. T. Brint and M. Brady. Stereo matching of curves. In *Proc. IARP Int'l Workshop on Multi-Sensor Fusion and Environment Modelling*, 1989.

[9] T. J. Cham and R. Cipolla. Stereo coupled active contours. In *Proc. IEEE Conf. Comp. Vis. Patt. Rec*, pages pp. 1094–1099, 1997.

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

[11] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. *Int'l J. Comp. Vis.*, 5:77–104, 1990.

[12] D. Geiger, A. Gupta, L. Costa, and J. Vlontzos. Dynamic programming for detecting, tracking, and matching deformable contours. *IEEE Trans. Patt. Anal. Mach. Intell.*, 17(3):294–302, 1993.

[13] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[14] M. Isard and A. Blake. CONDENSATION—conditional probability propagation for visual tracking. *Int. J. Comp. Vis.*, 29 (1):pp. 5–28, 1998.

[15] H. Ishikawa and D. Geiger. Occlusions, discontinuities and epipolar lines in stereo. In *Proc. 5th Euro. Conf. Comp. Vis*, 1998.

[16] I. H. Jermyn and H. Ishikawa. Globally optimal regions and boundaries. In *Proc. 7th IEEE Int'l Conf. Comp. Vis.*, pages 904–910, Kerkyra, Greece, September 1999.

[17] D. Jones and J. Malik. A computational framework for determining stereo correspondence from a set of spatial filters. In *Proc. Euro. Conf. Comp. Vis.*, Santa Maria Lighure, Italia, 1992.

[18] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiments. In *Proc. Image Understanding Workshop DARPA*, PA, U.S.A., September 1990.

[19] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. J. Comp. Vis.*, 1(4):321–331, 1988.

[20] E. L. Lawler. Optimal cycles in doubly weighted linear graphs. In *Theory of Graphs: International Symposium*, pages 209–213. Gordon and Breach, New York, U.S.A., 1966.

[21] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proc. Roy. Soc. London*, B 204:301–328, 1979.

[22] G. Medioni and R. Nevatia. Segment-based stereo matching. *Computer Vision, Graphics and Image Processing*, 31:2–18, 1985.

[23] N. Meggido. Combinatorial optimization with rational objective functions. *Mathematics of Operations Research*, 4:414–424, 1979.

[24] S. Menet, P. Saint-Marc, and G. B. Medioni. B-snakes: Implementation and application to stereo. In *Proc. DARPA Workshop on Image Understanding*, pages pp. 720–726, Pittsburgh, U.S.A., 1990.

[25] L. Robert and O. Faugeras. Curve-based stereo: Figural continuity and curvature. In *Proc. IEEE Conf. Comp. Vis. Patt. Rec.*, pages pp. 57–62, 1991.

[26] J. Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *Proc. Int'l Conf. Comp. Vis.*, pages 1154–1160, Bombay, India, January 1998.

[27] Y. Weiss. Smoothness in layers: Motion segmentation using non-parametric mixture estimation. In *Proc. IEEE Conf. Comp. Vis. Patt. Rec.*, pages 520–526, 1997.

[28] A. Yuille. Energy functions for early vision and analog networks. *Biological Cybernetics*, 61:115–123, 1989.