M.I.T. Media Lab Vision and Modeling Group Technical Report No. 163, Feb. 1991

Cooperative Robust Estimation Using Layers of Support

Trevor J. Darrell and Alex P. Pentland

Perceptual Computing Group
MIT Media Lab
20 Ames Street E15-388
Cambridge MA 02139
(Revised March 7, 1994)

Abstract

We present an approach to the problem of representing images that contain multiple objects or surfaces. Rather than use an edge-based approach to represent the segmentation of a scene, we propose a multi-layer estimation framework which uses support maps to represent the segmentation of the image into homogeneous chunks. This support-based approach can represent objects that are split into disjoint regions, or have surfaces that are transparently interleaved. Our framework is based on an extension of robust estimation methods which provide a theoretical basis for support-based estimation. The Minimum Description Length principle is used to decide how many support maps to use in describing a particular image. We show results applying this framework to heterogeneous interpolation and segmentation tasks on range and motion imagery.

1 Introduction

Real-world perceptual systems must deal with complicated and cluttered environments. To succeed in such environments, a system must be able to recover salient parameters that are relevant for the task at hand. However many of the techniques developed to estimate these parameters are designed for homogeneous signals, in which all the data comes from a single source that can be described by a single model. They often perform poorly on data which intermingles several sources with different underlying models or model parameters. To apply these techniques to complex environments, we must decompose the heterogeneous sensory signal into its constituent homogeneous chunks either before, or during, the estimation process.

This paper addresses the heterogeneous estimation task, and presents an approach to the problem using a support-based model of segmentation. In our approach, explicit boolean masks, called "support maps" or "support layers", are used to represent the extent of regions in an image. The use of this representation has several advantages over more traditional edge-based segmentation models, since it allows an unrestricted model of how regions may be occluded.

We will assume that we have a set of models capable of describing all regions of the signal, and that we can find the residual error of a particular model over a given region

This work performed under ONR contract N00014-93-J-0172

of the signal. Our method uses the residual error values of several estimates to determine where each estimate is accurately approximating the signal, and from this computes a set of support maps.

The following section motivates the use of support maps as a framework for segmentation, generalizing the concept of support found in the robust estimation literature. We will then discuss the Minimum Description Length (MDL) criteria, which provides a mechanism for deciding how many objects or processes exist in a signal, and thus how many support maps to use in estimating that signal. Finally, we will show results using this approach in several applications for which appropriate image models are available, including shape and motion segmentation tasks.

2 Background

A method for heterogeneous description must have a model of how objects or surfaces are combined in the image, as well as models of the object or surface processes themselves. The issue of how to fully represent the former, the segmentation of a scene, is often neglected in computer vision; in practice, having a expressive segmentation model is as important as having a good model of the underlying data.

Traditionally, edge-based approaches to segmentation have been used. Often a fixed "edge-detector" is run to find the edges of an image as a first stage of processing, which are then used to mark the boundaries of regions from which parameters are estimated. The line process, introduced by Geman and Geman [10] and later expanded upon by [28, 32, 6], merged these two steps into a single regularization/reconstruction framework. This approach simultaneously estimates an interpolated surface, which regularizes the data according to a prior model of surface smoothness, and a discontinuity field, which indicates allowed departures from the smoothness constraint. This discontinuity field, called the "line-process", allows for the successful reconstruction of scenes that contain simple occlusion boundaries, and has been used in several different domains of visual processing.

With complicated occlusion, however, an edge-based segmentation model is inadequate: when an object is split into disconnected regions on the image plane an edge-based segmentation prevents the integration of information across the entire object during the segmentation process. A good example of this type of phenomena is the transparent motion display developed by Husain, Treue and Andersen for psychophysical experimentation [16]. In these examples, dots are placed on an otherwise transparent cylinder which rotates around its major axis. Two populations of intermingled random dots are seen, one corresponding to the foreground surface of the cylinder and the other corresponding to the background. When presented with such transparent displays, an algorithm that extracts motion information using an edge-based segmentation method will not be able to group the two populations of random dots. As the results of Husain et. al. show, human and animal observers have no difficulty in grouping the two motions. The same phenomena can be found with static imagery: an object viewed through a fence or trees may be broken into several disjoint regions on the image plane. The regions of this image could not be grouped together using a line-process or other mechanism that relies solely on an edge map for segmentation.

Because of these limitations, we believe an edge-based segmentation model is insufficient for realistic natural scenes. Instead, we have explored a more descriptive approach in which we explicitly represent the shape of a region using a support map. A support map places no restrictions on the connectivity or shape of a region, so it can handle the cases described above. As we shall see in the next section, the use of a single support map to deal with occlusion and sensor noise has already been developed in the robust statistics literature. However, a single support map can only represent a single object, and we wish to work with scenes with multiple objects and complex occlusions. Our contribution is to extend this robust estimation paradigm to encompass multiple processes, using a different support map for each object in the scene. In our approach, we represent the segmentation of a scene as a set of support maps, each corresponding to a distinct homogeneous (but possibly disconnected) region.

Other authors have proposed approaches to segmentation which go beyond single edge-map representations. Nitzberg and Mumford propose a multi-layer signal representation which they call "The 2.1D sketch", in which the edges of each object occupy a distinct layer [24]. Adelson and Anandan proposed multi-layer representations for the modeling of static transparency [1]. Leclerc [19] has developed a region grouping strategy using MDL theory which can link disjoint regions but is dependent on an initial edge-based description stage to find candidate regions. Marroqiun [22] has extended the Markov Random Field formulation to include a notion of disconnected support, and has shown results using simple piecewise-constant models.

Our method has much in common with these approaches: the notion of multi-layer representations, the use of some mechanism to manage model complexity, and the direct use of support in the estimation process. Our work provides a direct connection between layers and robust estimation, can handle cases of transparency not addressed in the above work, and allows for the integration of different description strategies via the MDL framework. Our approach is derived from the robust statistics and estimation literature, based on the M-estimation framework and the notion of finding segmentation though minimal length description.

2.1 Robust estimation

Robust estimation methods have become popular for image processing, since they have been found to be tolerant to occlusion and other outlier contamination [23, 5]. The use of a support map for estimation is simply an instance of outlier rejection, which is a well known robust estimation method. In this approach a confidence factor is used to weight the contribution of each point to the estimation. The confidence value is itself iteratively updated based on the residual error of the current fit. Formally, this type of estimation is known as M-estimation [15].

M-estimators are maximum likelihood estimators which allow an arbitrary error norm. Given an image data vector $\mathbf{d}$, and a model, $\mathbf{y}(\mathbf{x})$, we wish to find parameters $\mathbf{x}$ which are most likely to have generated the observed data, $d_j, 0 < j < N$. Assuming a probability distribution where each observation is independent:

$$P(\mathbf{d}|\mathbf{y}(\mathbf{x})) = \prod_j e^{-\rho(d_j - y_j(\mathbf{x}))} , \tag{1}$$

and where $\rho()$ is specified *a priori*, an M-estimator attempts to find the optimal $\mathbf{x}$ that maximizes $P(\mathbf{d}|\mathbf{y}(\mathbf{x}))$. With the L_2 norm, $\rho(r) = \frac{1}{2}(r)^2$, corresponding to the Gaussian distribution, finding the maximum likelihood estimate is a linear problem. To be insensitive to outliers, an error norm which decreases the influence of high-error points more rapidly than a squared error norm must be used. However, estimating parameters for robust norms is much more difficult, since in general there are no analytic solutions to the normal equations for an arbitrary $\rho()$.

One solution technique which has been presented for solving the M-estimation problem is Iteratively Reweighted Least Squares (IRLS) [4]. This algorithm approximates an arbitrary norm by the iterative application of a least-squares norm with a dynamic weighting factor on each point. A support vector $\mathbf{s}$ (which we call a *support map* when working with image data) contains these weights and is computed based on the residual error at each point:

$$s_j = \frac{\psi(r_j)}{r_j} \,, \tag{2}$$

where r_j is the residual error of the approximation at point j, $\mathbf{r} = \mathbf{d} - \mathbf{y}(\mathbf{x})$, and $\psi()$ is a weighting function computed by differentiating the error norm:

$$\psi(r) = d\rho(r)/dr \,. \tag{3}$$

Given a model of linear basis functions in a matrix $\mathbf{B}$, a weighted least squares estimate of the parameter vector $\mathbf{x}$ can be obtained by solving the equation

$$\mathbf{x} \; = \; (\mathbf{B}^T\mathbf{W}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{W}\mathbf{d} \,, \tag{4}$$

where $\mathbf{W}$ is a diagonal matrix with $\mathbf{s}$ on the diagonal. In the IRLS method, each time a new $\mathbf{x}$ is estimated, the residual error and support weights are recomputed.

With a robust error norm, points that deviate excessively from an initial fit have their significance down-weighted in subsequent iterations. A well-known robust norm is Huber's minimax, which blends L_1 and L_2 norms. For a segmentation problem, a redescending norm that completely excludes outlier points is appropriate. For reasons of computational efficiency, we have used a "censored" robust norm that combines L_2 weighting for points whose residual error is less than a given threshold θ, and zero weight for "outlier" points that exceed that threshold:

$$\psi(r) \; = \; \begin{cases} r & r^2 < \theta \\ 0 & r^2 >= \theta \end{cases} \,. \tag{5}$$

Collapsing Eqs. (2) and (5), we can concisely express the support threshold rule as:

$$s_j = \begin{cases} 1 & r_j^2 \; < \; \theta \\ 0 & r_j^2 \; >= \; \theta \end{cases} \,. \tag{6}$$

2.2 The breakdown point of traditional M-estimators

The initial conditions of an M-estimator are critical for its success. In particular, if more than a certain number of "outlier" points are in its initial support, the M-estimator will fail. The breakdown point, b, of an M-estimator characterizes this limit of robust performance; it is defined as the smallest fraction of contamination which will force the value of an estimate outside an arbitrary range [23]. If there is more contamination than the breakdown point, the estimate will not necessarily converge to the optimal value. Thus to recover an optimal estimate using an M-estimator, at least $(1 - b)$ of the initial support must cover a single homogeneous region.

The support of an M-estimator is often initialized to cover the entire image (in the absence of any *a priori* knowledge); in this case the breakdown point places a severe limit on the amount of occlusion the estimator can handle. Li [21] has shown M-estimators have breakdown points that are less than $1/(p+1)$, where p is the number of parameters in the regression. Thus even a planar regression fit (with $p=3$) cannot reliably fit a surface when it becomes more than 25% contaminated with occluding data. Other techniques for robust estimation can improve this number, but for a single robust estimator it must be less than $\frac{1}{2}$. Even at this upper limit, a robust estimator will fail on any object that is more than half occluded in the image. In Section 3, we will present our approach to overcoming the breakdown point of M-estimators, by integrating information across multiple robust estimates.

2.3 Estimating Model Complexity

A critical issue in finding a heterogeneous description of a signal is deciding the appropriate level of model complexity to use in the representation. In our case, when we allow multiple estimators to describe a signal, we need to address the fundamental question of how many estimators to use for a given signal? Maximum likelihood estimation provides a means for finding the optimal parameters when the model complexity is fixed, but will not help in deciding how many models to use, or how to compare the performance of models of various order. We need a method for balancing model complexity with model accuracy.

In the Minimum Description Length (MDL) paradigm [33], this is formalized by the notion that the optimal representation for a given image is found by minimizing the combined length of encoding the representation and the residual error. (See also [37] for the related Minimum Message Length criteria.) The theory of information laid out by Claude Shannon, provides the motivation for this approach [31]. Shannon defined "entropy" to be to the lack of predictability between elements in a representation; if there is some predictability from one element to another, then entropy is not at its maximum, and a shorter encoding can be constructed. When the encoding cannot be compressed further, the resulting signal consists of "pure information." Thus if we find the representation with the shortest possible encoding, in some sense we have found the information in the image.[1]

In many cases the MDL criteria can be derived from Bayesian methods. Bayesian probabilistic inference has a long tradition in statistics as well as computer vision and artificial intelligence, and also incorporates the complexity vs. accuracy trade-off. Under this paradigm, we seek to find the representation (process type and parameters) that is most likely given some image data and prior knowledge. Each region of data has a certain probability $P(\mathbf{d}|\mathbf{y}(\mathbf{x}))$ of being generated by a particular representation, and each representation occurs with a certain *a priori* probability $P(\mathbf{y}(\mathbf{x}))$. Through Bayes' theorem

$$P(\mathbf{y}(\mathbf{x})|\mathbf{d}) \;=\; \frac{P(\mathbf{d}|\mathbf{y}(\mathbf{x}))P(\mathbf{y}(\mathbf{x}))}{P(\mathbf{d})}\,, \tag{7}$$

we compute the probability $P(\mathbf{y}(\mathbf{x})|\mathbf{d})$ that a representation accounts for a given region. The *Maximum a Posteriori* (MAP) principle dictates that we should pick the representation

[1] Similarly, simplicity and parsimony have been recognized as essential to notions of representation since the pioneering work of the Gestalt psychologists in the early twentieth century [17]. The *minimum principle* [11] holds that the best representation is the one that accounts for the data with the *simplest* model. The simplest representation is defined as the one with the shortest representation given a set of transformation rules allowed on the data; the search for rules that agreed with human perception was a central focus of their work. Recent researchers in this tradition have used structural rules to define simplicity [20] as well as process models [3].

that maximizes this Bayesian likelihood.

$$\mathbf{x}_{MAP} \;=\; \arg\max_{x}\{P(\mathbf{d}|\mathbf{y}(\mathbf{x}))P(\mathbf{y}(\mathbf{x}))\}\ . \tag{8}$$

The MAP choice is also the minimal encoding of the image when we use an optimal code. Given prior probabilities of various representations, the optimal encoding of each representation can be shown to use $-\log_2 P(\mathbf{y}(\mathbf{x}))$ bits, and the deviation of a region from a representation can similarly be encoded in $-\log_2 P(\mathbf{d}|\mathbf{y}(\mathbf{x}))$ bits. Selecting the minimal encoding is thus determined by:

$$\begin{aligned}
\mathbf{x}_{MDL} &= \arg\min_x\{-\log_2 P(\mathbf{d}|\mathbf{y}(\mathbf{x})) - \log_2 P(\mathbf{y}(\mathbf{x}))\} \\
&= \arg\max_x\{\log_2 P(\mathbf{d}|\mathbf{y}(\mathbf{x})) + \log_2 P(\mathbf{y}(\mathbf{x}))\} \\
&= \arg\max_x\{P(\mathbf{d}|\mathbf{y}(\mathbf{x}))P(\mathbf{y}(\mathbf{x}))\} \\
&= \mathbf{x}_{MAP}\ .
\end{aligned} \tag{9}$$

Thus, when we have “true priors” to use, and can thus find the Shannon-optimal codes, minimal encoding is equivalent to MAP. When these conventional priors are not obvious, the minimal encoding framework provides us with a method of approximating them: we pick the best practical representation we have available. As pointed out by Leclerc, this method is useful in vision problems because it gives us a way to produce estimates using image models that are too complex for calculation of direct priors [18].

We adopt the MDL approach for evaluating a representation, since we have no direct access to the relevant priors. Using the encoding cost of a representation is a natural bias for an information processing system that has finite resources; when used with the appropriate scene models, this framework provides an intuitive and powerful method for recovering non-trivial representations.

3 Robust Estimation with Multiple Models

Our goal is to make accurate estimates of the parameters and segmentation of objects in a scene despite complicated occlusion. As we have seen, in the case of a single object with a relatively small amount of occlusion this can be accomplished by M-estimation, a well known robust method. Unfortunately, the breakdown point limit of an M-estimator restricts its utility–it will only segment out relatively small amounts of a contaminating process. We overcome this limit by explicitly modeling the occluding processes in the signal and sharing this information between the estimates of each process. In our approach, called “Cooperative Robust Estimation”, we use multiple robust estimators to describe a scene, and reallocate the support between them in a cooperative fashion. Occlusion is not solely treated as a additional noise source in the image signal; instead, occluding processes can be modeled as “real” signals, each with a separate support map and robust estimator.

Most importantly, having multiple estimators allows us to use different models and initial conditions in each estimator. Instead of having initial support which covers the entire signal, an estimator can begin with a small window of support, or with a specific set of parameters. Whereas with uniform support the breakdown-point limit prevents an estimator from excluding more than a small fraction of the image as unsupported, with specific initial conditions no such limit exists. For example, if an object covers only 20% of the image, no single robust estimator which initially considers the entire image equally could possibly estimate the object correctly. However if we allow multiple hypotheses, with multiple initial conditions, than a hypothesis whose parameters are estimated from an initial window of support that is localized over the object will be able to accurately recover that object's parameters. With a sufficient number of initial hypotheses, there will be at least one such hypothesis for each object in the scene.

To manage multiple estimates, we use a hypothesize and test paradigm, in which the different estimators can compete to describe the signal. We feel there is intuitive appeal in this approach: if a scene contains several processes, then it seems natural that our model of the scene should incorporate multiple estimators.

3.1 Problem Statement

We assume the signal is composed of one or more components, each of which can be described by a model (or models) known *a priori*. We do not know how many components are in the signal, nor what their parameters or segmentation are. Formally, we assume the image $\mathbf{d}$, can be described as a sum of K masked individual components, with support masks $\mathbf{s}^{(k)}$, a model $M^{(k)}$ with parameters $\mathbf{x}^{(k)}$, plus an additive noise term η, with distribution $N(r)$, which for the examples in this paper we assume is normally distributed with zero mean and variance σ^2:

$$\mathbf{d} = \sum_{k=1}^{K} \mathbf{s}^{(k)} M^{(k)}(\mathbf{x}^{(k)}) + \eta \ . \tag{10}$$

Each model $M()$ is some function which can be applied to parameters $\mathbf{x}$ to generate an estimated image, and for which an estimator is available to perform a weighted estimation using a support map. For a model composed of a set of linear basis functions, we can use an M-estimator for this task. For other models a more complicated estimator is needed (such as the recursive structure from motion estimator used in the example in Section 5.3); however, most estimation methods can be adopted for use with a support map without much difficultly.

3.2 Method Overview

We tackle the problem in two steps. First, we attempt to estimate the number of components in the signal, e.g. to estimate K. To accomplish this also requires finding coarse estimates of the parameters, model, and support for each component. Second, once we have obtained an estimate of K, we revise the parameter and support estimates for each component, using an iterative refinement procedure.

3.3 Estimation of K

We have advocated the use of a Hypothesize, Test, and Select approach to estimating the number of objects in a signal [7, 26]. In our method a set of hypothesizes is generated, each is tested against the observed data to compute a set of support maps, and the subset of hypotheses which optimally (in an MDL sense) accounts for the data is selected. We will describe each of these stages in turn.

3.3.1 Initialization of Hypotheses

Hypotheses are comprised of a chosen model and associated parameters. A initial set of hypotheses,

$$\mathcal{I} = \{H_0,\ H_1,\ ...,\ H_M\}\ , \tag{11}$$

is constructed either by sampling the space of possible models and parameters, and/or by selecting models and small windows of the data and using these to estimate a set of model parameters. In general terms, for our system to be effective the initial set of hypotheses must have at least one hypothesis in rough correspondence with each real object or process in the scene. Redundant and/or completely erroneous hypotheses in the initial set are tolerated by our method, and are expected.

Depending on how the initial set is constructed, different constraints apply to the number of hypotheses needed in the set. If the initial set is specified by sampling parameter space, then the sampling must be fine enough to guarantee some hypothesis will support each object that may be in the scene. This will depend on the choice of support threshold, θ, described below. If the initial set is specified via estimates formed from windows on the data, then the windows must be chosen to guarantee (to some sufficient probability) that for each object in the scene, at least one hypothesis has approximately homogeneous support for that object, e.g. that percentage of points not from that object is less than breakdown point of the esimtator.

3.3.2 Hypothesis Support Testing

Given this set of initial hypotheses, with models and associated parameters, we compute a support map for each hypothesis, using Eq. (6). This step essentially entails the computation of an estimated surface for each hypothesis by applying its model to the parameters, computing the difference between the estimated surface and the observed data, and then thresholding the residual error according to the threshold parameter θ.[2]

The residual threshold θ is bound by two constraints. First, it must be large enough that the hypothesis closest to a given object in parameter space will support a majority of the points on that object. This ensures that a unique hypothesis can be found for each object. If θ is too low, then multiple hypothesis may be selected for a given object, creating "false alarms" (K will be over-estimated). Second, θ must be small enough so that the support for any estimate that supports a majority of points on one object will not also support a majority of points from another object. If θ is too large, a single hypothesis may account for two real objects in the scene, providing both a corrupted estimate and a "miss" (K will be under-estimated.)

The selection of θ is domain-dependent, and so should be considered a free parameter of our system. In practice we have found that the above heuristics are sufficient to determine θ given some *apriori* knowledge about the distribution of processes in the scene, and the standard-deviation of the noise process, σ.

3.3.3 Selection of Minimal Description subset

Once we have a set of hypotheses and their regions of support in the image, we find the subset of these estimates which best describes the entire scene. In the initial set there

[2]The framework allows different models to use different threshold parameters, as larger thresholds are counterbalanced by lower entropy savings in the selection stage. However in the examples presented here we have kept θ constant across models to simplify the presentation.

will be much redundancy: each object may be covered by several hypotheses, and many hypotheses will be covering regions of the scene that do not correspond to an single object. Our approach to this problem is to find the set of hypotheses which most parsimoniously describes the image signal.

Given the set of initial hypotheses, we search for the subset which is the *minimal length encoding* of the signal. Ideally, this solution will consist of one hypothesis for each object in the scene. For a subset $\mathcal{H} \subset \mathcal{I}$ which represent opaque regions of the image, the description length function $L(\mathcal{H})$ can be defined as the trivial point-by-point encoding of the image minus the encoding savings offered by that subset:

$$L(\mathbf{d}|\mathcal{H}) = L(\mathbf{d}) - S(\mathbf{d}|\mathcal{H}) \ . \tag{12}$$

Since $L(\mathbf{d})$ is constant with respect to $\mathcal{H}$ in Eq. (12), the minima of $L(\mathbf{d}|\mathcal{H})$ will correspond to the maxima of $S(\mathbf{d}|\mathcal{H})$, the encoding savings of all hypotheses in $\mathcal{H}$.

We define the encoding savings for a hypotheses to be the sum of the savings over the supported points in that hypothesis, less an overhead term. Assuming $\mathcal{H}$ is a partition and thus there is no overlap in the support of member hypotheses, we define the savings for a set a hypothesis to be the sum of the savings of the individual hypotheses:

$$S_p(\mathbf{d}|\mathcal{H}) \ = \ \sum_{H_i \in \mathcal{H}} S(\mathbf{d}|H_i) \ = \ \sum_{H_i \in \mathcal{H}} \sum_{j}^{N} h_j^{(i)} s_j^{(i)} - \alpha \ , \tag{13}$$

where $h_j^{(i)}$ is the encoding saved at point j by hypothesis H_i. If we have the probabilities $P(d_j)$, $P(d_j|H_i)$, and $P(H_i)$, then the Bayesian interpretation of h and α is simply:

$$h_j^{(i)} = -\log_2 P(d_j) + \log_2 P(d_j|H_i) \tag{14}$$

$$\alpha^{(i)} = -\log_2 P(H_i) \ . \tag{15}$$

Without such probabilities, we can use the encoding cost of the image in its current representation. We assume that residual error values are uniformly distributed, and thus save $\log_2 \frac{G}{\theta}$ bits, where G is the number of grey levels per pixel. The overhead is interpreted in an MDL context as the cost of storing the model index, model parameters, and support map in their current representation (e.g. the number of bits used).

If the models are not a perfect approximation of the data, and/or the image has considerable noise, the support maps in the initial hypothesis set will not perfectly reflect the shape of objects in the scene. In this case the overhead term is useful as a measure of how much support a hypothesis should have to be considered viable. It should be bounded above by the smallest amount of savings we expect in a hypothesis, otherwise a real object may be missed in the selection process and K underestimated. In addition, it acts as a counterbalance to prevent the degenerate solution of has one hypothesis per pixel in the image. It should thus be bounded below by the largest spurious hypothesis we expect, e.g., the amount of savings that be gathered from an object despite another selected hypothesis which already covers a majority of the support of that object. If α is too low, multiple hypotheses may be selected for a single object, over-estimating K.

Eq. (13) expresses the encoding savings for a set of hypotheses with no overlap in support. If a given hypothesis set is not a partition, and has elements with overlapping support, this equation will over-estimate the joint savings of the set. As support maps will often overlap during the intermediate states of the selection procedure, we need to augment this expression with a term to account for support overlap.

This is a type of "credit assignment" problem: in this case overlapping support means two different models are allowed to each take full credit for the same data point. There are a number of possible remedies: one could arbitrarily assign the credit for a point to one of the overlapping hypotheses, normalize the credit between them, or discount the credit for contested points altogether. We experimented with each of these strategies, and found the last was most reliable. (Arbitrary assignment yields unpredictable results, and normalizing credit leads to stable solutions with duplicated hypotheses.)

Our modification to Eq. (13) is to thus introduce a term to discount the effect of overlapping support. Our revised encoding savings function only takes credit for points which are supported by exactly one hypothesis in the selected set:

$$S(\mathbf{d}|\mathcal{H}) \;=\; \sum_{H_i \in \mathcal{H}} \left(\sum_{j}^{N} h_j^{(i)} \left[s_j^{(i)} - \sum_{l \neq i, H_l \in \mathcal{H}} s_j^{(l)} \right]_{>0} - \alpha \right) \;, \tag{16}$$

where $[x]_{>0} = x$ for positive x and 0 elsewhere, e.g. half rectification. Points which are contested (have multiple overlap) are not counted in the support of any hypothesis during the selection optimization.

Our task, then, is to find a subset $\mathcal{H}$ of our initial set of hypotheses $\mathcal{I}$ which maximizes $S(\mathbf{d}|\mathcal{H})$. To achieve this, we could employ exhaustive search to test all possible combinations of hypotheses in our candidate set, but this would be computationally infeasible. In practice we have found that it is sufficient to use a gradient descent method to perform this optimization. This optimization can be embedded in a continuation method when local minima are a problem.

3.3.4 Numerical solution

To find a maxima, we need to express $S(\mathbf{d}|\mathcal{H})$ in a differentiable form. First, we represent $\mathcal{H}$ by enumerating which elements of $\mathcal{I}$ are in $\mathcal{H}$ using a vector $\mathbf{a}$. The sign of each element a_i indicates the membership of an element H_i: a positive value means it is in $\mathcal{H}$, and a negative value means it is not. Initially a_i is set to zero for each description, representing an "unknown" condition.

We define $f(a)$ to be a selection function and transforms the real a values into a multiplicative weight between 0 and 1. Ideally, $f()$ would be the unit step function centered at the origin. However, this hard non-linearity makes it very difficult to maximize $S(\mathbf{d}|\mathbf{a})$. Instead, we use the "softer" sigmoid function,

$$f(x) = \frac{1}{1 + e^{-kx}} \;. \tag{17}$$

With this, we can re-express Eq. (16) as

$$S(\mathbf{d}|\mathcal{H}) \;=\; S(\mathbf{d}|\mathbf{a}) \;=\; \sum_{i}^{M} f(a_i) \left(\sum_{j}^{N} h_j^{(i)} \left[s_j^{(i)} - \sum_{l \neq i}^{M} f(a_l) s_j^{(l)} \right]_{>0} - \alpha \right) \;. \tag{18}$$

We update $\mathbf{a}$ with

$$\frac{da_i}{dt} = \frac{1}{C} \frac{dS(\mathbf{d}|\mathbf{a})}{df(a_i)}$$

$$= \frac{1}{C}\left(\sum_{j}^{N} h_j^{(i)}\left[s_j^{(i)} - \sum_{l\neq i}^{M} f(a_l)s_j^{(l)}\right]_{>0} - \alpha\right) . \tag{19}$$

This rule increments a_i whenever the hypothesis H_i is generating enough encoding savings to offset the overhead cost penalty α, discounting any contested points. We implement this optimization using forward-Euler discretization, with C serving as an integration constant [30]. As shown in Appendix B, Eq. (19) will converge to a local maxima of $S(\mathbf{d}|\mathbf{a})$.

In most cases, including all the examples presented in Section 4, we have found that the local maxima found via this gradient ascent procedure corresponded to correct estimates of K, and perceptually salient decompositions of the scene. However, if the space of support maps is quite large and complex it is possible that spurious local maxima may exist. In previous work [25] we have employed a continuation method on the overhead cost α when local maxima were a problem. This continuation method initially biases the method to first find descriptions that cover large regions of the image, and then find smaller descriptions. Use of this continuation method has shown itself useful at avoiding spurious minima, and in addition can improve the convergence rate.

3.4 Refinement of estimates for fixed K

The previous sections describe a method for initializing a set of robust estimators with different initial states and selecting the subset that provides the most parsimonious description of the image. Roughly, these steps estimate "how many" processes are in the image signal, and coarsely determine their support and parameters.

Once we have this estimate of the number of processes, we refine the parameter and support estimates in the hypothesis set using a cooperative update rule. In this step we assume that current set of hypotheses consists of one hypothesis for each actual process in the scene, e.g. that the estimate of K is correct. We then apply a stronger rule for support determination than that used by single robust M-estimators, by taking into account the residual error of *all* the models. Generalizing the robust estimation reweighting function, Eq. (2), we use a reweighting function that depends on the entire set of residuals in $\mathcal{H}$ for a point j, as well as the local residual at that point for description i, $r_j^{(k)}$:

$$s_j^{(k)} \;=\; \frac{\psi\left(r_j^{(k)},\;\; \mathcal{R}_j\right)}{r_j^{(k)}}. \tag{20}$$

where $\mathcal{R}_j$ is defined as

$$\mathcal{R}_j \;=\; \{r_j^{(0)}, r_j^{(1)}, r_j^{(2)}, ..., r_j^{(K)}\} \tag{21}$$

Since the only form of overlap we are considering is occlusion, we limit the cooperative ψ function to only depend on the minimum residual at a particular image point. We can minimize the total residual error in all description estimates by only allowing the estimator with the lowest residual to keep a given point. We use

$$\psi(r_j, \mathcal{R}_j) = \begin{cases} r & if \;\; (r_j)^2 \;<\; \theta^{MAX} \\ & and \;\; (r_j)^2 \;=\; \min_i (r_j^{(i)})^2 \\ 0 & otherwise \end{cases} \tag{22}$$

where θ^{MAX} is the largest possible residual error value generated by the noise model $N()$. With this function, each estimator iteratively performs a least-squares estimation over the points for which it has lowest residual error, subject to the maximum residual threshold θ^{MAX}. In the case of Gaussian $N()$ the distribution has infinite extent, so $\theta^{MAX} = \infty$ and is thus ignored. However for distributions with compact distributions and/or when there are true outliers which cannot be modeled by any hypothesis, the use of θ^{MAX} can improve the segmentation result.

The information provided to an estimator by the cooperative ψ function allows the exclusion/segmentation of outliers based not just on their deviation from the prior model, but also on the ability of some other estimator to account for the point in question. Combining Eqs. (20) and (22), our cooperative update rule is:

$$s_j^{(k)} = \begin{cases} 1 & if \;\; (r_j^{(k)})^2 \; < \; \theta^{MAX} \\ & and \;\; (r_j^{(k)})^2 \; = \; \min_i (r_j^{(i)})^2 \\ 0 & otherwise \end{cases} \tag{23}$$

4 Examples

The problem of reconstructing a signal containing several homogeneous but disjoint chunks has broad application in image processing and perception. We have experimented with our framework both in the domain of shape-based segmentation of range images, and in the domain of motion-based segmentation of image sequences.

4.1 Segmentation of range images using piecewise-polynomial models

A straightforward application of our method is the approximation of images with disconnected regions consisting of piecewise polynomial patches. The model used in this example consisted of linear basis functions for global approximation combined with thin-plate regularization for local smoothing. We used the M-estimator described in section 2, with polynomial basis functions. The image was modeled by

$$I(u,v) \;=\; x_0^{(k)} + x_1^{(k)}u + x_2^{(k)}v + x_3^{(k)}uv + \ldots \,, \tag{24}$$

where $\mathbf{x}^{(k)}$ is the parameter vector for hypothesis H_k. A matrix $\mathbf{B}$ is constructed with columns corresponding to the polynomials of the model, so that when the image is expressed as a data vector $\mathbf{d}$ the model can be expressed as

$$\hat{\mathbf{d}} \;=\; x_0^{(k)}\mathbf{b_0} + x_1^{(k)}\mathbf{b_1} + x_2^{(k)}\mathbf{b_2} + x_3^{(k)}\mathbf{b_3} + \ldots \;=\; \mathbf{B}\mathbf{x}^{(k)} \,. \tag{25}$$

Since even poor polynomial fits will usually have some points where the local error of approximation is zero (in particular at any point where the the data and the estimated surface accidentally cross), we regularize the squared error signal using a thin plate model. This will cause very small regions of support, such as those from accidental zeros, to be reduced. Parameters are estimated from the data via Eq. (4), and residual error is computed by

$$r_j^{(k)} = \mathcal{E}_\lambda(d_j - (\mathbf{B}\mathbf{x}^{(k)})_j) \,, \tag{26}$$

where $\mathcal{E}_\lambda$ is a thin-plate regularization function [28]. In these examples we used $\lambda = 1.0$.

We show the results of our system on several different piecewise-polynomial images. First we show an example using constant regions, to illustrate pedagogically the stages in processing. (Since in this example the parameter space is scalar and the edges relatively large-scale, we do not wish to suggest conventional techniques could not be successfully applied to this image.) The subsequent examples utilize models that generate image regions of considerable complexity, which would be difficult or impossible to segment using conventional edge-based methods.

- Figure 1(a) shows an image with six constant regions, two of which are spatially disjoint. We constructed an initial hypothesis set ($M = 16$) by uniformly sampling the (scalar) parameter space. Figure 1(b) shows the support maps computed for the initial hypotheses, using $\theta = 16$ to ensure that the full range of parameters would be "covered" by the hypothesis set. As can be seen, most hypotheses have "found" at least one object. However, there is redundancy in the set, as each object is covered by more than one hypothesis. Also, the segmentation at this stage is not perfect, since the regularization has "smoothed" out the edges.

The next stage in processing is to select the subset of hypotheses which constitutes a minimum lengh description of the signal, as described in Section 3.3.3. In all the examples shown in this paper, the model overhead term used in the selection method, α, was kept constant across models, and was set to be 1% of the total entropy of the image being processed. The selection stage selected 6 hypotheses, shown in Figure 1(c), corresponding to the actual regions in the image. Parameters were re-estimated for these hypotheses and the cooperative support update rule (Eq. (23)) applied, yielding the final support maps shown in Figure1(d). The segmentation is perfect, as the cooperative rule eliminated the distortion in support shape induced by the smoothing process.

Figure 2 shows the results of repeating this example for differing levels of additive Gaussian noise, from $\sigma = 5$ to $\sigma = 20$. To avoid the "false alarm" situation described in section 3.3.2, θ was set to be the inter-hypothesis distance, or the standard deviation of the noise, whichever was greater ($\theta = \max(16, \sigma)$). The solution was insensitive to small variations in α; perturbations of up to 40% of α were tolerated without impairing the correct estimation of both K in these experiments. While the precision of the estimated support maps degrades as it becomes increasingly difficult to separate the different populations with a threshold value, the important result is that in each case K and the coarse model parameters were correctly estimated. This would allow for a later stage of processing to apply more strict domain knowledge, if available.

- Next, we tested our method on second-order polynomial surfaces with high-order discontinuities. Figure 3 shows the construction of an image that contains three different underlying surfaces, with a sinusoidally shaped discontinuity between two regions, and a straight but high-order discontinuity between the other two. Note that the high-order discontinuity is (usually) not visible to human observers.

  An initial set of hypotheses was created by fitting parameters to $8x8$ patches of the image. Figure 4 shows the results for the noise-free case ($\theta = 1$), and Figure 5 for the case where $\theta = \sigma = 5$. (Since hypotheses were initialized using windows rather than parameters, no minimum θ was needed.) Despite the complicated edge structure, nearly perfect segmentation is achieved in the noise-free case. And as in the previous examples, the noisy case degrades the support map estimates, but does not prevent the correct estimation of K.

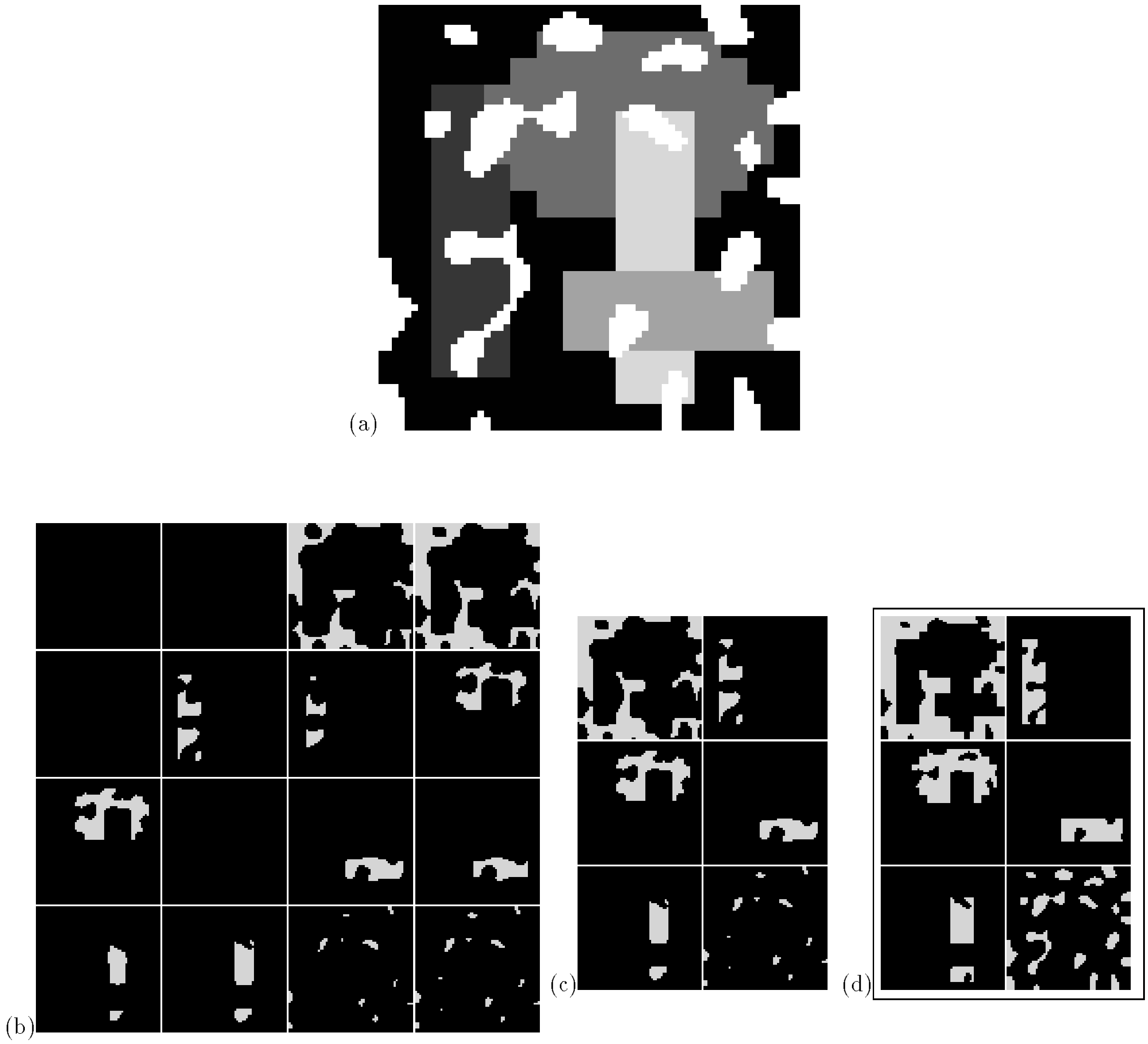


Figure 1: Segmentation of image with constant regions. (a) image (b) initial hypothesis set (c) selected hypotheses (d) selected hypotheses after support was reallocated using cooperative support update rule.

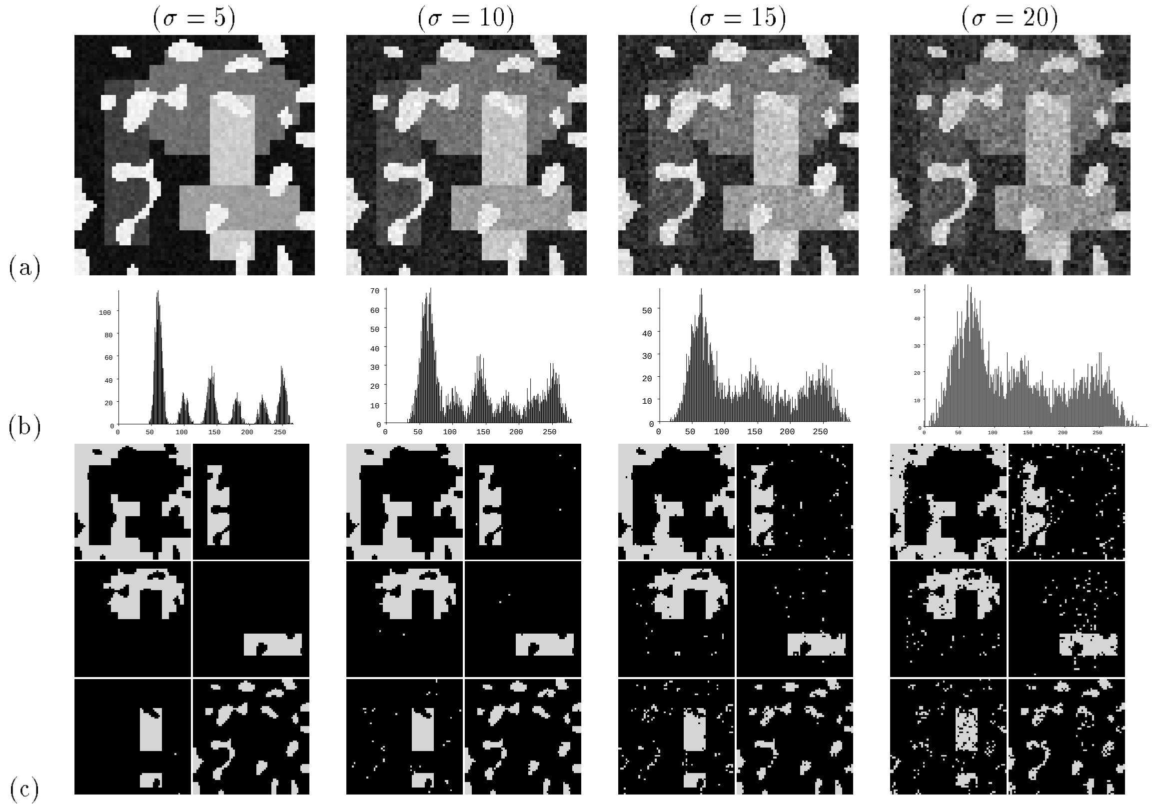


Figure 2: Segmentation of image with constant regions for varying amounts of additive Gaussian nose (a) image (b) histogram of image (c) computed support maps

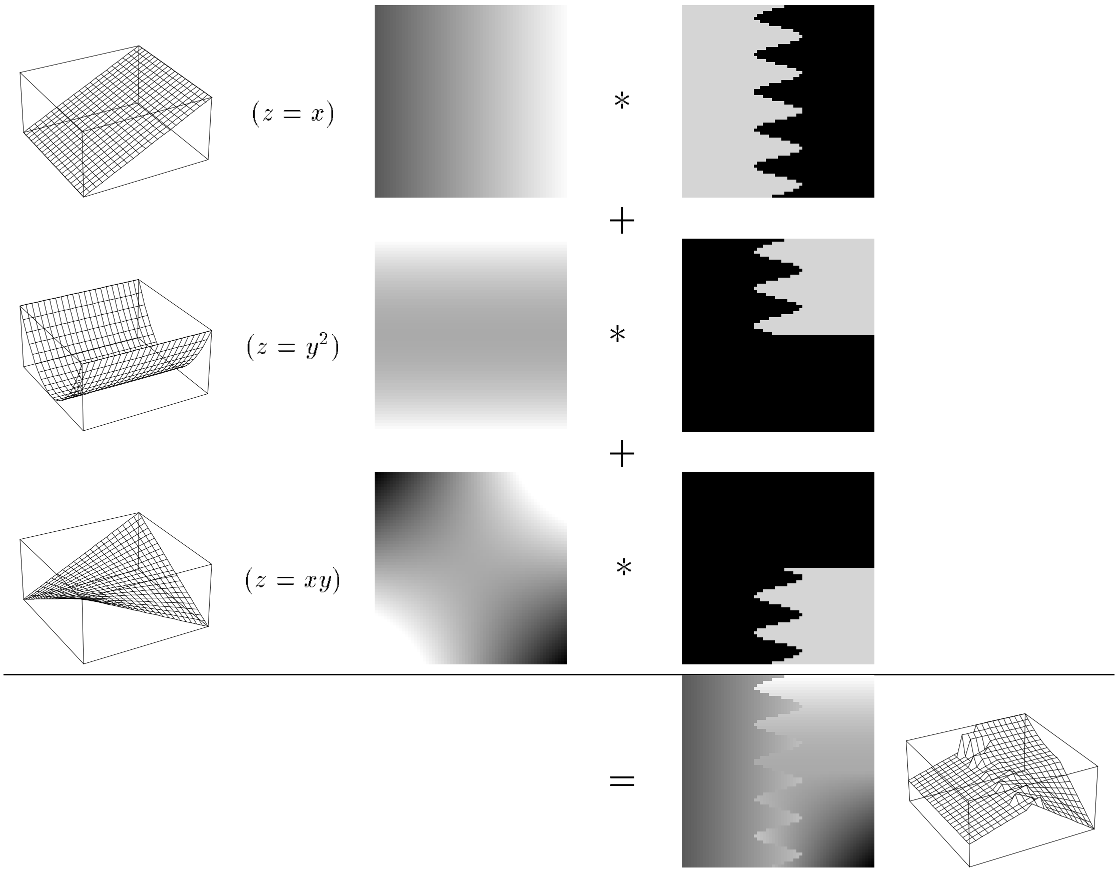


Figure 3: Construction of synthetic image with complicated occlusion boundaries.

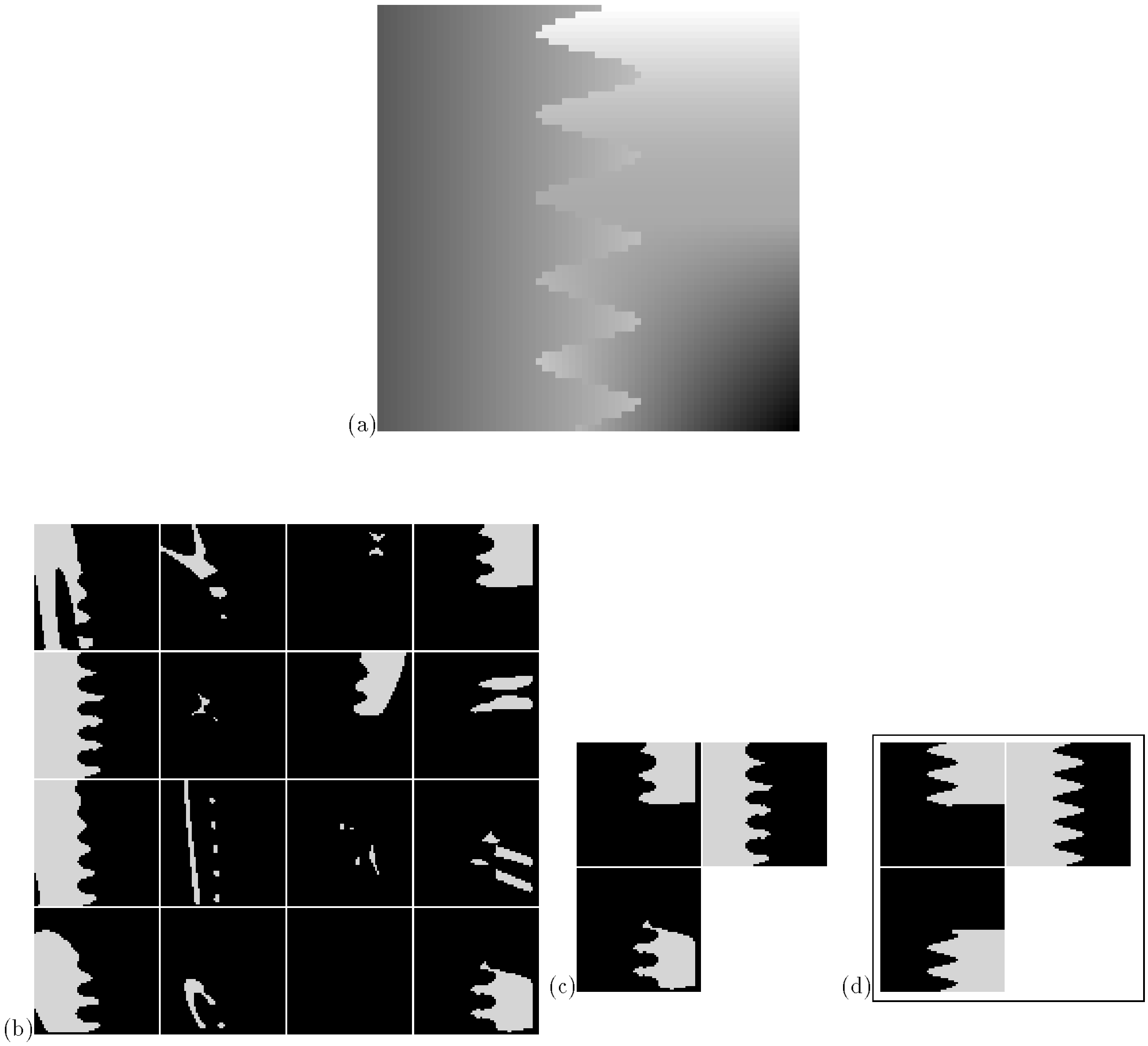


Figure 4: Segmentation of image with second-order regions. (a) image (b) initial hypothesis set (c) selected hypotheses (d) selected hypotheses after support was reallocated using cooperative support update rule.

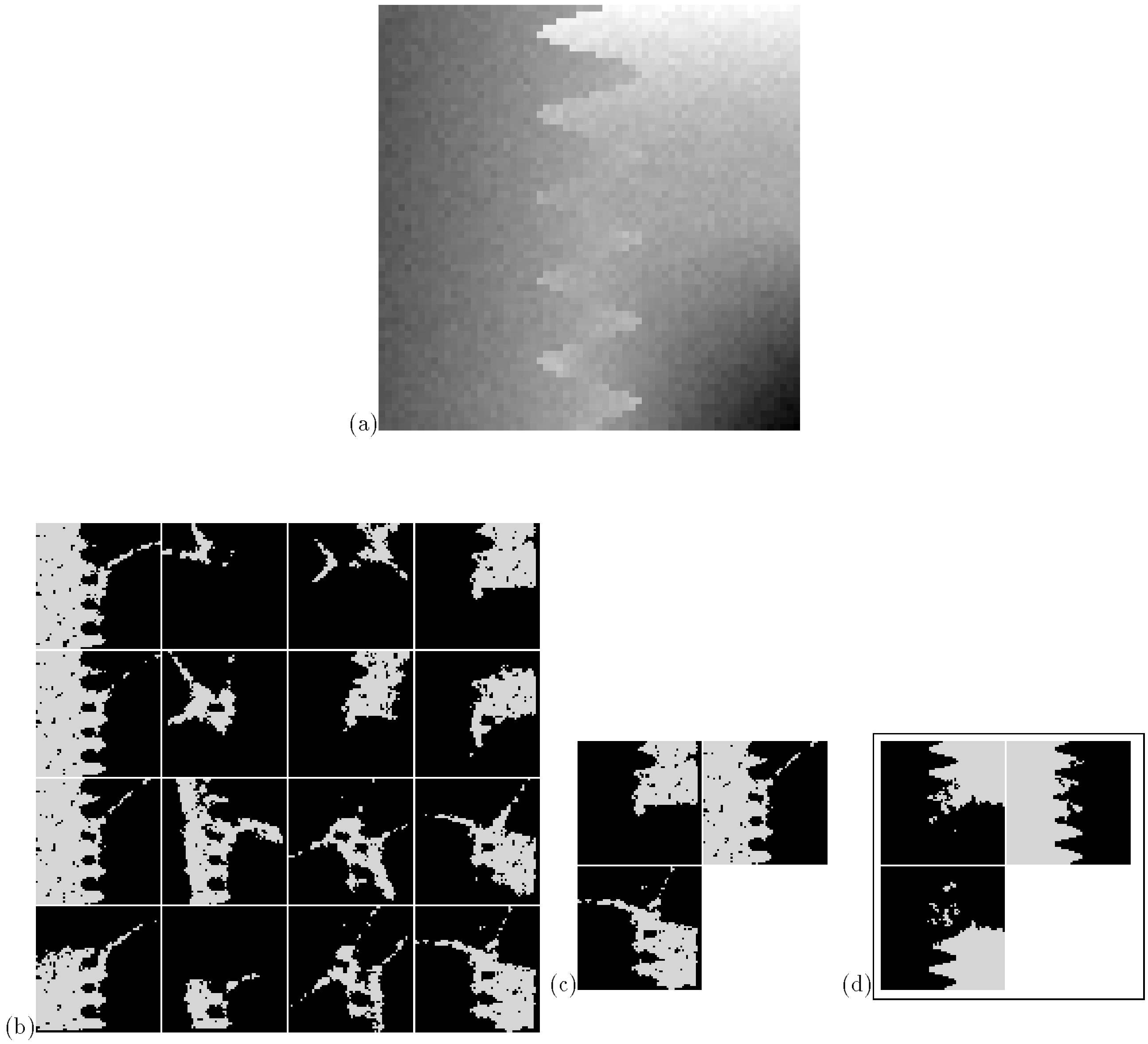
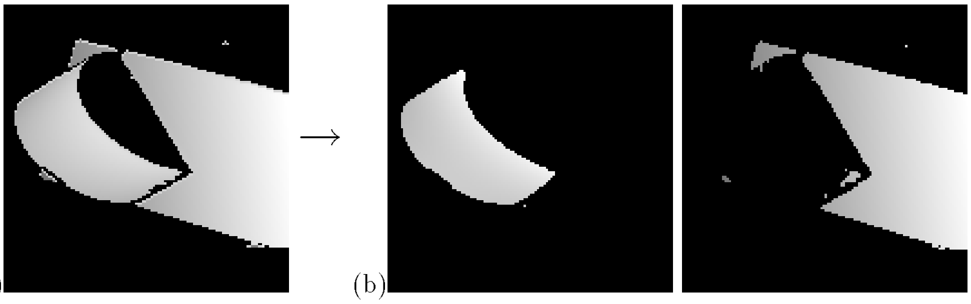


Figure 5: Segmentation of image with second-order regions and additive Gaussian noise ($\sigma = 5$). (a) image (b) initial hypothesis set (c) selected hypotheses (d) selected hypotheses after support was reallocated using cooperative support update rule.



(a) (b)

Figure 6: (a) Range data for an arch sitting on a block. (b) hypotheses found based on fitting second order polynomial surfaces.

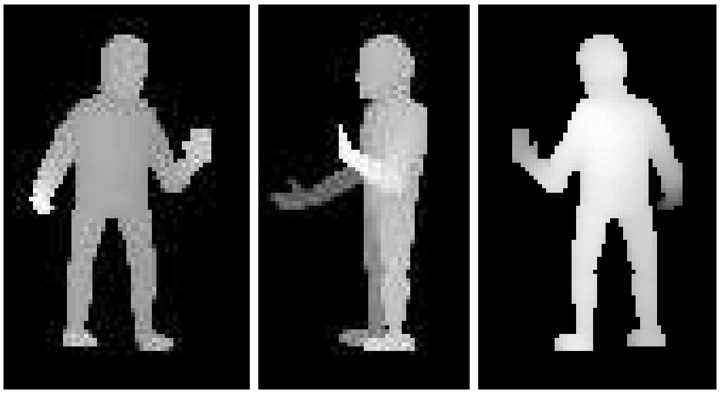
- Figure 6(a) shows a range image of an arch sitting on a block, obtained from a laser range finder.[3] We applied our method as in the previous example. As shown in Figure 6(b), two hypotheses were selected to account for the image. For the planar hypothesis, several small groups of points which are disjoint from the main support were successfully grouped together, despite being far (in the image) from the main region of support. However, because the polynomial basis functions are not a perfect model for the actual surfaces in the scene, a small number of points are misclassified in the recovered support map. Better models of the surface (such as modal models [27]) would help to alleviate this.

- To test the stability of segmentations using this method, we applied it to a series of different views of a 3D figure. Figure 7(a) shows synthetic range data of a human figure taken from three different viewpoints. Figure 7(b) shows the segmentation produced for each view. Note that the system was able to find the major parts in each view, and was able to distinguish the chest part from the torso part. Since the second order surface model was only a coarse approximation of the actual shapes, we used a relatively large threshold value, $\theta = 30$. As a result some of the smaller parts (e.g. hands and feet) were not described successfully, and certain aligned parts were incorrectly merged into a single hypothesis (e.g. the leg parts.) Nonetheless the important result is that the segmentations were stable across different views, recovering the same essential part-based description to represent the figure.

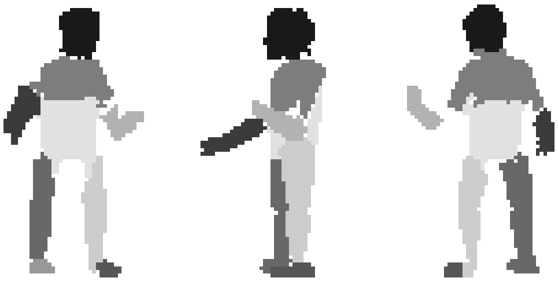
  We were able to use the resulting segmentations to recover an 3-D model from the range data. A modal model part was fit to data for a corresponding part in each view using the ThingWorld modeling system [27]. Camera position was known for the views used in this example, so the part correspondences were straightforward to compute. The recovered 3-D model is shown in Figure 7(c).

[3] Range image obtained via anonymous FTP from the MSU Pattern Recognition and Image Processing Lab.

(a)



(b)



(c)

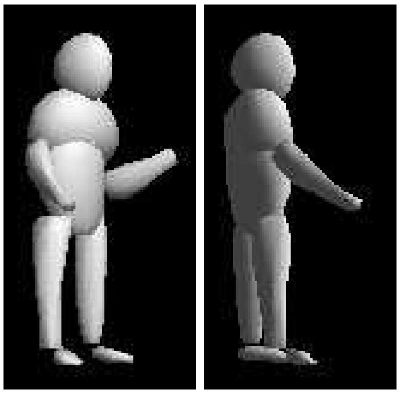


Figure 7: (a) Synthetic range data for a human figure, taken from three views. (b) Color-coded segmentations for each view obtained using second-order polynomial shape models. (c) Segmentations from previous figure were used as input to a 3-D shape estimation process. Two views are shown of the recovered 3-D model.

4.2 Image sequence segmentation using global velocity field models

We have also applied our method to the domain of motion segmentation using global velocity field models [8]. In this case the task is to decompose an image sequence into a set of layers corresponding to homogeneous motions, based on a global model of coherent flow and a gradient-based model of local velocity measurements. We adopt a global model of velocity fields in the scene, using a linear combination of horizontal and vertical shifts together with a looming field:

$$V(x,y) = a\begin{bmatrix} 1 \\ 0 \end{bmatrix} + b\begin{bmatrix} 0 \\ 1 \end{bmatrix} + c\begin{bmatrix} x-d \\ y-e \end{bmatrix} , \tag{27}$$

where a, b, c, d, e are the parameters of the model. This model is appropriate for a large class of common sequences, e.g. those that predominantly contain translations in 3-D. For more complicated sequences higher-order velocity field models may be appropriate, such as the affine model [29, 36].

In our model, each hypothesis is a global velocity field, which will imply a local velocity at every point in the scene. To test the support of a hypothesis, we thus need to estimate whether a particular velocity is present in a local region. We compute support by thresholding a residual error signal (via Eqs. (6),(23)) which is small at points where the predicted velocity is likely to be present, given the information in the image. According to the gradient constraint, the spatial and temporal image derivatives at a point obey the equation $D(v(x,y))I(x,y,t) \;=\; 0$, where

$$D(v) = \begin{bmatrix} v_x \\ v_y \\ 1 \end{bmatrix} \cdot \nabla = (v_x\frac{\partial}{\partial x} \;+\; v_y\frac{\partial}{\partial y} \;+\; \frac{\partial}{\partial t}) \, . \tag{28}$$

Simoncelli [35] has derived a probabilistic model for optic flow computation, which we use to define the residual error of a velocity hypothesis. Assuming there is only a single motion in the neighborhood,[4] we can use:

$$r_j = -\log P(\nabla I | v_j) \approx \frac{||D(v_j)I||^2}{||\nabla I||^2} \, . \tag{29}$$

Image derivatives were computed using 5-tap linear filters, and the threshold value θ set to be the smallest value for which there were layers covering all regions of the scene which had some motion or texture.

Here we show the results of our system on three different image sequences:

- We first constructed a transparent, flickering random dot image sequence similar to the one used in the Husain, Treue, and Andersen experiment described in Section 2. Our image sequence had two interleaved populations of moving dots, one undergoing a looming motion, and the other undergoing a translating motion, as shown in Figure 8(a). Since our approach recovers a global motion estimate for each region in each frame, no explicit pixel-to-pixel correspondences are needed over long sequences. After each frame 10% of the dots died off and randomly moved to a new point on the 3-D surface. We rendered ten frames of this motion sequence; the first frame is shown in Figure 8(b).

[4] for the extension of our model to the case of motion with additive transparency, see [9]

We adopted an initial set of velocity field hypotheses corresponding to 8 planar shifts:

$$\begin{aligned}\{(a,b,c)\} = \{&(-1,-1,0),\ (-1,0,0),\ (-1,1,0),\ (0,-1,0),\ (0,1,0), \\ &(1,-1,0),\ (1,0,0),\ (1,1,0)\}\end{aligned} \tag{30}$$

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

We applied our method independently to each frame in the sequence, and in each case two hypotheses were found to represent for the motion information in the scene. The hypotheses recovered for a particular frame are shown in Figure 8(c). One was estimating the looming motion in the sequence, and the other the translating motion. The recovered motion parameter estimates were quite stable across different frames, despite the random appearance and disappearance of sample points. An edge-based segmentation method would have yielded nonsensical results on this image sequence, indicating many "edges" (and thus many regions) in a scene which in fact contains only two regions, transparently combined.

- Next, we ran our model on real intensity sequences with complicated occlusion, using the same initialization conditions as in the previous example. Figure 9(a) shows a frame of an image sequence containing two plants in the foreground with a person moving behind them. The person is occluded by the plant's leaves in a complex manner, so that the visible surface is broken up into many disconnected regions. Figure 9(b) shows the segmentation provided by our method when run on this sequence. Two regions were found, one for the person and one for the two plants. Most of the person has been correctly grouped together despite the occlusion caused by the plant's leaves. Note that there is a cast shadow moving in synchrony with the person in the scene, and is thus grouped with that description.

- Figure 10 shows a frame from a similar image sequence, taken from the MPEG test suite. The image sequence shows a house and a flower garden with an occluding tree, and the camera motion is panning from right to left. We applied our method as described above to the frame shown in the figure, except support windows rather than parameter sampling were used to define the initial hypothesis set. We estimated motion parameters from 256 8x8 windows, and used these as the initial hypotheses. A support map over the entire image was computed for each hypothesis. The selection stage recovered three layers corresponding to different depth planes in the sequence. Both the house and garden regions are split in two by the occluding tree; our method correctly grouped both of these in their respective layers. Note that areas without significant spatial or temporal variation (such as the sky and center of the tree) are not assigned to any layer.

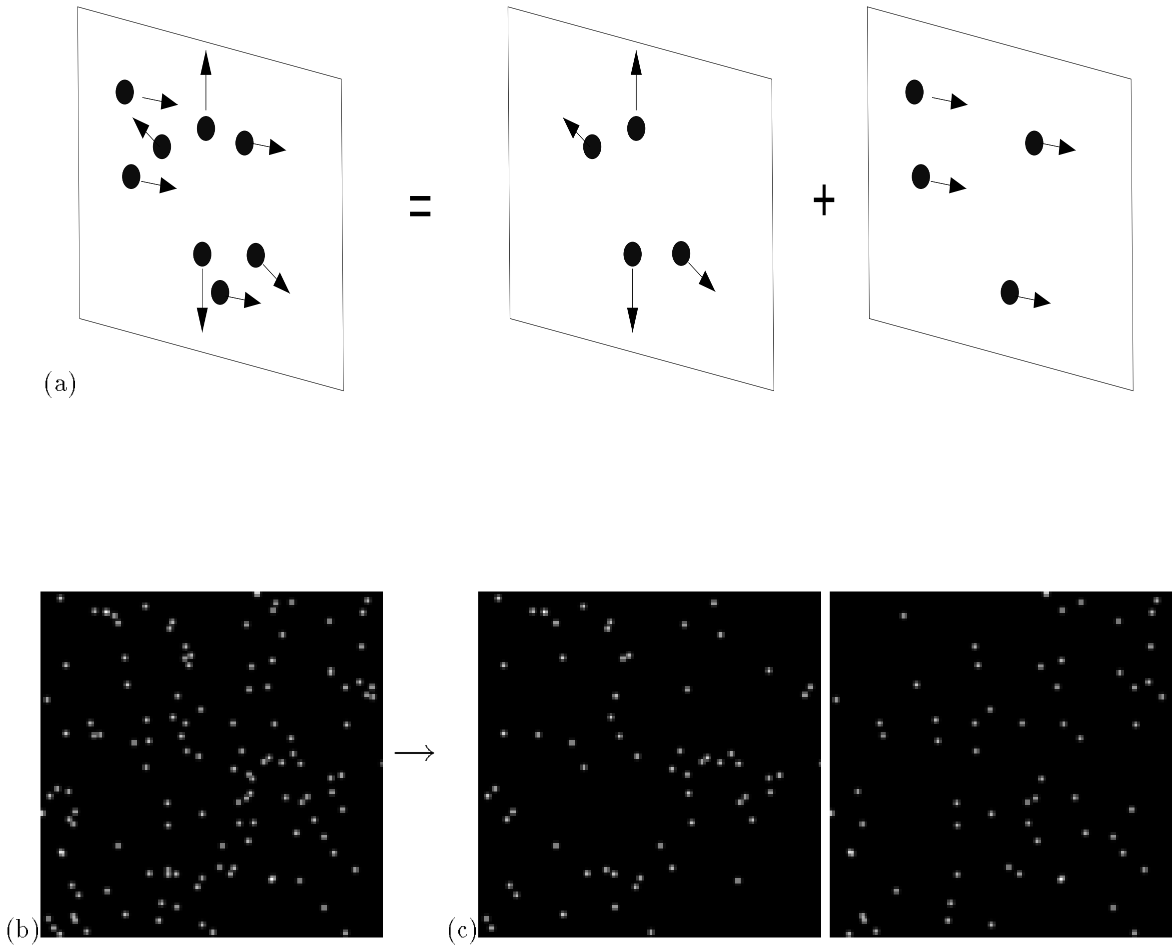


Figure 8: (a) Overview of two-motion transparent, flickering random dot stimulus; (b) first frame of sequence. (c) Two layers were found by our system for each frame, corresponding to the two coherent motions in the sequence (see text for details).

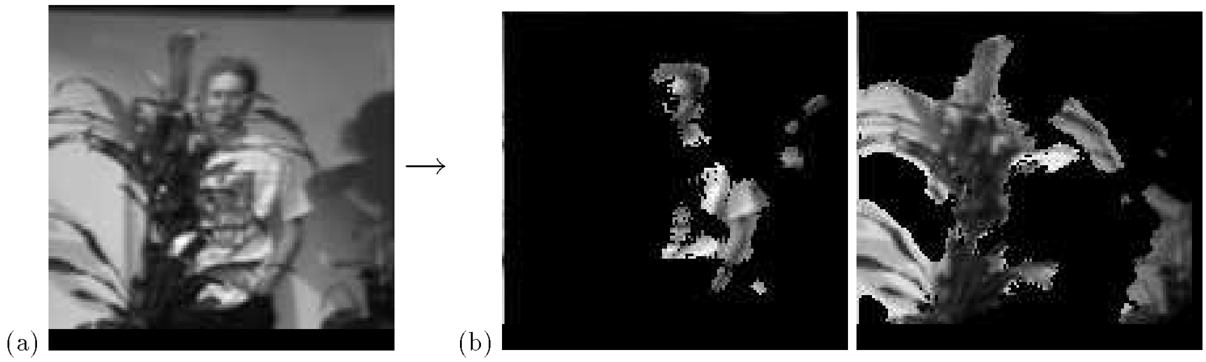
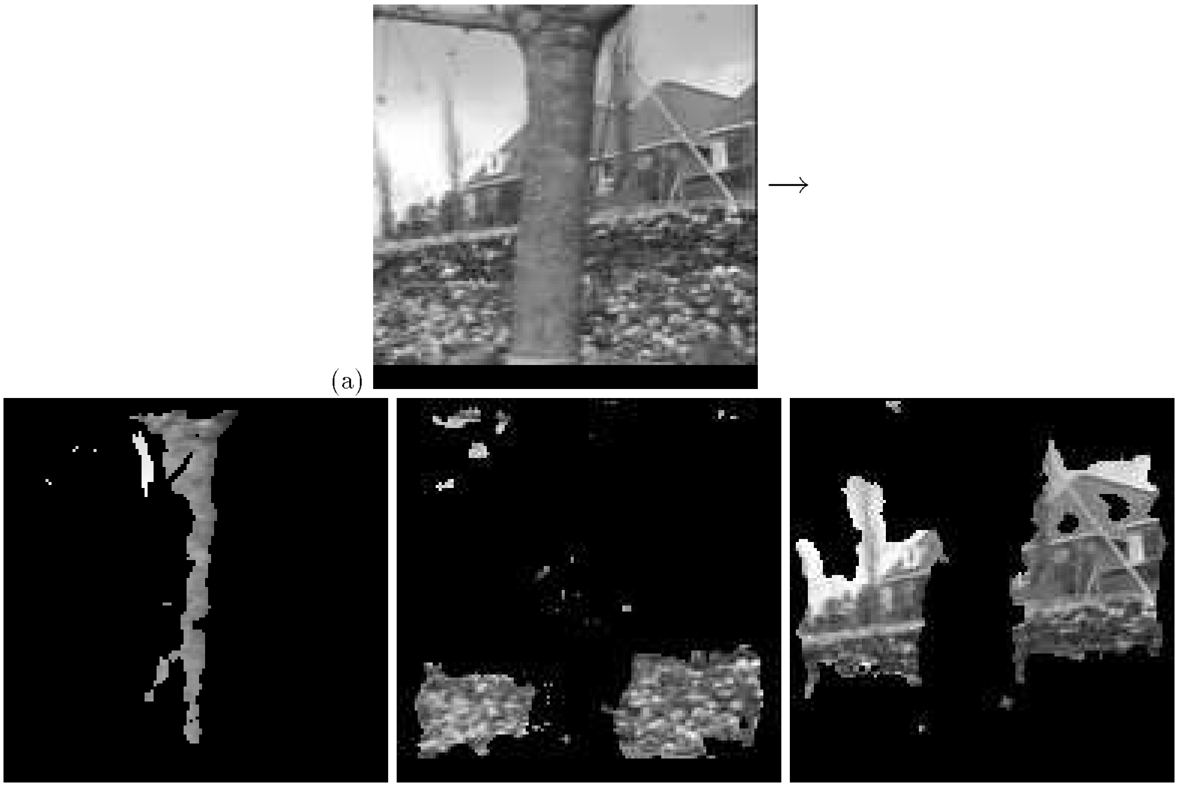


Figure 9: (a) First frame from image sequence and (b) final hypotheses.

(a)



(b)

Figure 10: (a) First frame from image sequence and (b) final hypotheses.

4.3 Segmentation of Tracked Points using a Recursive Structure From Motion Model

Finally, we have applied our segmentation scheme to segment tracked points using a rigid-body motion model. We use the recursive estimation approach to estimating Structure From Motion (SFM) of Azarbayejani and Pentland [2]. This method formulates the estimation problem as an instance of an Extended Kalman Filter (EKF) which solves for the optimal estimate of the relative orientation and pointwise structure of an object at each time step, given a set of explicitly tracked features which belong to a single coherent object. Our extension is thus to handle the multiple-motion case, where there are an unknown number of rigid-body motions in the scene.

First we briefly review the recursive structure from motion model presented in [2]. This formulation embeds the constraints of the well-known relative orientation approach to structure from motion [13, 14] directly into an EKF measurement relationship. Motion is defined as the 3-D translation and rotation of the camera (or object) with respect to the first frame in the sequence. Similarly, structure is represented as the 3-D locations of points seen in the first camera frame. Since features are chosen from the image, there is actually only one unknown parameter for each feature point: the depth along the perspective ray established in the first camera frame.

The state vector for the EKF contains six motion parameters and N structure parameters:

$$\mathbf{x}(t) = \left(\mathbf{t}_{CO}^T(t), \rho^T(t), c_1, \cdots, c_N\right)^T \tag{31}$$

where c_j is the depth for point j and the rotational vector $\rho(t)$ contains three Euler angles which describe the (small) rotation between time $t - \Delta t$ and time t. After every time step, the small rotation is composed into a global rotation quaternion which is maintained external to the EKF.

Appendix C presents the derivation of the EKF measurement and update equations, which compute the best linear least squares estimate and the error covariance. Figure 11 shows the results of this method applied to a moving soccer ball, when appropriate features were selected and tracked by hand.

Given a sequence of features with an unknown number of motions and unknown segmentation, we apply our hypothesize, test, and select method. We generate hypotheses by taking spatially localized random samples of sets of features, and using those to compute a motion estimate. We then compute a support map for each hypothesis; the residual error function is refined to be the difference between the predicted feature locations for each tracked feature and the feature locations actually observed. This can be computed by estimating a structure parameter c given the motion parameters in the hypothesis, and then generating the synthetic feature track predicted by those motion and structure parameters. We use the sum of squared differences of the velocities (instantaneous differences) between the predicted feature track and the observed feature track. The predicted feature track is computed by fixing the motion parameters (setting their initial covariances to 0) and running the SFM estimator.

We show results of this method on two examples with multiple rigid-body motions.

- Figure 12 shows the construction of a synthetic sequence of feature tracks from two rotating spheres. For this example, we generated 200 hypotheses, taking random samples of 4 feature tracks. Figure 13 shows the support maps for 8 of these hypotheses. Many of the initial hypotheses yield degenerate motion/structure estimates, and accordingly have few supported points, since their initial support will have come from

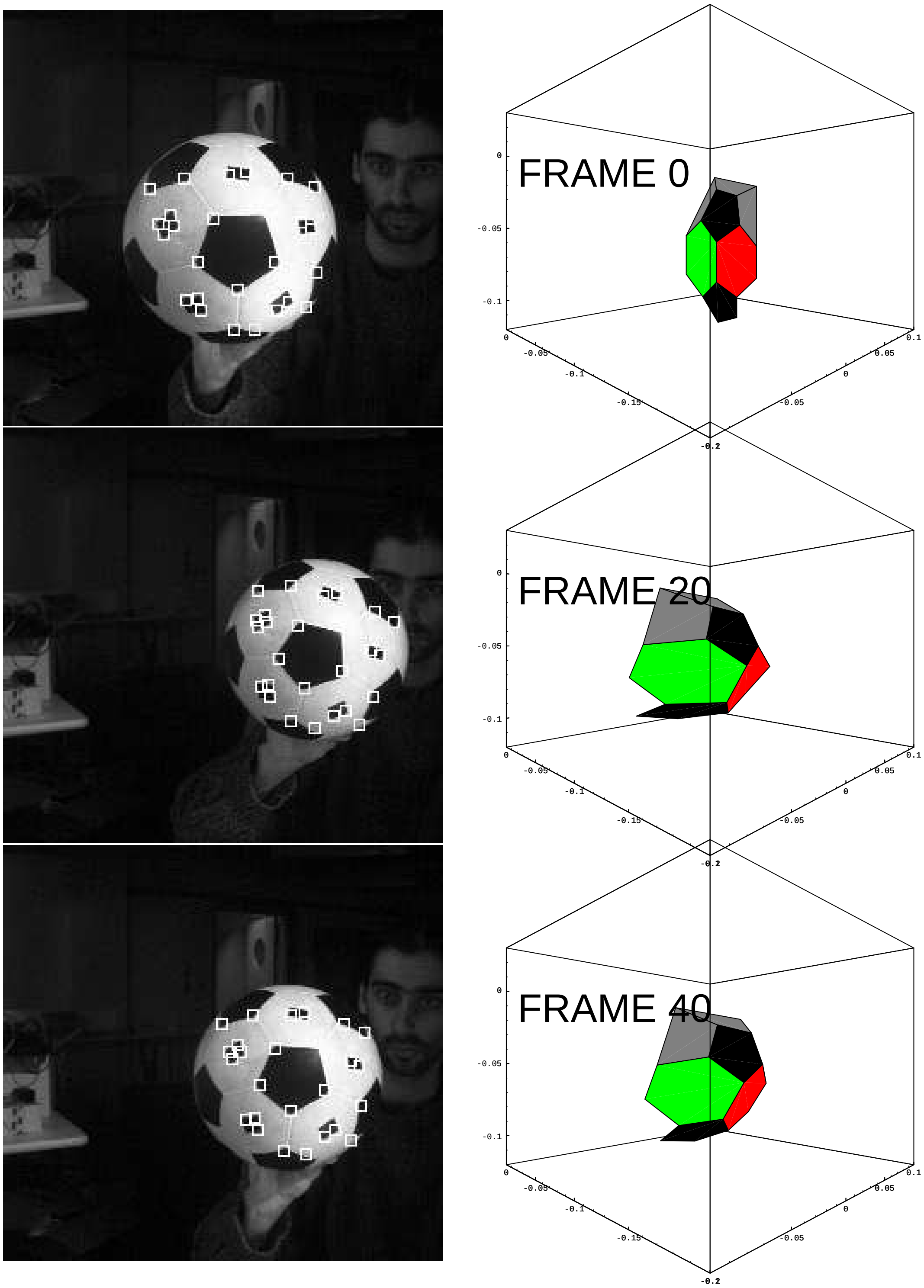


Figure 11: Image with (hand-selected) features and recovered 3-D structure. From [2].

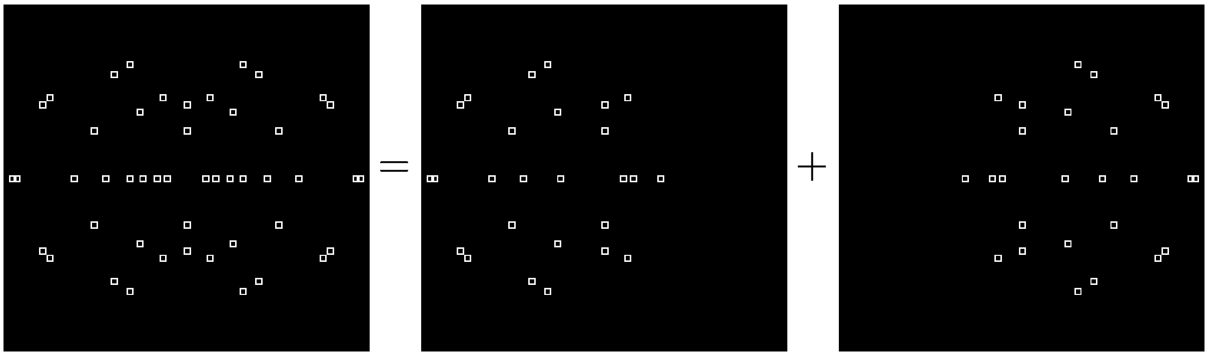


Figure 12: Construction of synthetic sequence with two rigid motions, and no static segmentation cues. Dots were placed on two spheres with different rotations; in the center image the sphere is rotating right, in the right image the sphere is rotating up. The left image shows the combined sequence (only the first frame is displayed)

heterogeneous or degenerate feature sets. However a few will exist that have initial support that covers only a single object (with known probability). These will have motion estimates that coarsely match the actual object motion, and will consequently have support maps that cover a majority of the object. The selection method chose two such hypotheses, which corresponded to the two original motions used to synthetically create the sequence: their support maps are shown in Figure 14.

- Figure 15 shows a real image sequence with two motions (egomotion and the motion of the soccer ball). Features were tracked on this sequence using the (human-assisted) method described in [2]. Hypotheses were constructed by taking random selections of 4 points and computing a motion estimates based only on these points. Support maps were then computed as described above, to see which other points in the scene agreed with the hypothesized motion. 500 hypotheses were generated; Figure 16 shows 8 of these support maps. The selection stage chose two hypotheses, corresponding to the actual motions in the scene. Figure 17 shows the support after refinement using the cooperative update rule Eq. (23).

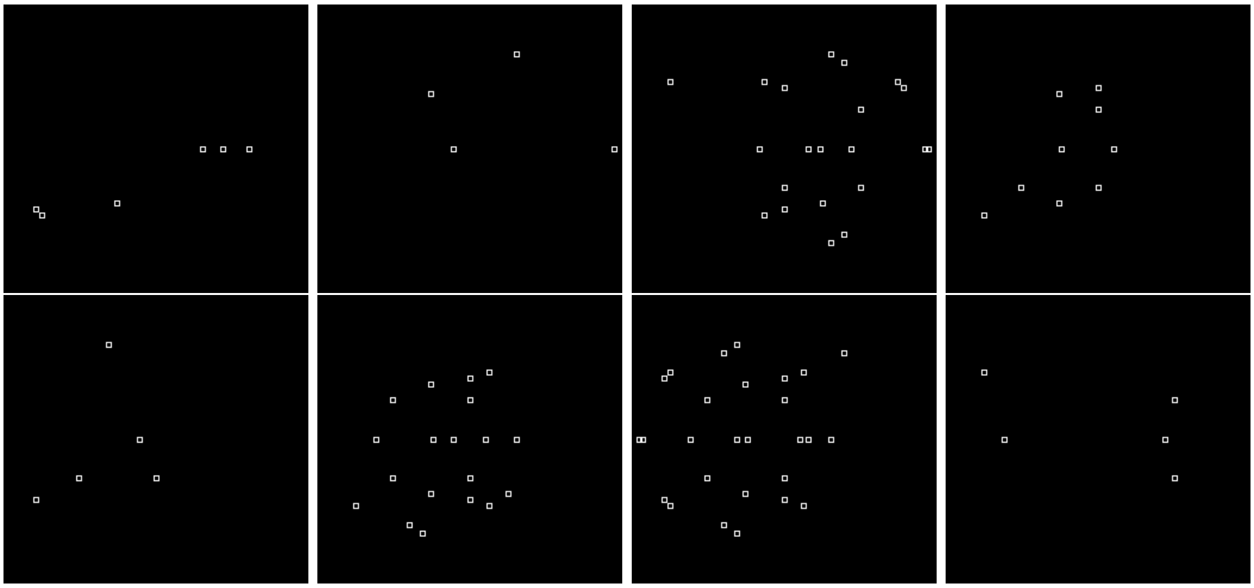


Figure 13: Eight (of 200) support maps for rigid-body motion hypotheses. Each hypotheses was based on computing a motion estimate using 4 randomly selected features from the sequence. Most solutions are degenerate, but a few have "locked on" to a real object in the scene.

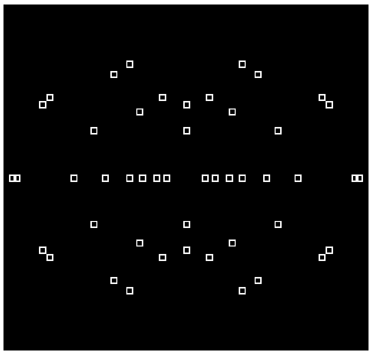
(a)  $\longrightarrow$ (b) 

Figure 14: (a) Input sequence and (b) Selected hypotheses for synthetic rotation example

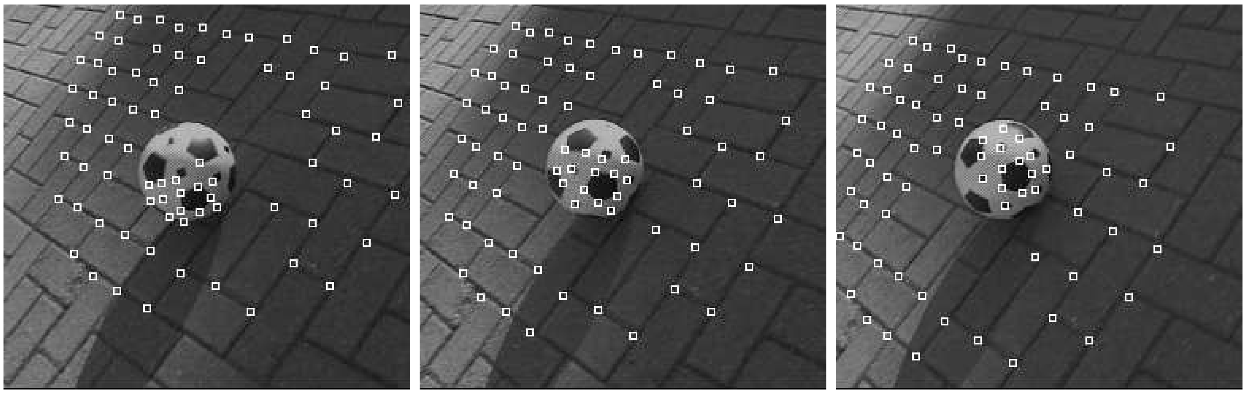


Figure 15: Three frames from a sequence of intensity images with tracked features from a image sequence with two motions: a rotating soccer ball and a moving ground plane due to camera motion.

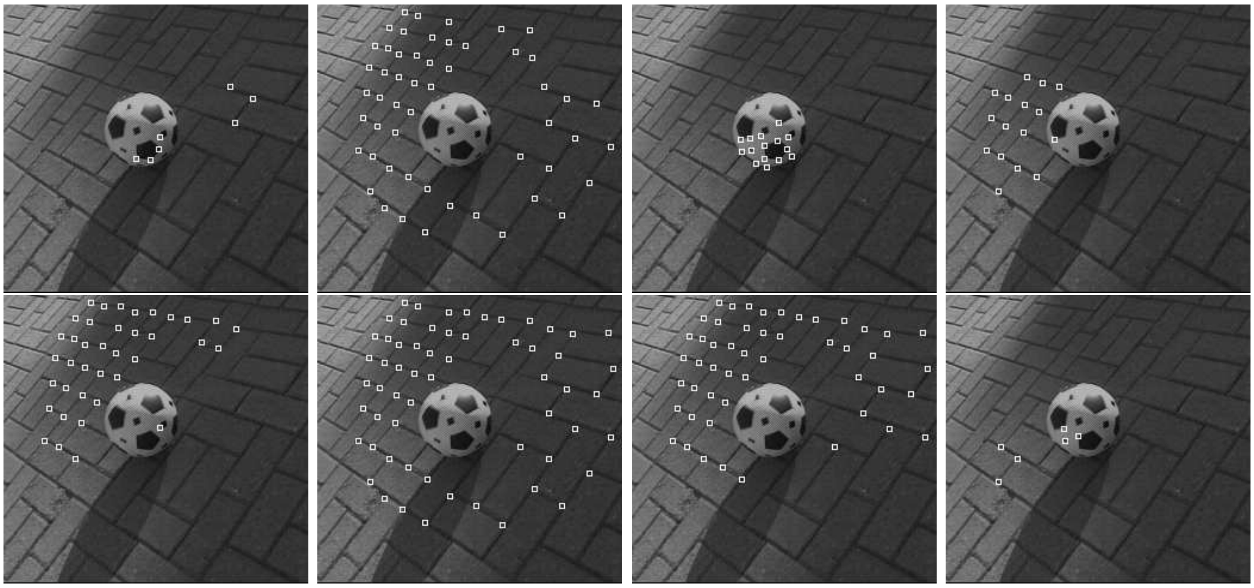
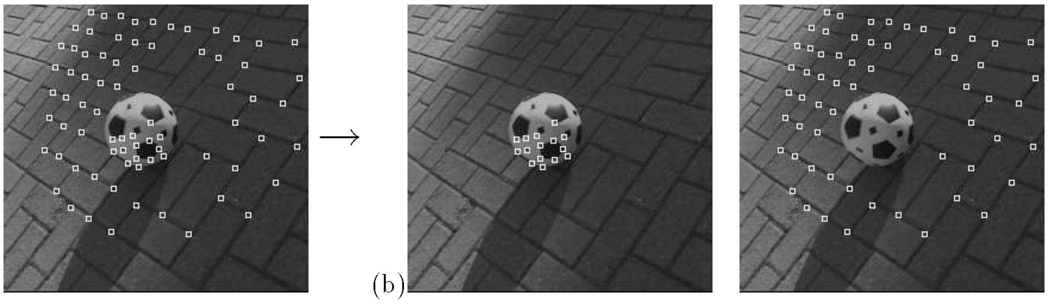


Figure 16: Eight support maps for different rigid-body motion hypotheses for the rolling ball sequence. 500 such support maps were generated in total, by randomly selecting features from the sequence, making a motion estimate with them, and then thresholding the residual error of the predicted feature locations for all other features. Since the random selection process will usually generate hypotheses which intermingle features from more than one object most solutions are degenerate, but a few are based on homogeneous initial features and have "locked on" to a real object in the scene.



(a) (b)

Figure 17: (a) Input sequence and (b) Final Hypotheses for rolling soccer ball example

5 Conclusion

We have presented an approach to the problem of describing images that contain multiple objects or surfaces. We have argued that edge-based representations are insufficient for images that contain transparent or complex occlusion phenomena, and instead propose a multi-layer estimation framework which uses support maps to represent the segmentation of the image into homogeneous (but possibly disconnected) chunks. This support-based framework is an extension of the M-estimation methods found in the robust estimation literature, which deal with the use of a single support map (inverse outlier map). Our method utilizes many of these estimators in parallel, and the Minimum Description Length principle to decide how many robust estimators are needed to describe a particular image.

A major advantage of this approach is that it allows the simultaneous use of multiple models within a robust estimation framework. The description length formulation can consider an arbitrary number of candidate support maps, each generated by a robust estimator with a different initial condition or a different model. We have found that minima of the description length function correspond to perceptually salient segmentations of the image, and demonstrated our results on shape and motion segmentation tasks.

6 Acknowledgment

The authors would like to acknowledge the anonymous reviewers of this paper, whose comments greatly helped the clarity and consistency of the presentation.

A Convergence of Selection Method

Under Eq. (19), $S(\mathbf{d}|\mathbf{a})$ will converge to a local maxima. We can see this by noting the equality

$$\frac{dS(\mathbf{d}|\mathbf{a})}{dt} = \sum_i \frac{dS(\mathbf{d}|\mathbf{a})}{df_i} \frac{df_i}{dt} \tag{32}$$

and substituting $\frac{dS(\mathbf{d}|\mathbf{a})}{df_i}$ via Eq. 19,

$$\frac{dS(\mathbf{d}|\mathbf{a})}{dt} = \sum_i C \frac{da_i}{dt} \frac{df_i}{dt} \tag{33}$$

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

$$\frac{da_i}{dt} = \frac{d\sigma^{-1}(f_i)}{dt} = \frac{d\sigma^{-1}(f_i)}{df_i} \frac{df_i}{dt} \tag{34}$$

and thus substituting back into Eq. 33

$$\frac{dS(\mathbf{d}|\mathbf{a})}{dt} = \sum_i C \frac{d\sigma^{-1}(f_i)}{df_i} \left(\frac{df_i}{dt} \right)^2 \tag{35}$$

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

$$C \frac{d\sigma^{-1}(f_i)}{df_i} \left(\frac{df_i}{dt} \right)^2 \geq 0 \quad \forall i \tag{36}$$

and thus $\frac{dS(\mathbf{d}|\mathbf{a})}{dt}$ will never decrease under Eq. 19. Since the values of f_i are bounded, $S(\mathbf{d}|\mathbf{a})$ is bounded, and thus must converge.

B Extended Kalman Filter for Rigid-body Structure from Motion

B.1 Single motion formalism

The following presentation of the EKF is adapted from [2]. Given the state vector described by Eq. (31), the linear dynamics model is

$$\mathbf{x}(t+\Delta t) = \mathbf{\Phi}(\Delta t)\mathbf{x}(t) + \xi(t) \tag{37}$$

where

$$\mathbf{\Phi}(\Delta t) = \begin{pmatrix} I & 0 & I\Delta t & 0 & 0 \\ 0 & I & 0 & I\Delta t & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{pmatrix} \tag{38}$$

and $\xi(t)$ is an error term, modeled as multi-variate Gaussian distributed white noise.

The observation at each time step is a set of N 2D measurements which is a nonlinear function of the state vector:

$$\mathbf{y}(t) = \begin{pmatrix} \mathbf{y}_1(t) \\ \vdots \\ \mathbf{y}_N(t) \end{pmatrix} = \mathbf{h}(\mathbf{x}(t)) + \eta(t) \tag{39}$$

The imaging geometry outlined in Section 2 (see [2] for details) define the nonlinear function $\mathbf{h}(\mathbf{x}(t))$, with an uncertainty term $\eta(t)$, modeled as Gaussian distributed white noise.

The notation $\hat{\mathbf{x}}(t_1|t_2)$ represents the linear least squares estimate (LLSE) of $\mathbf{x}(t_1)$ based upon $\mathbf{y}(\tau), \tau \leq t_2$. The notation

$$\mathbf{P}(t_1|t_2) \triangleq E(\mathbf{x}(t_1) - \hat{\mathbf{x}}(t_1|t_2))(\mathbf{x}(t_1) - \hat{\mathbf{x}}(t_1|t_2))^T \tag{40}$$

represents the error covariance of the LLSE.

The EKF update equations are

$$\hat{\mathbf{x}}(t|t) = \mathbf{\Phi}(\Delta t)\hat{\mathbf{x}}(t|t-1) + \mathbf{K}(t)\left(\mathbf{y}(t) - \mathbf{h}(\hat{\mathbf{x}}(t|t-1))\right) \tag{41}$$

$$\mathbf{P}(t|t) = \left(\mathbf{I} - \mathbf{K}(t)\mathbf{C}(t)\right)\mathbf{P}(t|t-1) \tag{42}$$

where the *Kalman gain* is defined as

$$\mathbf{K}(t) \triangleq \mathbf{P}(t|t-1)\mathbf{C}^T(t)(\mathbf{C}(t)\mathbf{P}(t|t-1)\mathbf{C}^T(t) + \mathbf{R}(t))^{-1} \tag{43}$$

and

$$\mathbf{C}(t) \triangleq \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right]_{\mathbf{x}=\hat{\mathbf{x}}(t|t-1)} \tag{44}$$

$$\mathbf{R}(t) = E\eta(t)\eta^T(t), \quad E\eta(t) = \mathbf{0}. \tag{45}$$

The prediction step is

$$\hat{\mathbf{x}}(t+1|t) = \mathbf{\Phi}(\Delta t)\hat{\mathbf{x}}(t|t) \tag{46}$$

$$\mathbf{P}(t+1|t) = \mathbf{\Phi}(\Delta t)\mathbf{P}(t|t)\mathbf{\Phi}(\Delta t)^T + \mathbf{Q}(t) \tag{47}$$

where

$$\mathbf{Q}(t) = E\xi(t)\xi^T(t), \quad E\xi(t) = \mathbf{0}. \tag{48}$$

B.2 Robust Formalism

Given a support map, we need to make an estimate of structure and motion parameters based on only those features in the subset specified by the support map. In practice we can do this constructing a new filter with only the state variables corresponding to the set of supported features. However there is also a direct way to do this in the full EKF formulation, by adjusting the measurement noise covariance term $\mathbf{R}$. (This approach also has the advantage that it can be more easily generalized to the case of non-boolean support.)

For each feature track, $\mathbf{R}$ specifies the confidence of the measurement, and weights the contribution of that feature to the estimation process through the Kalman gain matrix. When $\mathbf{R}$ is 0, there is implicitly no noise in the measurement, and the feature vector contributes fully to the estimation. If $\mathbf{R}$ is large, the contribution of the feature vector is minimized.

If we wish to specify that a feature track is an outlier and should not be included in the estimation, we can set its measurement noise model to be infinitely large, and it's contribution to the estimation will be infinitesimal. Thus, if we have a support map $\mathbf{s}$ which indicates which points ought to be supported in the estimate, we can simply set the measurement covariance to be the inverse support: $\mathbf{R} = 1$ where $\mathbf{s} = \mathbf{1}$, and $\mathbf{R} = \infty$ where $\mathbf{s} = 0$, and run the full EKF structure-from-motion estimator. [5]

[5] In this case it can be advantageous to use the "information" form of the Kalman filter, since it is not adversely effected by numerical stability problems with infinite covariances.

References

[1] Adelson E. H., and Anandan, P., "Ordinal characteristics of transparency", in *Proc. AAAI workshop on Qualitative Vision*, pp. 77-81, Boston, MA, 1990.

[2] Azarbayejani, A. J., Horowitz B.,, and Pentland, A. P., Recursive Estimation of Structure and Motion using the Relative Orientation Constraint In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, New York, June 1993.

[3] Attneave, F., "Pragnanz and soap-bubble systems: A theoretical exploration", in J. Beck (Ed.) *Organization and representation in perception.* Hillsdale, New Jersey: Erlbaum, 1992.

[4] Beaton, A. E, and Tukey, J. W. "The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data", *Technometrics*, vol. 16, pp. 147-185, 1974.

[5] Besl, P. J., Birch, J. B., and Watson, L. T., "Robust Window Operators", in *Proc. 2nd. Intl. Conf. Computer Vision*, pp. 591-600, 1988.

[6] Blake, A. and Zisserman, A. *Visual Reconstruction.* Cambridge, MA: MIT Press. 1987.

[7] Darrell, T., Sclaroff, S., and Pentland, P. "Segmentation by Minimal Description", in *Proc. 3rd Intl. Conf. Computer Vision*, pp. 112-116, 1990.

[8] Darrell, T., and Pentland, A. P., "Robust Estimation of a Multi-Layer Motion Representation", in *Proceedings IEEE Workshop on Visual Motion*, pp. 173-177, 1991.

[9] Darrell, T., and Simoncelli, E. P., "Separation of Transparent Motion into Layers using Velocity-Tuned Mechanisms", presented at *Proceedings Assn. Research in Vision and Opth. annual conference (ARVO93)*, available as MIT Media Lab Percom TR-244.

[10] Geman S., and Geman D. "Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images", in *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol 6, pp. 721-741, 1984.

[11] Hochberg, J.E., and McAlister, E. "A quantitative approach to figure goodness", *J. Experimental Psychology*, vol 46, pp. 361-364, 1953.

[12] J. J. Hopfield, and D. W. Tank, Neural computation of decisions in Optimization Problems. *Biological Cybernetics,* Vol. 52, pp. 141-152. 1985.

[13] B. K. P. Horn. *Robot Vision.* MIT Press, 1986.

[14] B. K. P. Horn. Relative orientation. *International Journal of Computer Vision*, 4(1):59–78, January 1990.

[15] Huber, P. *Robust Statistical Procedures* SIAM CBMS-NSF series in Appl. Math., Vol. 27, 1977.

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

[17] Koffka, K. *Principles of Gestalt Psychology.* New York: Harcourt, Brace & World, 1935.

[18] Leclerc, Y., "Constructing Simple Stable Descriptions for Image Partitioning", *Intl. J. Computer Vision*, Vol 3, pp.73-102, 1989.

[19] Leclerc, Y., "Region Grouping using the Minimum-Description-Length Principle", in *DARPA Image Understanding Workshop Proceedings*, pp. 473-479, 1990.

[20] Leeuwenberg, E.L.J, "A perceptual coding language for visual and auditory patterns", *American Jour. of Psychology*, Vol 84, pp. 307-399, 1971.

[21] Li, G. "Robust Regression", In D.C. Hoaglin, F. Mosteller and J.W. Tukey (Eds.) *Exploring Data, Tables, Trends and Shapes.* New York: John Wiley & Sons, pp. 281-343, 1985.

[22] Marroquin, J. L., "Random Measure Fields and the Integration of Visual Information", *IEEE Trans. Sys. Man., Cybernetics*, vol. 22, pp. 705-716, 1992.

[23] Meer, P., Mintz, D., and Rosenfeld, A., "Robust regression methods for computer vision: A review", *Intl. J. Computer Vision*, vol 6, pp. 60-70, 1991.

[24] Nitzberg, M., and Mumford, D., "The 2.1-D Sketch", *Proc. 3rd Intl. Conf. on Computer Vision*, pp. 138-144, 1990.

[25] Pentland, A., "Part Segmentation for Object Recognition", *Neural Computation*, vol 1, pp. 82-91, 1989.

[26] Pentland, A., "Automatic recovery of deformable part models", *Intl. J. Computer Vision*, vol 4, pp. 107-126, 1990.

[27] Pentland, A., and Sclaroff, S., "Closed form solutions for physically based shape modeling and recognition", *IEEE Trans. Pattern Anal. and Machine Intell.*, vol 13, pp. 715-729, 1991.

[28] Poggio, T., Torre, V., and Koch, C. "Computational vision and regularization theory", *Nature*, vol. 317(26), 1985.

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

[30] Press, W., et. al; *Numerical Recipes in C.* New York: Cambridge University Press, 1988.

[31] Shannon, C.E. *The Mathematical Theory of Communication*, Urbana: U. Illinois Press, 1949.

[32] Terzopoulos, D. "The computation of visible surface representations", *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 10:4, 1988

[33] Rissanen, J. "A universal prior for integers and estimation by minimum description length", *The Annals of Statistics*, vol 11(2), pp. 416-431, 1983.

[34] Rissanen, J. *Stochastic Complexity in Statistical Inquiry.* World Scientific: Teaneck NJ, 1989.

[35] Simoncelli, E., Adelson, E. H., and Heeger, D. J., "Probability Distributions of Optical Flow", *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1991.

[36] Wang, J., and Adelson, E. H., "Layered Representations for Image Sequence Coding", to appear in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1993.

[37] Wallace, C.S., Boulton, D.M., "An Information Measure for Classification", *Computing Journal*, vol. 11(2), pp. 185-195, 1968.