A New Approach to Image Feature Detection with Applications[1]

B. S. Manjunath
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106-9560
manj@ece.ucsb.edu

C. Shekhar
R. Chellappa
Center for Automation Research
University of Maryland, College Park, MD 20742
chella@engr.umd.edu

Abstract: Image feature detection is a fundamental issue in many intermediate level vision problems such as stereo, motion correspondence, image registration, and object recognition. In this paper we present an approach to feature detection based on a scale-interaction model. This feature detector is responsive to short lines, line endings, corners and other such sharp changes in curvature. We provide extensive experimental results to demonstrate its potential applications to several image analysis problems.

1. This research was partially supported by grants DARPA No. 6989, DACA 76-89-C-0019 and a grant from UCSB academic senate.

1 Introduction

Feature detection is an important early vision problem. Previous work on feature detection include the use of grey level statistics (e.g.: Moravec's operator [1],[2]) and the detection of edges and corners [3]. Methods based on detecting edges and corners are particularly useful in applications such as analysis of aerial images of urban scenes, airport facilities, image to map matching, etc. Algorithms based on grey level statistics are applicable to a wider variety of images such as desert scenes and vegetation, which may or may not contain any man-made structures. Features, by definition, are locations in the image that are *perceptually interesting*. One can characterize an image feature detection algorithm by two attributes -- (a) Generality, and (b) Robustness. Given that the nature of salient features vary from application to application, it is desirable that a feature selection algorithm be as general as possible. In case of structured objects such features could be corners and locations with significant curvature changes. When analyzing human faces, features of interest could be the eyes, nose, mouth, etc. The generality criterion addresses the issue of whether a given feature detection algorithm can be used in a wide variety of applications.

The second criterion, that of robustness, is equally important in applications such as image registration. A feature detection algorithm can be considered robust if it identifies the same feature locations independent of rotation and translation, as well as minor scaling and perspective deformations. Most feature detection schemes which obtain a symbolic representation in terms of edges and corners are not quite general, whereas it has been observed that general purpose feature detection algorithms such as the Moravec operator or its variants are not robust [4]. The method we describe below is both robust and of general utility, and has been tested successfully on several wide-ranging applications. A third attribute of our scheme is that it provides a simple representation mechanism as well, and this is useful in applications such as human face recognition.

The model we describe in Section 2 is in part motivated by our earlier work on texture image segmentation [5]. It is based on the observation that certain textures (such as the classical L - + texture) have no orientation or scale preference, and differ only in the distribution of line endings and intersection. This in turn led to the scale interaction model we proposed to detect these features in [5]. In addition to texture discrimination, this model was used to explain the perception of certain types of illusory contours. It is interesting to note that there are cells in the visual cortex which also exhibit sensitivity to line endings, and are called endstopped cells or hypercomplex cells.

During the course of this work we became aware of the related work by Dobbins et al [6],[7] on modeling the endstopping cells and their extensive simulation studies on relating endstopping with curvature detec-

tion. Although our scale interaction model for feature detection was developed independently, there are many commonalities that one can observe between our model and the one in [7]. In particular, both these models are based on the observation that curvature response of the feature detectors results from the difference of two low-pass responses of different bandwidths. These models are quite non-linear and not amenable for easy mathematical analysis. The empirical studies performed by Dobbins et al. agree well with our own simulations in relating curvature with the feature responses, and the main focus of this paper is to demonstrate the robustness of the proposed feature detection mechanism as we explore its applications to several image analysis problems.

In demonstrating the utility of this feature detection we have chose three different applications: The first one concerns the image registration problem which is a classical problem in image analysis. Image registration involves obtaining a correspondence between two or more images of a scene taken from different viewing positions or at different times. Image registration is the first step in many image analysis systems such as geometric stereo, image data fusion, and applications involving template matching. In order to accurately register the images, one need to identify salient image features, and the feature detection algorithm should be robust to rotation and small geometrical distortions. In this context, our approach to feature detection has been tried successfully on hundreds of images with no parameter tuning [8].

Another interesting application we investigate is that of face recognition [9]. In this case, we use the feature information at the salient image locations to represent face images as topological graphs, and use a simple graph matching algorithm for recognition. A variation of this approach is also used in the third application, that of motion tracking.

This paper is organized as follows: In the next section we introduce our scale interaction model for feature detection and representation based on the Gabor wavelet transform. In Section 3 application to image registration, face recognition and motion correspondence are presented along with detailed experimental results. Section 4 presents the conclusions.

2 The Scale Interaction Model for Feature Detection

Our formulation of the feature detection model is based on filtering using a class of self-similar Gabor functions or Gabor wavelets. Gabor functions are Gaussians modulated by complex sinusoids. In its general form, the 2-D Gabor function and its Fourier transform can be written as [10]:

$$g(x, y\ ; u_0, v_0) \;=\; \exp\left(-\left[\frac{x^2}{2\sigma_x^2} + \frac{y^2}{2\sigma_y^2} \right] + 2\pi i\,[u_0 x + v_0 y] \right) \tag{1}$$

$$G(u, v; u_0, v_0) \;=\; \exp\left(-2\pi^2 \left(\sigma_x^2 (u - u_0)^2 + \sigma_y^2 (v - v_0)^2 \right) \right) \tag{2}$$

σ_x and σ_y define the widths of the Gaussian in the spatial domain and (u_0, v_0) is the frequency of the complex sinusoid. A well known property of these functions is that they achieve the minimum possible joint resolution in space and frequency [10]. A signal such as a delta function which is concentrated at a point in space has no frequency localization. Likewise, a delta function which is concentrated in frequency, has no spatial localization. A good measure of localization in the two domains is given by the product of the bandwidths in space and frequency. The effective bandwidth of a signal is defined as the square root of the variance of the energy of the signal. Let δx and δy be the effective widths of the signal in the horizontal and vertical directions in space respectively and δu, δv the corresponding widths in frequency. Then the following inequalities (also called the uncertainty relations) hold: (a) $\delta x\ \delta u \geq 1/(4\pi)$, and $\delta y\ \delta v \geq 1/(4\pi)$. Gabor functions are unique in attaining the minimum possible value of this joint uncertainty.

Gabor functions form a complete but non-orthogonal basis set and any given function $f(x, y)$ can be expanded in terms of these basis functions. Such an expansion provides a localized frequency description and has been used in image compression [11],[12], face recognition [9],[13] and texture analysis [5],[14]. Local frequency analysis, however, is not suitable for feature representation as it requires a fixed window width in space and consequently the frequency bandwidth is constant on a linear scale. However, in order to optimally detect and localize features at various scales, filters with varying support rather than a fixed one are required. This would suggest a transformation similar to wavelet decomposition rather than a local Fourier transform. We now consider such a wavelet transform where the *basic wavelet* is a Gabor function of the form:

$$g_\lambda(x, y, \theta) \;=\; \exp\left(-\left(\lambda^2 x'^2 + y'^2 \right) + i\pi x' \right) \tag{3}$$

$$x' \;=\; x\cos\theta + y\sin\theta$$

$$y' \;=\; -x\sin\theta + y\cos\theta$$

where λ is the spatial aspect ratio and θ is the preferred orientation. To simplify the notation, we drop the subscript λ and unless otherwise stated assume that $\lambda = 1$. The family of basis functions corresponding to the basic wavelet in (3) is obtained by translations and dilations of $g(x, y, \theta)$. For practical applica-

tions, discretization of the parameters is necessary. The discretized parameters must cover the entire frequency spectrum of interest. Let the orientation range $[0, \pi]$ be discretized into N intervals and the scale parameter α be sampled exponentially as $\alpha^j, j \in Z$. This results in the wavelet family

$$\left(g\left(\alpha^j (x - x_0, y - y_0), \theta_k \right) \right), \alpha \in \Re, j = \{0, -1, -2, \ldots\} \quad \textbf{(4)}$$

where $\theta_k = (k\pi)/N$. Then we define the transform by

$$W_j(x, y, \theta) = \int f(x_1, y_1)\, g^*\left(\alpha^j (x - x_1, y - y_1), \theta \right) dx_1 dy_1 \quad \textbf{(5)}$$

At each resolution in the representation hierarchy these wavelets localize the information content in both frequency and spatial domains simultaneously. Any desired orientation selectivity can be obtained by controlling the parameter θ.

2.1 Feature Detection and Localization

We now define the response of the feature detector, denoted by $Q_{ij}(x, y, \theta)$ at location (x, y) with preferred orientation θ as:

$$Q_{ij}(x, y, \theta) = f(W_i(x, y, \theta) - \gamma W_j(x, y, \theta)) \quad \textbf{(6)}$$

where $\gamma = \alpha^{-2(i-j)}$ is the normalizing factor, and $f(.)$ is a non-linear transformation function (e.g.: a sigmoid function). Locations (x, y) in the image which are identified as feature locations satisfy

$$Q_{ij}(x, y, \theta) = \max_{(x', y') \in N_{xy}} Q_{ij}(x', y', \theta) \quad \textbf{(7)}$$

N_{xy} represents a local neighborhood of (x, y) within which the search is conducted. Computing the feature response $Q_{ij}(x, y, \theta)$ can be thought of as a two stage process: In the first stage the image data is convolved with filters at two different scales to extract the first level *simple features.* These represent lines and edges at the two scales in the image. The second stage involves taking a difference of these simple features and performing a non-linear transformation, which results in the scale-interaction model. The difference of the filtered outputs is similar to a difference of Gaussian filter (except for the orientation tuning), and will have a positive middle region and negative end zones (see Figure 1). This makes them responsive to start line segments, line endings, and in general changes in curvature.

In a simplistic way, such a processing can be related to the behavior of endstopped (hypercomplex) cells in the visual cortex. The original idea of using such scale interactions dates back to the early work of Hubel

and Wiesel [15]. As mentioned earlier, a similar approach using difference of responses from filters of different bandwidths have been explored in [7], where the authors focus on the relationship between endstopping behavior and curvature representation. They provide extensive simulation results indicating curvature selectivity of endstopped cells, and suggest that in biological systems these cells provide a representation for curves. The activities of these cells represent curvature changes at different spatial scales, thus representing in some sense a *curvature primal sketch* [16]. It should be noted that unlike parametric models often used in computer graphics and image processing for representing curves and curvature singularities, this approach does not involve any underlying image model. However, the non-linearities inherent in these models make it difficult to provide a detailed mathematical analysis.

It has been suggested that endstopped cells help in localizing texture boundaries. We have provided a clear demonstration of their role in texture boundary perception in [5]. The role of these cells in illusory contour perception is discussed in [17],[18], and some of the observations are also used in our model for boundary detection in [5]. That these cells respond to line-ends is nothing but one extreme example. This is further illustrated in Figure 1.

Figure 2 illustrates the observation that the feature locations correspond to points with significant curvature changes. All the corners in this hand-drawn hammer picture are located by the algorithm, although only one particular set of parameters is used for the scales. Figure 3 shows the types of features that are detected on face images. Information at these locations is used in the recognition process and will be discussed in detail in the following. We now discuss in more detail applications to face recognition and in estimating motion parameters by tracking features over a sequence of images.

3 Applications

3.1 Image Registration

Figure 4 shows an example where there are no well defined structures to obtain a registration. The two images shown are part of a sequence of images of Mojave desert taken from a camera attached to a balloon. This setup was used to simulate the Mars'94 project where one of the goals is to measure the 3-D wind velocity on Mars surface. The proposal is to use a downlooking camera attached to a balloon to measure the motion of the balloon (by using the image sequence) and hence determine the wind velocity. Since there are no significant structures, registering such images is a challenging and difficult problem. Tradi-

tional solutions to this problem are unreliable when the rotation and scale change between the two frames is significant.

A computational vision approach to solve this registration is proposed in [8],[19] using the feature detection algorithm described earlier. A small number of feature points are located in the two images from the sequence using the algorithm described in the previous section. Typically, 20-40 features are identified per image. The figure shows these feature locations (marked with +s) superimposed on the original images. The rotation between the two frames is estimated using an illuminant direction estimation method [20]. By estimating the illuminant direction in each frame, we can estimate the rotation between the two frames and simplify the matching process. Since the common area between the two frames can be much smaller than the image field, and in addition there may be scaling between successive frames, methods based on correlation matching become unreliable. An initial estimate of translation and scaling is obtained by pair-wise matching of the detection feature points. Subsequently, a hierarchical correlation matching is performed to obtain an accurate camera motion estimate.

For more details about this registration scheme the reader is referred to [8],[19]. It has been tested on many different data sets including stereo image pairs and satellite image data. This application illustrates the robustness of this feature detection method in identifying a consistent set of features irrespective of significant amounts of rotation, scaling and perspective distortion between pairs of images.

3.2 Human Face Recognition

3.2.1 Previous work on Face Recognition

Human faces provide a very good example of a class of natural objects which do not lend themselves to simple geometrical representations, and yet the human visual system does an excellent job in efficiently recognizing these images. Considerable research has been done in developing algorithms to solve this problem. A comprehensive survey of computer recognition of faces can be found in [21],[22]. Most of this work is either recognition by using facial profiles (for example, see [23],[24]) or using the frontal views. In this paper we are interested in the latter case where the input is an intensity image of the frontal view of a face. Previous related work can be found in [25], the WISARD system ([26],[27]) and the dynamic link architecture for face recognition [28]. One of the early systems built for this task is described in [29]. The system automatically localizes features such as corners of the eyes, nostrils, mouth etc. Then a set of sixteen facial parameters corresponding to these features is computed. They correspond to ratios of distances and areas, and angles to compensate for scaling differences. A simple Euclidean distance measure is then

used to compute the similarity between a test face and a stored face. The best case performance of the system was 15 correct identifications out of 20 test faces. The test data differed from the training data in that there was a period of one month between the acquisition of the samples; in both cases a full frontal view was used.

In [30],[31] the authors describe a real time face recognition system using the Karhunen-Loeve Transform. Their system tracks a person's head and identifies the face by comparing its features with a known database. The basic idea is to find a low dimensional feature space to represent the intensity data, for which they use principal component analysis. Since intensity data is directly used in the recognition process, such a system will be prone to local fluctuations in the image. This approach to recognition is similar to many earlier attempts in transforming a 3-D recognition problem to a 2-D matching, without detecting any perceptually significant features. See for example the associative memory models for face detection ([32],[33],[34])

Our approach to face recognition is somewhere in between the two extremes of using the raw intensity data and using the high level face feature information. It is feature based, but does not depend on the explicit use of high level facial features. As our experimental results indicate, it is robust to facial distortions and changes in facial expressions while being computationally less expensive.

3.2.2 Representation and Recognition Using Graphs

Given a face image, salient feature points are detected using the scheme outlined in the previous section. In the following discussion we will assume that these features are detected using a specified pair of scales in the end-inhibition model, and to simplify the notation we drop the subscripts *i* and *j* in (6). Information about the faces is represented using the available information at the feature points, for which we use topological graphs. For convenience the features detected in a given image are numbered $\{1, 2, \ldots\}$ in any arbitrary, but consistent way. Corresponding to each feature point *i* in the image there is an associated node V_i in the graph. Each node V_i is characterized by the pair $\{X_i, \boldsymbol{q}_i\}$, where $X_i = (x_i, y_i)$ represents the spatial location, and

$$\boldsymbol{q}_i = [Q(x_i, y_i, \theta_1), \ldots, Q(x_i, y_i, \theta_N)] \quad \textbf{(8)}$$

is the feature vector corresponding to the *i*th feature. Let N_i denote the set of neighboring feature points of the *i*th node. Directional edges connect the neighbors in the graph (i.e., the neighborhood is not symmetric). The neighborhood of a node is determined by taking into account both the maximum number of neighbors allowed as well as the distance between them. The Euclidean distance between two nodes V_i

and V_j is denoted by d_{ij}. The resulting graph structure forms our representation of a face image. Thus each face in the database is represented as a labeled graph. Recognizing an input face image then involves converting it first into a graph representation using the method described above (and we refer to this graph as the input graph I) and obtaining a match with one of the stored graphs.

Matching the input with an object from the database is an important research issue by itself. In the following, however, we are interested in evaluating the feature representation mechanism for face recognition rather than in developing an optimal search scheme. Matching the input graph with a stored one is formulated as an optimization problem involving minimization of a cost function. The cost function has two parts, one measuring the similarity between matched features and the other corresponding to the topology of the features. The algorithm involves local search, is deterministic in nature and extremely fast. The algorithm, however, does not guarantee optimizing the criterion function. In spite of this the recognition rate is comparable to that of most face recognition schemes that we are aware of, demonstrating further the robustness of our feature extraction. Although this implementation calls for matching the input graph with each one of the stored graphs (hence linear search complexity in the number of stored objects) we note that this can be implemented easily on a parallel hardware (for example, artificial neural networks). Our implementation of the matching algorithm is given below:

In the following, subscripts *i, j* refer to nodes in the input graph I, and *i*', *j*', *m*', *n*' correspond to nodes in a stored graph O.

1. The input graph I is spatially aligned with the stored graph O by matching the centroids of the feature set $\{V_i\}$ and $\{V_{i'}\}$.
2. Let R_i be the spatial neighborhood for the *i*th feature in the input graph, over which a search is conducted to find the best matching feature node $V_{i'}$ in the stored graph, such that

$$S_{ii'} = 1 - \frac{\boldsymbol{q}_i \cdot \boldsymbol{q}_{i'}}{\|\boldsymbol{q}_i\| \|\boldsymbol{q}_{i'}\|} = \min_{m' \in R_i} S_{im'} \quad \textbf{(9)}$$

   R_i is typically a circular region of specified radius (seven pixels in our implementation) around the *i*th feature location.
3. After all the individual features are matched, the total cost is computed by taking into account the topology of the matched graphs. Let the nodes *i* and *j* match *i*' and *j*' respectively, and further let $j \in N_i$ (i.e., V_j is a neighbor of V_i. Let $\rho_{ii'jj'} = \min \{d_{ij}/d_{i'j'}, d_{i'j'}/d_{ij}\}$. Then the topological cost for this particular pair of nodes is computed as

$$T_{ii'jj'} = 1 - \rho_{ii'jj'} \quad (10)$$

Note that if the match is perfect, $d_{ij} = d_{i'j'}$ and $T_{ii'jj'} = 0$.

4. The total cost for matching input graph I to a stored graph O is then given by

$$C_1(I, O) = \sum_i S_{ii'} + \lambda_t \sum_i \sum_{j \in N_i} T_{ii'jj'} \quad (11)$$

where λ_t is a scaling parameter which controls the relative importance of the two cost functions.

5. The total cost is then scaled appropriately to reflect the difference in the number of features between the input and stored graphs. If n_I, n_O denote the number of feature nodes in the input and stored graphs respectively, then the scaling factor $s_f = \max\{n_I/n_O, n_O/n_I\}$, and the scaled total cost $C(I, O) = s_f C_1(I, O)$.

6. The best candidate match O^* then satisfies

$$C(I, O) = \min_{O'} C(I, O') \quad (12)$$

Note that the above algorithm does not take into account the topological cost during the matching process. The topological cost is computed only after the features are matched. The advantage is that there are no iterations, and no stochastic elements involved in the search, resulting in a very fast algorithm for matching.

3.2.3 Experimental Results

We have implemented a simple face recognition system based on the above principles. The input is a 128×128 image, having a white background. In our current implementation, the feature responses are computed corresponding to the scale parameters $i = -2, j = -5$ and $\alpha = \sqrt{2}$ in (6). Typical numbers of feature points detected in a face image using (7) vary from 35 to 50. The number of discrete orientations used was $N = 4$ (in (8)) corresponding to $\theta = \{0, 45, 90, 135\}$. One byte of information is stored for each of the components in the feature vector, or approximately 200 bytes of information per face. Compared to the original intensity data of 16K bytes, this results in a considerable savings in storage memory.

The database we have used has face images of 86 persons, with two to four images per person, taken with different facial expressions and/or orientations. Often there is a small amount of translation and scaling as well. There are a total of 306 such face images in the current database. For each face image, the stored information corresponds to the feature graph discussed in the previous section. The neighborhood set N_i of the ith feature node consists of its five nearest neighbors. Note that this set is not necessarily symmetric.

The performance of the system is evaluated as follows: For each entry of a face image in the database, the cost of associating another entry in the database is computed according to (11). The parameter λ_t in (11) is set to 0.2, so as to have equal contributions to the total cost from the similarity measure and the topological cost (as the summation over *j* is over the neighbors, which in our case total five). These costs are then sorted and the best match is the one having the minimum associated cost as in (12). Note that in doing this self-matches (which obviously result in zero total cost) are ignored. The recognition accuracy in terms of the best match corresponding to the right person was 86%, and in 94% of the cases the correct person's face was in the top three candidate matches. The graph matching steps 1 through 5 discussed in Section 3.2.2 typically take less than 0.5 seconds for each graph (on a SUN Sparc 2 workstation). Some results are shown in Figure 5.

In a typical application of this system, one can store 10 to 20 images of each person's face, taken from different angles, with different facial expressions. Any incoming face image can then be matched to this set of images, and a threshold can be associated with the matching cost to either accept a match or to reject. Due to the nature of representation used, the associated memory requirements are minimal. The entire matching process can be implemented on a parallel hardware or connectionist network for real time applications. Among the issues to be addressed for future work are the scale invariance and use of high level feature information.

3.3 Motion Correspondence

The previous application to face recognition illustrated the use of feature information in representing shape. The following application to motion correspondence demonstrates another aspect of this feature detection scheme, that of robustness. While the feature information is directly used in the face recognition case, here we use the entire Gabor wavelet at each feature location (we refer to this as the Gabor jet [13]). The goal here is to extract salient points from a sequence of images, and to obtain the image plane trajectories of these points. This is formulated as a recursive tracking problem, with the dual objective of estimating the motion of the camera, and tracking feature points in the image sequence. The method used for feature point matching is discussed in Section 3.3.1. The motion estimation aspects are discussed in detail in [35], and are summarized in Section 3.3.2.

The problem of motion correspondence is somewhat similar to the face recognition problem in the sense that both require a correspondence between distinct features in two or more images, or between stored patterns and a test pattern. In both cases, labeled graph matching provides the required invariance to limited amounts of distortion, unlike correlation-based methods which are known to be sensitive to distortion.

There are two main conceptual differences between the two applications. Firstly, in motion tracking, it is possible to interleave feature point matching with the recursive estimation of motion parameters. Current 3-D motion information can be used to predict the positions of feature points in the incoming image, thereby reducing the search time for finding match points. The second difference is in the nature of the search: the goal here is not to find the best matching graph from a number of stored graphs, but to extract the graph from the incoming image that best matched the graph obtained for the current image. Feature points are not assumed to have already been extracted in all the images in the sequence; instead, feature points extracted from the first image are *tracked* over successive images in the sequence by graph matching between consecutive image frames.

3.3.1 Matching

Let I_1 and I_2 be two successive images from an image sequence. Feature point matching between I_1 and I_2 is performed using the principles of labeled graph matching and motion coherence (Figure 6). Let us suppose that feature points in I_1 have been extracted using the method described in Section 2, and that we wish to find the matching points in I_2. The feature points are treated as nodes in a labeled graph, where the label vectors, called *jets*, are obtained by convolution with the Gabor wavelet kernels. The jet for a point *i*, denoted by $jet(i)$, is of the form

$$jet(i) = \begin{bmatrix} W_{j_1}(x, y, \theta_1) \\ W_{j_1}(x, y, \theta_2) \\ \vdots \\ W_{j_1}(x, y, \theta_n) \\ W_{j_2}(x, y, \theta_1) \\ W_{j_2}(x, y, \theta_2) \\ \vdots \\ W_{j_2}(x, y, \theta_n) \\ \\ \vdots \\ W_{j_m}(x, y, \theta_n) \end{bmatrix}$$

where the W's are computed as described in Section 2. The number of scales *m* and the number of orientations *n* can be adjusted to obtain the desired performance.

Neighboring feature points in I_1 are linked to form a topological graph, using inter-point distances as the basis for linkage. Matching then consists of dynamically assigning image points in I_2 to the given feature points in I_1, starting from initial positions determined by current position and motion. This assignment is guided by three criteria (a) the similarity of the label vectors of potential match points (b) the preservation of the local topology of the graphs of the feature points in the two images and (c) the closeness of the match points to their *predicted* locations. The matching is treated as minimization of a cost function of the form

$$\sum_i S_{ii'} + \alpha \sum_i \sum_{j \in N_i} T_{ii'jj'} + \beta \sum_i D_{ii'} \quad \textbf{(13)}$$

where $S_{ii'}$ is the similarity cost, $T_{ii'jj'}$ the topological cost, and $D_{ii'}$ is the *diffusion cos*t, all costs being determined locally for the *i*th feature point. The similarity and topological costs are computed as in Section 3.2.2. Computation of the diffusion cost will be explained in the next subsection. The optimal values of the weighting parameters α and β are determined experimentally. If the $S_{ii'}$ or $D_{ii'}$ cost terms for a point are inordinately high after minimization, it is assumed that the point has been *lost* due to occlusion or other causes, and it is not tracked any further.

The actual minimization can be done in several ways. In our implementation, we used a hierarchical approach, first finding the best match at the coarsest resolution, and then refining it at the finer resolutions. This makes effective use of the intrinsic multi-scale nature of the Gabor wavelet representation. At each scale, the best match is found by a locally exhaustive search, the size of the search neighborhood being proportional to the scale.

3.3.2 Interleaving Matching and Motion Estimation

The matching process for the motion correspondence problem is interleaved with the recursive estimation of 3-D motion parameters. Details of the recursive estimation technique used may be found in [35]. They are briefly summarized here.

The motion parameters consist of 3-D feature point positions, camera velocities and camera pose parameters. These are contained in a state vector $\boldsymbol{s}$. The recursive estimator used is the extended Kalman filter, which operates in two steps, a time update or prediction, based on the motion model, followed by a measurement update or filtering, based on the information in the incoming image. In the discussion that follows, *k* refers the index of the incoming image in the sequence. Let the estimate $\hat{\boldsymbol{s}}(k|k-1)$ denote the predicted estimate, just after a time update, and the estimate $\hat{\boldsymbol{s}}(k-1|k-1)$ the filtered estimate, just after

a measurement update. The motion model is represented by a state transition matrix F, while the observation model, or the relationship between the parameters in the state vector and the measured image coordinates of feature points, is represented by a nonlinear function $\boldsymbol{h}$. The position of the ith feature point ρ_i in the incoming image can be predicted as:

$$\hat{\rho}(k|k-1) = \boldsymbol{h}_i[F\hat{\boldsymbol{s}}(k-1|k-1)] \quad \textbf{(14)}$$

where $\boldsymbol{h}_i$ is the portion of the nonlinear measurement $\boldsymbol{h}$ corresponding to the ith feature point. Let $P(k|k-1)$ be the covariance of the predicted state vector at time instant k, and let $H_i(k)$ be the $2 \times d$ linearized measurement function, each row corresponding to one of the two image plane coordinates of feature point i, evaluated at the state estimate $\hat{\boldsymbol{s}}(k|k-1)$.

The covariance, or uncertainty, of the prediction in (14), denoted as $C_i(k|k-1)$, can be shown to be:

$$C_i(k|k-1) = H_i(k)\,P(k|k-1)\,H_i(k) \quad \textbf{(15)}$$

The predicted feature point positions can then be used to initialize the matching process, and the covariance of the prediction to control the *diffusion* or search of match points during the matching. In other words, the search for a match point is conducted in a region around its predicted location, the size of this region being proportional to the uncertainty of the prediction. Further, the diffusion cost term $D_{ii'}$ in equation (13) is chosen so as to favor matches close to their predicted locations. To be precise, it is selected to be the *Mahalanobis* distance from the predicted locations, i.e.

$$D_{ii'} = (\hat{\rho}(k|k-1) - \rho_{i'}(k))^T C_i(k|k-1)^{-1} (\hat{\rho}(k|k-1) - \rho_{i'}(k)) \quad \textbf{(16)}$$

The matching procedure yields the measurements $\rho_{i'}(k)$, which are then used to perform a measurement update on the predicted state vector $\boldsymbol{s}(k|k-1)$. The system is then ready to process the next image in the sequence.

3.3.3 Experimental Results

Experimental results on a real image sequence, called the UMASS Rocket sequence, are presented in this paper. For labeling feature points, Gabor wavelets at four different resolutions $m = 4$ and four orientations $n = 4$ were used. Each feature point was topologically linked with its three nearest neighbors. The 1st, 8th and 16th frames from this sequence are shown in Figure 7. Feature points extracted using scale interactions are shown in Figure 8 (top), and trajectories of selected points are shown in Figure 8 (middle) and (bottom), superimposed on the 1st and 16th frames, respectively.

4 Conclusions

A simple and robust feature detection algorithm is presented here. The motivation behind the algorithm is the observation that salient image feature points are often characterized by changes in local curvature, and the scale interaction model localizes such image information. Unlike parametric models often used in graphics and image processing, we do not assume any underlying image model while detecting the feature points. This model for feature detection has been quite successful in applications including image registration, face recognition and motion correspondence. Our extensive experimental results demonstrate the robustness of this approach to image feature detection.

Acknowledgments: We would like to thank Prof. von der Malsburg for his suggestions during the course of this work and providing access to the USC face image database. Dr. Qinfen Zheng did the experiments on the balloon image sequence. Peter Kroger of JPL provided us with the balloon image sequence, and the UMASS rocket sequence was provided by Rabi Dutta and Raghavan Manmatha.

5 References

[1] H. P. Moravec, "Towards automatic visual obstacle avoidance," in *Proc. 5th Int. Joint Conf. Artificial Intell.*, (Cambridge, MA), p. 584, August 1977.

[2] M. J. Hannah, "Bootstrap stereo," in *Proc. Image Understanding Workshop*, (College Park, Maryland), pp. 201--208, April 1980.

[3] G. Medioni and R. Nevatia, "Matching using linear features," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 6, pp. 675--685, November 1984.

[4] H. Li and S. K. Mitra, "Automatic selection of control points for image registration," Tech. Rep. CIPR 91-16, Center for Information Processing Research, University of California, Santa Barbara, September 1991.

[5] B. S. Manjunath and R. Chellappa, "A unified approach to boundary perception: edges, textures and illusory contours," *IEEE Trans. Neural Networks*, Vol. 4(1), pp. 96--108, January 1993.

[6] A. Dobbins, S. W. Zucker, and M. S. Cynader, "Endstopped neurons in the visual cortex as a substrate for calculating curvature," *Nature*, vol. 329, pp. 438--441, October 1987.

[7] A. Dobbins, S. W. Zucker, and M. S. Cynader, "Endstopping and curvature," *Vision Research*, Vol. 29, No. 10, pp. 1371--1387, 1989.

[8] Q. Zheng, R. Chellappa and B. S. Manjunath, "Balloon motion estimation using two frames," in *Proc 25th Asilomar Conference on Signals, Systems and Computers*, pp. 1057-1061, Pacific Grove, CA, October 1991

[9] B. S. Manjunath and R. Chellappa, "A feature based approach to face recognition," in Proc. IEEE Conference on Computer Vision and Pattern Recognition '92, (Champaign, IL), pp. 373-378, June 1992.

[10] J. G. Daugman, "Uncertainty relations for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *J. Opt. Soc. Am. A*, vol. 2, no. 7, pp. 1160--1169, 1985.

[11] M. Porat and Y. A. Zeevi, "The generalized gabor scheme of image representation in biological and machine vision," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-10, pp. 452--468, July 1988.

[12] J. G. Daugman, "Relaxation neural network for non-orthogonal image transforms," in *Proc. Int. Conf. on Neural Networks*, vol. 1, (San Diego, CA), pp. 547--560, June 1988.

[13] J. Buhmann, J. Lange, and C. von der Malsburg, "Distortion invariant object recognition by matching hierarchically labelled graphs," in *Proc. Int. Joint Conf. on Neural Network*s, vol. 1, (Washington D.C.), pp. 155--159, July 1989.

[14] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-12, pp. 55--73, January 1990.

[15] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture in two nonstriate visual areas(18 and 19) of the cat," *Journal of Neurophysiology*, vol. 28, pp. 229--289, March 1965.

[16] H. Asada and M. Brady, "The curvature primal sketch," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 2--14, January 1986.

[17] R. von der Heydt and E. Peterhans, "Mechanisms of contour perception in monkey visual cortex. I. lines of pattern discontinuity," *Journal of Neuroscience*, vol. 9, pp. 1731--1748, May 1989.

[18] E. Peterhans and R. von der Heydt, "Mechanisms of contour perception in monkey visual cortex. II. contour bridging gaps," *Journal of Neuroscience*, vol. 9, pp. 1749--1763, May 1989.

[19] Q. Zheng and R. Chellappa, "A computational vision approach to image registration," *IEEE Trans. Image Processing*, Vol. 2, No. 3, pp. 311--326, July 1993.

[20] Q. Zheng and R. Chellappa, "Estimation of illuminant direction, albedo and shape from shading," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-13, pp. 680-702, July 1991.

[21] A. Samal and P. A. Iyengar, "Automatic recognition and analysis of human faces and facial expressions: A survey," *Pattern Recognition*, vol. 25, no. 1, pp. 65--77, 1992.

[22] V. Bruce and M. Burton, "Computer recognition of faces," in *Handbook of Research on Face Processing* (A. W. Young and H. D. Ellis, eds.), pp. 487--506, Elsevier Science Publishers B.V. (North Holland), 1989.

[23] L. D. Harmon, M. K. Khan, R. Lasch, and P. F. Ramig, "Machine identification of human faces," *Pattern Recognition*, vol. 13, no. 2, pp. 97--110, 1981.

[24] C. J. Wu and J. S. Huang, "Human face profile recognition by computer," *Pattern Recognition*, vol. 23, no. 3/4, pp. 255--259, 1990.

[25] R. J. Baron, "Mechanisms of human facial recognition," *Intl. Journal of Man-Machine Studies*, vol. 15, pp. 137--178, 1981.

[26] I. Aleksander, "Emergent intelligent properties of progressively structured pattern recognition nets," *Pattern Recognition Letters*, vol. 1, pp. 375--384, 1983.

[27] J. Stonham, "Practical face recognition and verification with wisard," in *Aspects of Face Processing* (F. N. H. Ellis, M. Jeeves and A. Young, eds.), Dordrecht: Martinus Nijhoff, 1986.

[28] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture." pre-print, 1991.

[29] T. Kanade, *Picture processing system by computer complex and recognition of human faces*. Ph.D. thesis, Kyoto University, Department of Information Science, November 1973.

[30] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (Maui, Hawaii), pp. 586--591, June 1991.

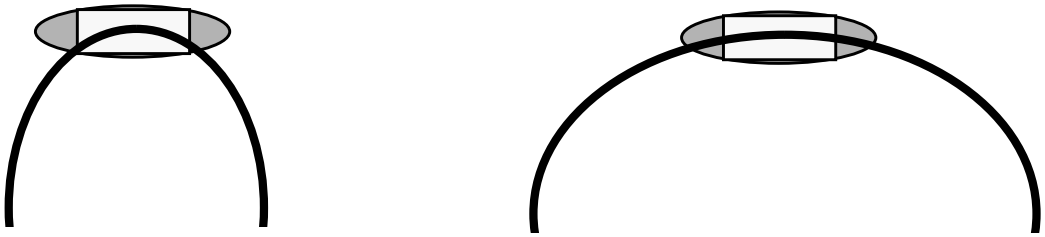
[31] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition CVPR '94, (Seattle, Washington), pp. 84-91, June 1994.

[32] T. Kohonen, *Self-Organization and Associative Memory.* New York: Springer-Verlag, 1989.

[33] A. Fuchs and H. Haken, "Pattern recognition and associative memory as dynamical processes in a synergetic system I," *Biological Cybernetics*, vol. 60, pp. 17--22, 1988.

[34] A. Fuchs and H. Haken, "Pattern recognition and associative memory as dynamical processes in a synergetic system II," *Biological Cybernetics*, vol. 60, pp. 107--109, 1988.

[35] S. Chandrashekhar and R. Chellappa, "Passive navigation in a partially known environment," in *IEEE Workshop on Visual Motion*, (Princeton, NJ), pp. 2--7, October 1991.



(a) (b)

FIGURE 1. Illustrating the selectivity of the feature detector (shown as an ellipse) to local curvature changes. In (a), the negative end zones (darker shaded regions at the two ends of the ellipse) are not activated, and the feature detector responds strongly to the local curvature, (b) the negative end zones suppress the detector's response. In our model, the negative end zones are a result of the difference of responses of filters with different scale parameters.

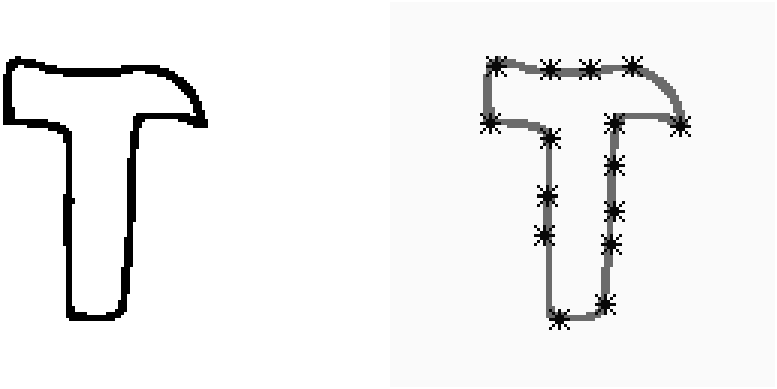
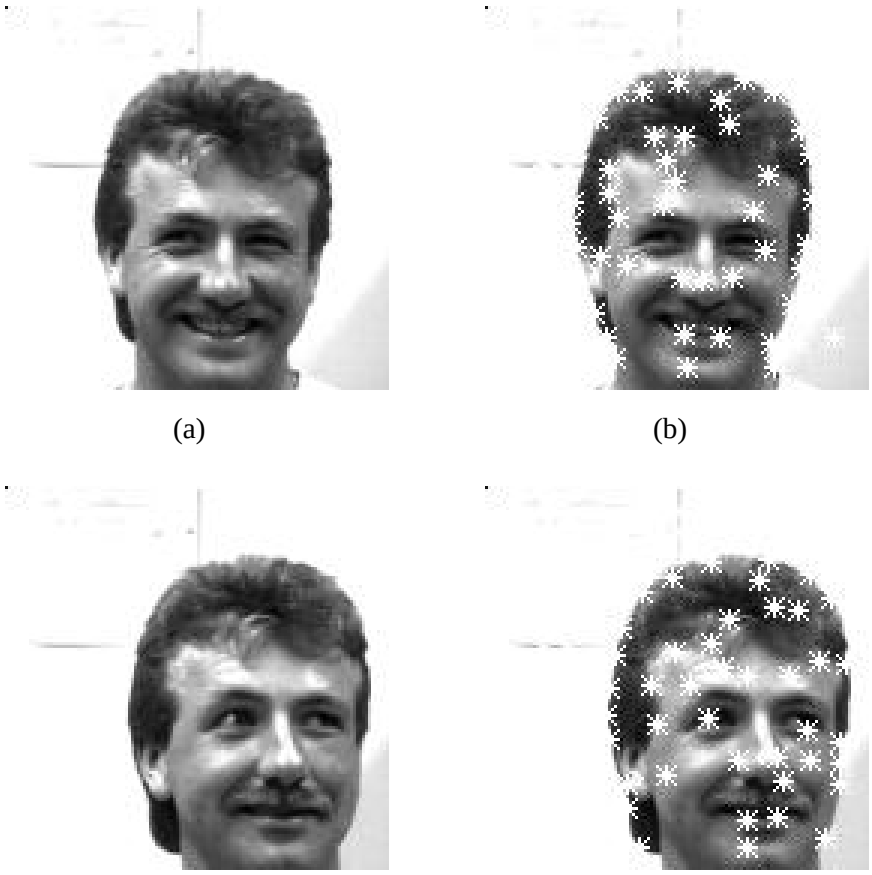


FIGURE 2. Salient feature locations identified by the model;. Parameter values used are i = 0, j = -6, and $\alpha = \sqrt{2}$. These correspond to 1 and 8 pixel standard deviation of the Gaussians, respectively.



(a) (b)

(c) (d)

FIGURE 3. Feature locations marked for face images. Scale parameters used in the experiment correspond to $i = -2, j = -5, \alpha = \sqrt{2}$. Information at these locations is used for recognition. The two faces shown here are matched from a database of over 300 images.

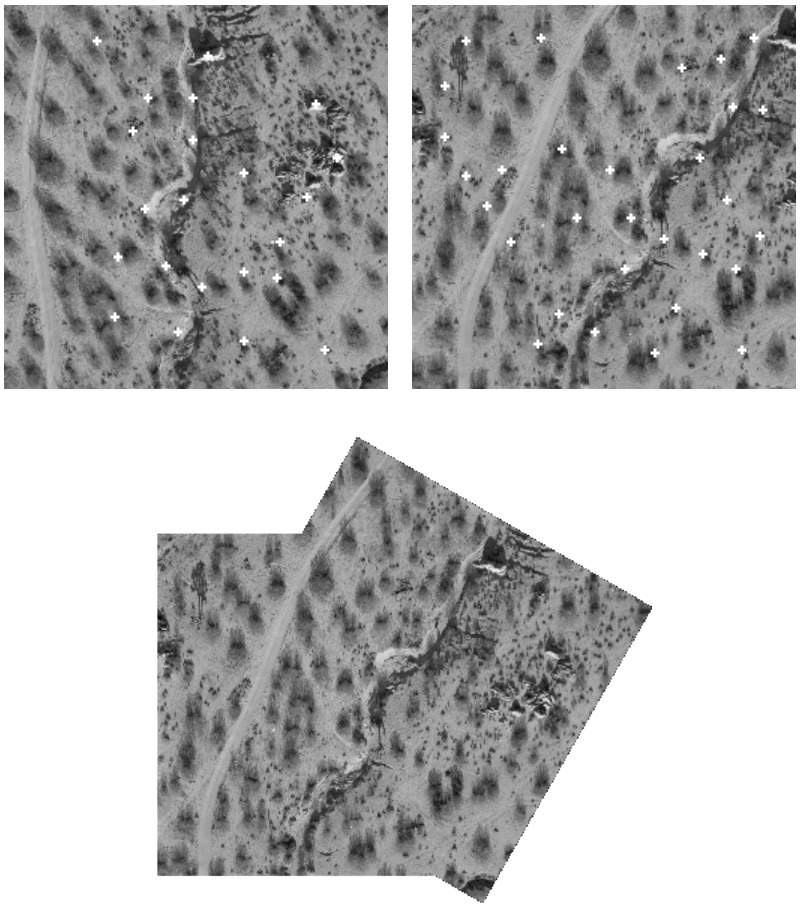


FIGURE 4. Two successive images from a motion sequence with the feature locations identified by the model. The composite image is obtained using a novel computer vision based registration technique by Zheng [IP Transaction'93]

FIGURE 5. Examples of successful matches. The left image of each pair is the input image and the right image is the best match found from a database of over 300 images. In 86% of the cases the best match was the correct match, and 94% of time the correct match can be found in the top three matches.

FIGURE 6. Examples of failures. The first image in each row is the input image, and the following three images are the top three matches found. Note that in the first two rows the correct match is among the best three matches.

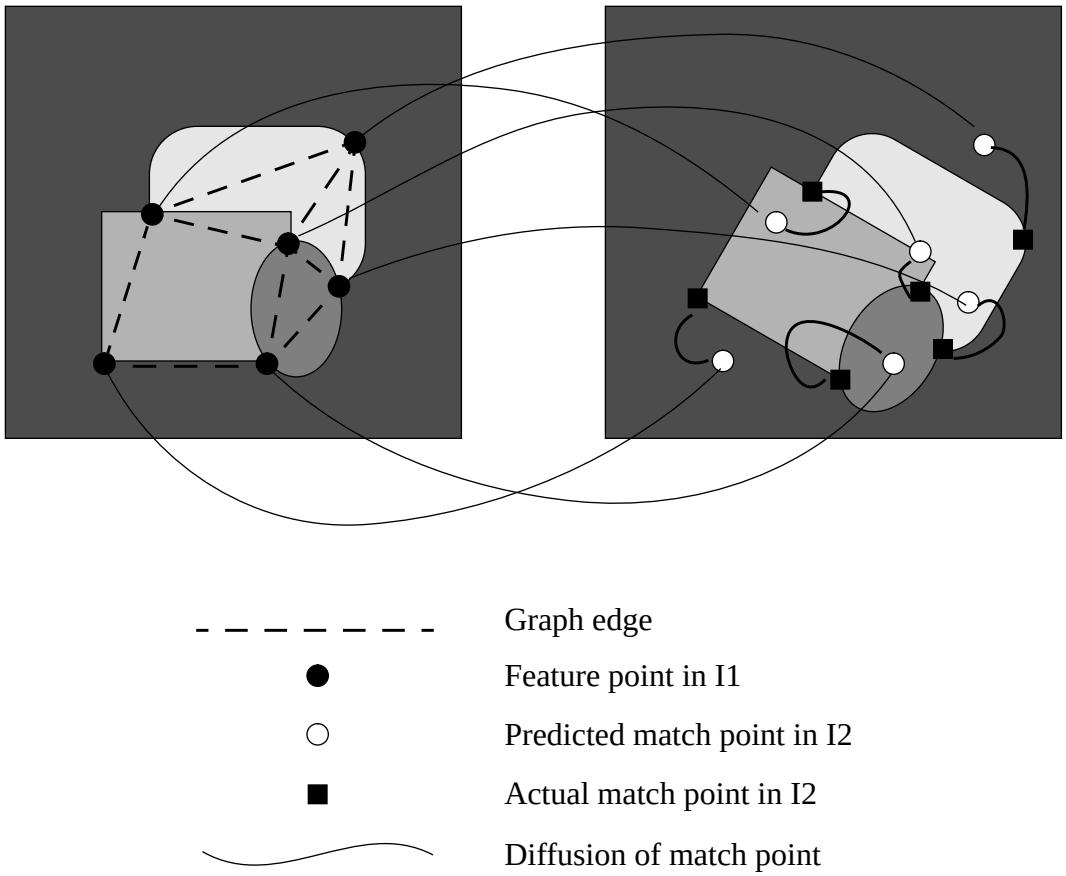


FIGURE 7. Labeled graph matching applied to motion correspondence.

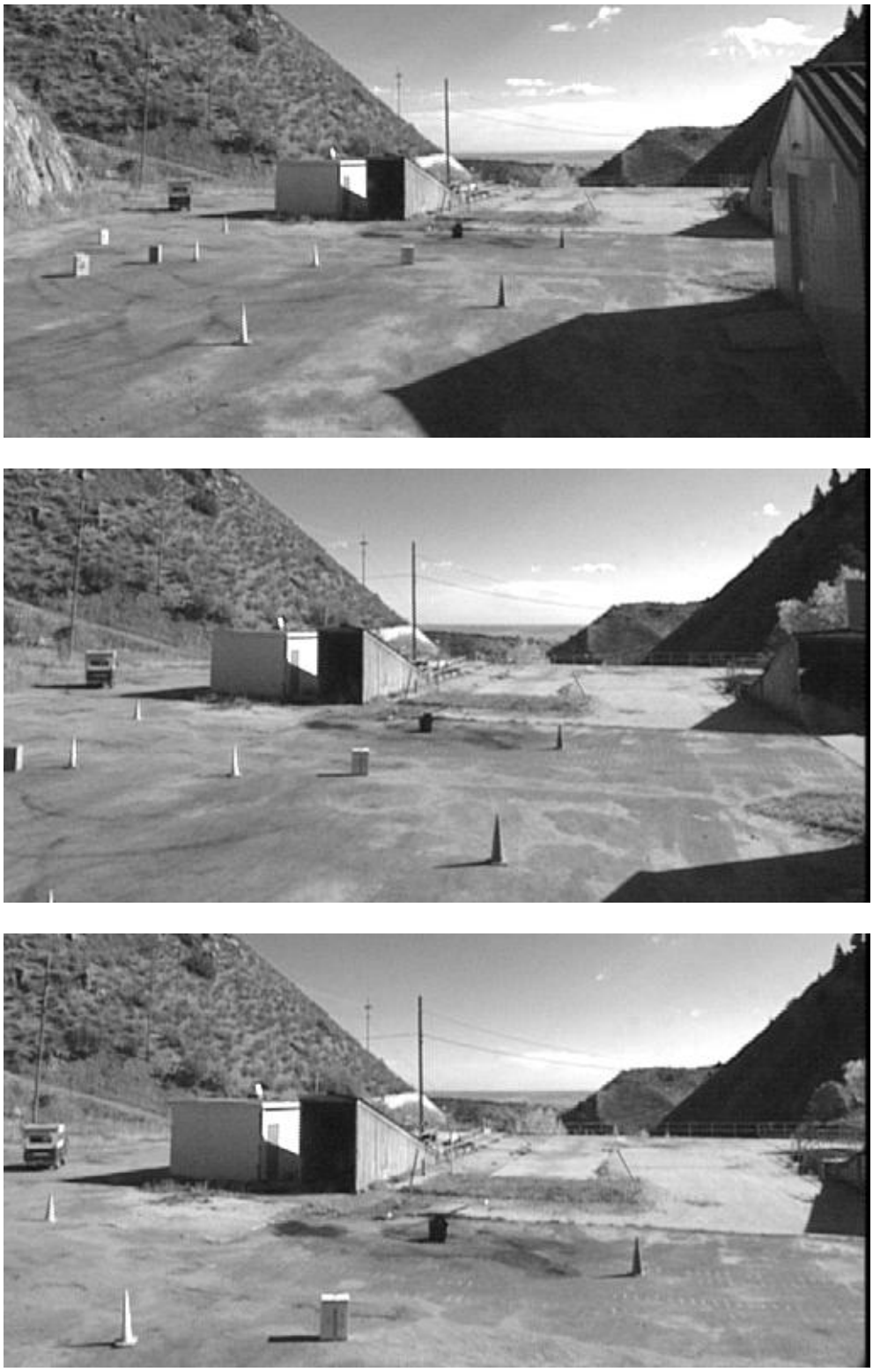


FIGURE 8. Frames 1, 8, and 16 of the Rocket sequence

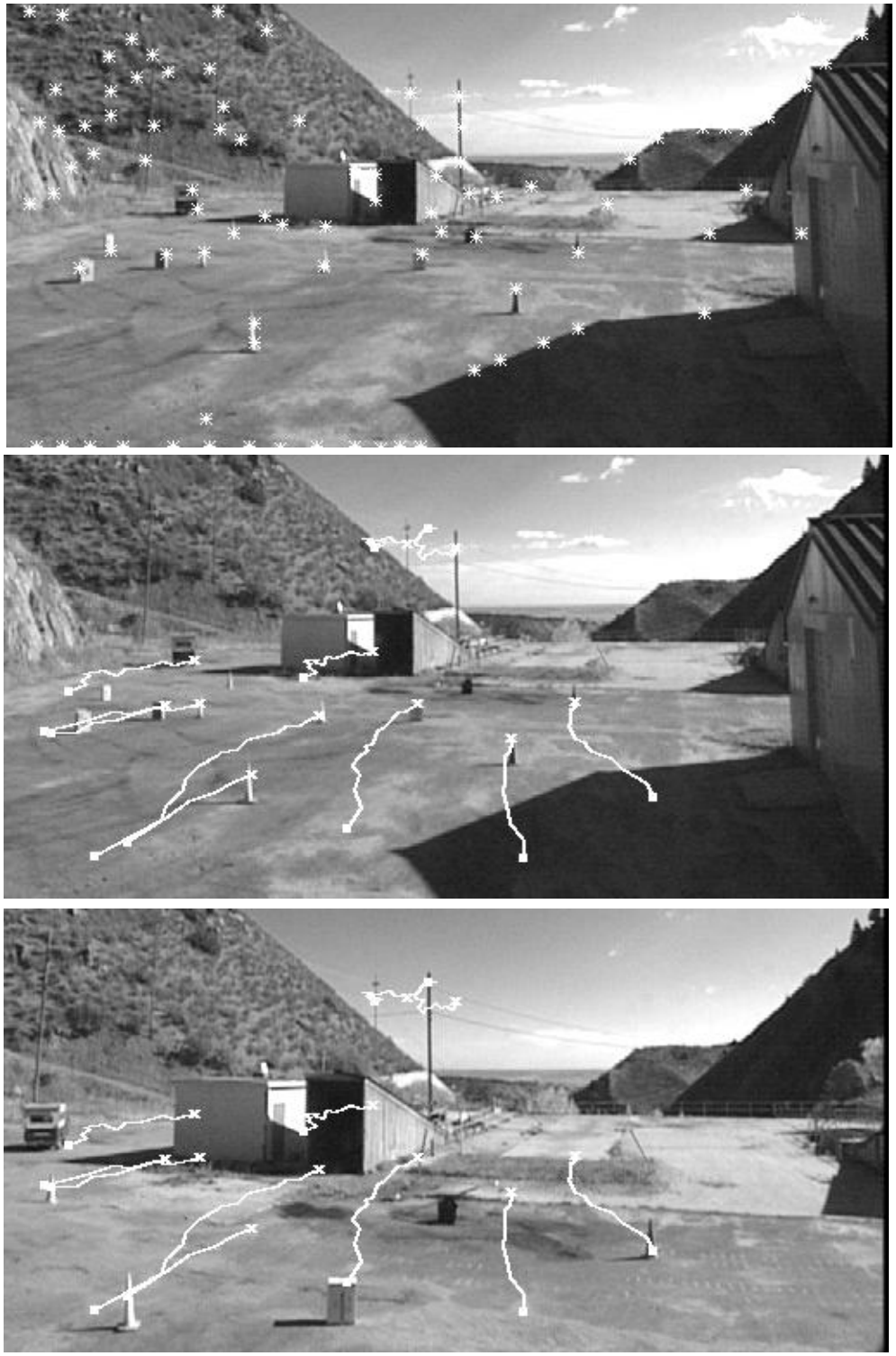


FIGURE 9. Feature points extracted from the first image of the Rocket sequence (top), and trajectories of selected points, superimposed on the first and last image in the sequence (middle and bottom).

Biography

B. S. Manjunath received his B.E. degree in Electronics (with distinction) from the Bangalore University in 1985, M.E. degree in Systems Science and Automation (with distinction) from the Indian Institute of Science in January 1987, and Ph.D. in Electrical Engineering from the University of Southern California in 1991. During September 1987 to July 1991 he was a research assistant at the Signal and Image Processing Institute, USC, and in the summer of 1990 he worked at the IBM T.J. Watson research center at Yorktown Heights, NY. He joined the Electrical and Computer Engineering department at the University of California, Santa Barbara as an assistant professor in July 1991.

Dr. Manjunath was awarded the University Gold Medal for the best graduating student in Electronics Engineering in 1985. He was also a recipient of the National merit scholarship from the Government of India during the period 1978-1985. His current research interests include computer vision, medical image analysis, pattern recognition and information retrieval in large image databases.

Chandra Shekhar is an Assistant Research Scientist at the Center for Automation Research at the University of Maryland, College park, where he has been working since Dec. 1994. He has a Bachelor's degree in electronics, a Master's degree in computer science, and a PhD in electrical engineering.

His PhD thesis (1987-'92, University of Southern California) was on visual motion analysis from image sequences. After his thesis, he worked for 18 months as a post-doctoral researcher at I.N.R.I.A. in Sophia-Antipolis, France. There his research focussed on automatic supervision of image-processing programs. He then worked as an engineer on the Prometheus traffic safety project for a year (1993-'94). During this tenure, he developed a module for the real-time supervision of the perception programs in an intelligent road vehicle, which was successfully demonstrated at the Intelligent Vehicles Conference in Oct. 1994. He is currently working on the integration of ATR algorithms.

Rama Chellappa is a Professor in the Department of Electrical Engineering at the University of Maryland at College Park, where heis also affiliated with the Center for Automation Research (as Associate Director). He was previously an Associate Professor and Director of the Signal and Image Processing Institute at the University of Southern California at Los Angeles. Over the last fourteen years, he has authored twenty book chapters and over hundred and fifty peer reviewed journal and conference papers. he has edited the Collected Papers on Digital Image Processing. He has co-authored a research monograph on *Artificial Neural Networks* f*or Computer Vision* and a co-edited the book M*arkov Random Fields: Theory and Applications*, published by Academic Press. He has served as an associate editor for several lead-

ing journals, including *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *IEEE Transactions on Neural Networks* and *IEEE Transactions on Image Processing*. He has received numerous prestigious awards including the 1985 National Science Foundation (NSF) Presidential Young Investigator Award, the 1985 IBM Faculty Development Award and the Best Paper Award (with Q. Zheng) at the 1992 International Conference on Pattern Recognition.He also received the Excellence in Teaching award from the School of Engineering at the University of Southern California. He was the General Chairman of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition and of the IEEE Computer Society Workshop on Artificial Intelligence for Computer Vision, both held in San Diego in June 1989. He is also the Technical Program Chair of The Second International Conf. on Image Processing. His current research interests are Computer Vision, Image Processing, Automatic Target Recognition and Neural Networks.