COMPUTER VISION AND IMAGE UNDERSTANDING
Vol. 71, No. 1, July, pp. 110–136, 1998
ARTICLE NO. IV970637

Quantitative Measures of Change Based on Feature Organization: Eigenvalues and Eigenvectors*

Sudeep Sarkar

Image Analysis Research Lab, Department of Computer Science and Engineering, University of South Florida, Tampa, Florida 33620
E-mail: sarkar@cs.csee.usf.edu

and

Kim L. Boyer

SAMP-Laboratory, Department of Electrical Engineering, The Ohio State University, Columbus, Ohio 43210
E-mail: kim@ee.eng.ohio-state.edu

Received April 4, 1996; revised April 29, 1997

One important task of site monitoring is change detection from aerial images. Change, in general, can be of various types. In this paper we address the problem of developmental change at a site. For instance, we would like to know about new construction at a previously undeveloped site and possibly monitor its progress. Model based approaches are not suited for this kind of change as it usually happens in unmodeled areas. Since it is difficult to infer construction activity by predicting and verifying specific local features, we rely on more global statistical indicators.

The thesis of this paper is that the change induced by human activity can be inferred from changes in the organization among the visual features. Not only will the attributes of the individual image features change but also the relationships among these features will evolve. With the progress of construction we expect to see increased structure among the image features. We exploit this emerging structure, or organization, to infer change. In this paper, we propose four measures to quantify the global statistical properties of the individual features and the relationships among them. We base these measures on the theory of graph spectra. We provide extensive analysis of the robustness of these measures under various imaging conditions and demonstrate the ability of these organization-based measures to detect coarsely incremental developmental changes.

CONTENTS

* This research is supported in part by a RADIUS Phase-II seed grant from Lockheed-Martin and National Science Foundation CAREER Grant IRI-9501932.

I. INTRODUCTION

Manmade objects are organized; even their placement tends to be regular. For example, buildings exhibit geometric forms, road widths vary slowly, airplanes are parked parallel to one another, and a full parking lot has a regular structure. This 3D world structure manifests as regular 2D image feature organizations. We explore the role of this emergent structure or organization among the 2D image features in *change detection*. Specifically, we address the problem of detecting new construction at a previously undeveloped site and possibly monitoring its progress. This kind of change, by its very nature, is usually not predictable; it may occur in unmodeled areas such as forests. Thus, traditional approaches based on prior models do not work well. We suggest global statistical approaches as opposed to those which rely on model based local prediction and verification.

We detect changes associated with construction activity by monitoring the statistics of significant regular groupings of image features. Change induced by human activity can be inferred from change in the organization among the visual features. For example, as vehicles move in and out of parking facilities, the regularity of the visual structure waxes and wanes. New construction of buildings, roads, or airfields is also accompanied by the emergence of regular groupings. Bomb damage will associate with the disappearance or rapid change of organization. Thus, the appearance and disappearance of patterns or organizations offer clues to detecting change and can guide more

1077-3142/98 $25.00

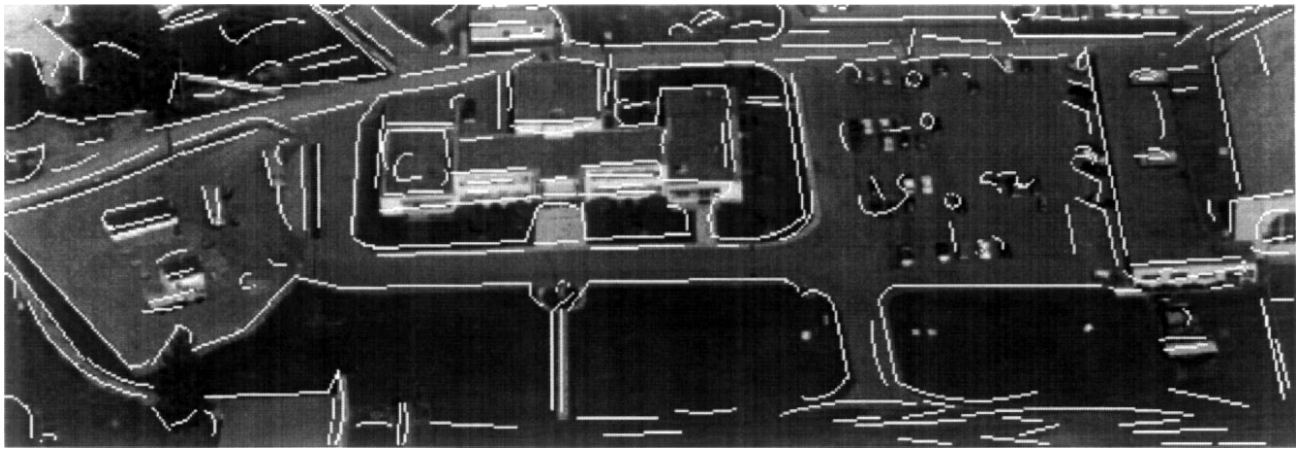

FIG. 1. A *fully* constructed building site overlaid with the constant curvature edge segments.

specialized image understanding algorithms or signal image analysts to quantify and characterize that change. We propose statistical measures to capture both the individual geometric attributes of the image features and the regularity of the relationships among these features. Changes in the proposed measures provide evidence for construction activity.

Figures 1 and 2 show a fully developed site and an undeveloped forest site, respectively. The gray level images are overlaid with constant curvature edge segments. Note the difference in the organization among the edge features. The edges in the fully developed site exhibit more parallelism, continuity, closure, and perpendicularity than those in the undeveloped site. Also, the edge segments in the fully developed site are generally larger than those in the undeveloped site. We quantify these qualitative differences using measures which capture both the statistics of the individual edge features and the relationships among them. The underlying theory is based on the spectra (eigenvalues and eigenvectors) of graphs. The relationships among the image edge features are represented as a *relation graph*. The eigenvalues and the eigenvectors of the adjacency matrix of this graph provide us with measures which capture the global relationship among the edge features.

The use of eigenvalues and eigenvectors in computer vision is not new. The concept of eigenvalue has been used for edge detection [1], nonrigid motion estimation [2], face recognition [3], and region segmentation [4]. However, the usage has been mostly at the signal level, except for Shapiro and Brady [5] and Sengupta and Boyer [6, 7]. Shapiro and Brady used the eigenvectors of a

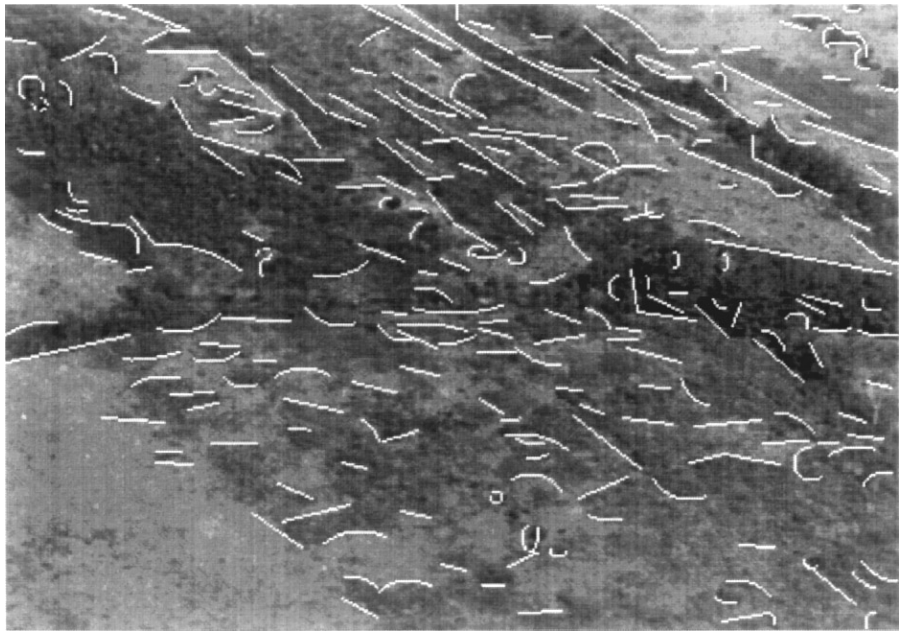

FIG. 2. An undeveloped forest site overlaid with the constant curvature edge segments.

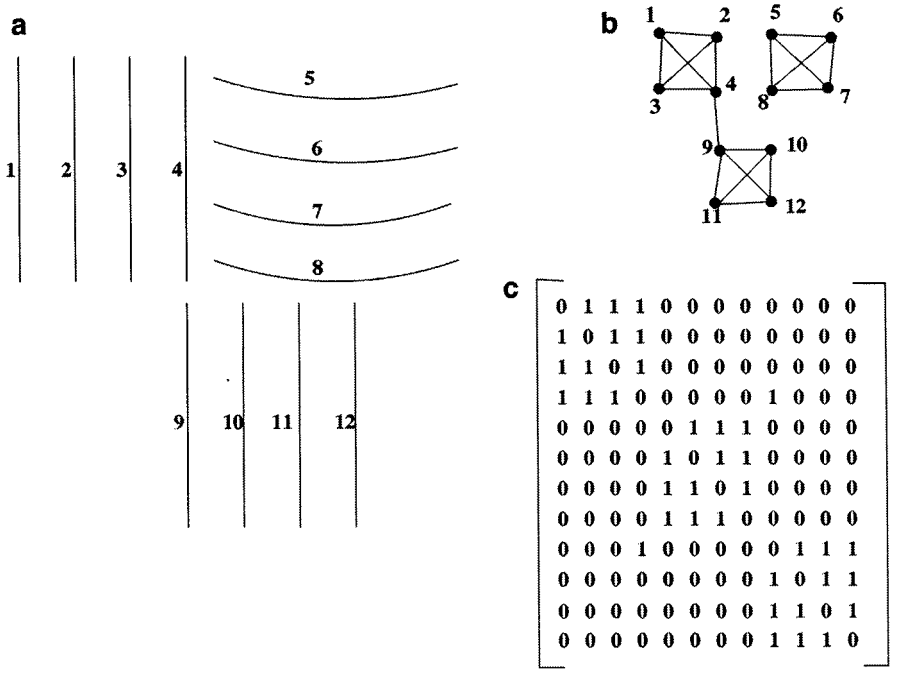

FIG. 3. (a) An example arrangement of edge features. (b) The relation graph with nodes corresponding to the individual edge segments and the links denoting pairwise compatibility. (c) The adjacency matrix corresponding to the relation graph.

proximity graph to establish correspondence between two sets of features. Sengupta and Boyer used the eigenvalues of a connectivity relation matrix of a 3D model to extract global attributes of an object. In this paper, we generalize the previous use of eigenvalues and apply it for a very different purpose. We also provide an extensive study of the stability and the robustness of the eigenvalues.

Previous work in the area of change detection includes the contributions by Huang, Mundy, and Rothwell [8] who attacked the problem of focused change detection in the context of a site model. The three types of change they monitored are parking lot occupancy, counting of buildings in a region given their models, and detecting door states of a building (again) given the site model. Benjamin, Huertas, Medioni, and Nevatia [9] also tackled the problem of model-based change detection. The algorithm has three parts. First, they registered a given image with the site model using shadows. Second, the model features were matched to image features. And third, the model objects were validated. Among other change detection algorithms are pixel based differencing algorithms [10, 11] which obviously suffer from the problem of being unable to differentiate between imaging condition changes and those due to important structural changes and which require precise registration or common camera position and orientation.

The organization of this paper is as follows. In Section II, we discuss the theory of graph spectra. The construction of the *relation graph* is discussed in Section III. Section IV presents empirical results on the stability of the eigenvalues with respect to imaging condition changes. Section V introduces the concept of *eigenclusters*. We propose the measure for change detection in Section VI. Section VII describes the use of the eigenvalues in change detection. We conclude with Section VIII.

II. THEORY: GRAPH SPECTRA

The global relationship among image features can be very effectively captured in the form of a graph whose nodes represent the image features and whose links denote compatibility between the features. Two image features are said to be compatible if they exhibit pairwise organization, e.g., the two structures are of the same type, similar size, and have similar orientation (generalized parallelism). We call this graph the *relation graph*.

Consider Fig. 3a which shows an example arrangement of constant curvature edge segments. The associated relation graph along with its adjacency matrix are shown in Figs. 3b and 3c, respectively. The graph nodes represent the constant curvature segments and the links denote compatibility. We assume that compatibility is a crisp relationship only for this example. In practice, the links in the graph are weighted according to the degree of compatibility between two nodes. Our task is to formulate measures which capture global properties of this relation graph. Eigenvalues and eigenvectors of the relation graph provide exciting possibilities as a basis for such measures. The concept of graph eigenvalues is motivated as follows [12].

A. Eigenvalues of Graphs

Let us consider the task of finding node clusters in a weighted graph $\mathcal{G}$ of n nodes. Given a node cluster, we can motivate a cluster cohesiveness measure in the following way.

TABLE 1
Eigenvalues and the Corresponding Eigenvectors for the Edge Segments Shown in Fig. 3

		Eigenvectors (nodes)											
Eigenvalues		1	2	3	4	5	6	7	8	9	10	11	12
−1.7	$\mathbf{x_1}$	−0.17	−0.17	−0.17	0.64	0.0	0.0	0.0	0.0	−0.64	0.17	0.17	0.17
−1	$\mathbf{x_2}$	0.03	0.03	−0.05	0.0	−0.80	0.46	0.37	−0.04	0.0	0.08	−0.08	0.01
−1	$\mathbf{x_3}$	−0.01	−0.01	0.03	0.0	0.25	−0.16	0.61	−0.69	0.0	0.13	−0.21	0.07
−1	$\mathbf{x_4}$	0.03	0.03	−0.07	0.0	−0.04	−0.15	−0.08	0.27	0.0	0.17	−0.73	0.57
−1	$\mathbf{x_5}$	−0.04	−0.04	0.07	0.0	0.07	0.34	−0.14	−0.28	0.0	−0.67	0.11	0.56
−1	$\mathbf{x_6}$	0.02	0.02	−0.03	0.0	−0.21	−0.61	0.47	0.34	0.0	−0.41	0.25	0.16
−1	$\mathbf{x_7}$	0.79	−0.57	−0.22	0.0	0.02	0.01	−0.01	−0.02	0.0	−0.02	0.02	0.0
−1	$\mathbf{x_8}$	−0.20	−0.58	0.78	0.0	−0.07	−0.03	0.03	0.08	0.0	0.06	−0.06	0.0
−0.3	$\mathbf{x_9}$	0.25	0.25	0.25	−0.57	0.0	0.0	0.0	0.0	−0.57	0.25	0.25	0.25
2.8	$\mathbf{x_{10}}$	0.37	0.37	0.37	0.29	0.0	0.0	0.0	0.0	−0.29	−0.37	−0.37	−0.37
3.0	$\mathbf{x_{11}}$	0.0	0.0	0.0	0.0	−0.50	−0.50	−0.50	−0.50	0.0	0.0	0.0	0.0
3.3	$\mathbf{x_{12}}$	0.33	0.33	0.33	0.43	0.0	0.0	0.0	0.0	0.43	0.33	0.33	0.33

Let us represent a node cluster using a column vector, $\mathbf{x}$, whose kth entry captures the participation of node k in that cluster. If a node does not participate in a cluster, the corresponding entry is zero. We allow for a graded membership of a node in a cluster. We also impose the restriction that the norm of this weight vector $\mathbf{x}$ be one or $\mathbf{x}^T\mathbf{x} = 1$. Then, based on the link weights of the graph, w_{ij}, we can define the measure for the cohesiveness of the node cluster as

$$\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}x_i x_j = \mathbf{x}^T\mathbf{A}\mathbf{x}, \tag{1}$$

where $\mathbf{A}$ is the weighted adjacency matrix. Note that since the entries in $\mathbf{x}$ corresponding to the nonparticipating nodes in a cluster are zero, only a submatrix of $\mathbf{A}$ will be essentially involved in the above equation. The more well connected the cluster nodes are, the larger is the cluster cohesiveness.

A maximally cohesive cluster $\mathbf{x}$ can be found by maximizing the above expression. The Rayleigh–Ritz theorem [13, p. 176] states that the maximum value of the above expression will be $\lambda_{\max}$, the maximum eigenvalue[1] of $\mathbf{A}$, and the corresponding eigenvector will be the optimal $\mathbf{x}$. In fact, the theorem also implies that the minimum value will be $\lambda_{\min}$, the minimum eigenvalue of $\mathbf{A}$.

In general, we will have n eigenvalues, which is the total number of nodes in the graph. The (nondecreasing) ordered set of eigenvalues $\{\lambda_1, \ldots, \lambda_n\}$ is referred to as the *spectrum* of the graph. An interpretation for all the eigenvalues and eigenvectors of $\mathbf{A}$ can be given using the Courant–Fischer theorem [13, p. 179], which is a generalization of the Rayleigh–Ritz theorem. Let $\{\mathbf{x}_{\lambda_n}, \ldots, \mathbf{x}_{\lambda_{n-k+1}}\}$ denote the set of eigenvectors corresponding to the $(k-1)$ largest eigenvalues, $(\lambda_n, \ldots, \lambda_{n-k+1})$, respectively. Then,

$$\lambda_{n-k} = \max_{\mathbf{x}\perp\{\mathbf{x}_{\lambda_n},\ldots,\mathbf{x}_{\lambda_{n-k+1}}\}} \mathbf{x}^T\mathbf{A}\mathbf{x}. \tag{2}$$

Thus, the kth largest eigenvalue, λ_{n-k}, is the maximum value of the cluster cohesiveness measure for all node weight assignment vectors $\mathbf{x}$ that are orthogonal to the eigenvectors with the $(k-1)$ largest eigenvalues. For example, the components of the eigenvector with the second largest eigenvalue give a cluster weight assignment which is *orthogonal* to the cluster weight assignment (eigenvector) with the largest eigenvalue. In practical terms, we can use this orthogonality constraint to define disjoint clusters.

Thus, we see that the spectrum of a graph provides us with a natural clustering of its nodes. The components of an eigenvector denote the node participations in the associated cluster, while the corresponding eigenvalue denotes the coherency of the cluster. If the nodes participating in a cluster are all connected to each other with weight one (a perfectly coherent group), then its eigenvalue is the maximum possible, the number of nodes in the group minus one. The eigenvalue decreases as the interconnections get sparser. Thus, the larger the eigenvalue, the greater the coherence and the size of its associated cluster.

In general, the components of an eigenvector will not be all positive. Eigenvectors with negative components are not meaningful in the above physical interpretation of a graph spectrum since the membership of a node in a cluster cannot be negatively valued. Thus, in this paper, we consider only positive eigenvectors. An eigenvector $\mathbf{x_i}$ is said to be positive if *all* the components of $\mathbf{x_i}$ or $-\mathbf{x_i}$ are positive and its corresponding eigenvalue is positive. (Note that if $\mathbf{x}$ is an eigenvector then so is $-\mathbf{x}$.) We call the collection of nodes corresponding to the nonzero components of a positive eigenvector an *eigencluster*. For the above example, the graph in Fig. 3b has the eigenvalues and the eigenvectors of its adjacency matrix shown in Table 1. We have two positive eigenvectors, $\mathbf{x_{11}}$ and $\mathbf{x_{12}}$. The eigenclusters

[1] For any $N \times N$ matrix $\mathbf{A}$, eigenvalues are constants (λ) such that $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ for a column vector $\mathbf{x}$, the corresponding eigenvector.

are (1, 2, 3, 4, 9, 10, 11, 12), and (5, 6, 7, 8). Note that nodes 4 and 9 are "stronger" members of the first cluster since they have the largest node valency.

The eigenclusters will in most cases be disjoint. Recall that the clusters are derived from the nonzero components of the positive eigenvectors and from matrix theory we know that the eigenvectors of a Hermitian matrix ($\mathbf{A} = \bar{\mathbf{A}}^{\mathbf{T}}$) with all distinct eigenvalues are orthogonal to each other. As formulated in this paper, the underlying relation matrix will be Hermitian and we have found that in practice the eigenvalues are distinct. However, as we will see in Section V, in practice we relax the definition of a "positive" eigenvector to allow for clusters with small (up to 5%) overlaps.

The graph spectrum also captures other important structural properties of the graph. We summarize a few of those properties here. The average node valency of an unweighted graph is less than the maximum eigenvalue, $\lambda_{\max}$. For the example in Fig. 3, the average valency is 3.15 and $\lambda_{\max}$ is 3.3. A complete unweighted graph of n nodes has $n-1$ eigenvalues equal to -1 and *one* eigenvalue equal to $n-1$. Two isomorphic graphs have the same spectrum. However, two graphs with the same spectrum need not be isomorphic. There are other graph properties which can be computed from the graph spectrum such as its interior stability, the chromatic number, the number of complete subgraphs, and bipartiteness. The reader is referred to [12] for more details.

B. Variation of Eigenvalues with Graph Structure

For the spectrum to form the basis for stable measures we need to study its sensitivity to two basic forms of change, namely, the addition or deletion of links and the increase or decrease of the link weights. We start with a randomly generated graph of 100 nodes with a sparseness index of 0.025, a typical value for relation graphs from real images according to our method of building them. We define the sparseness as the ratio of the number of links to the total *possible* number of links. We conducted two studies. In the first study we randomly perturbed the links and the weights of the graph. In the second study we systematically changed the adjacency graph.

1. Random perturbations. Since the eigenvalues are continuous functions of the graph link weights, we have reason to believe that if these weights are perturbed by small values, then the eigenvalues should not change too drastically. In fact, Weyl's theorem can be used to provide us with a bound on the variation of the eigenvalues [13, p. 367]. Let $\mathbf{A}$ and the perturbation matrix, $\mathbf{E}$, be two real symmetric matrices with $\lambda_1 \leq \lambda_2 \leq \cdots \lambda_n$ and $\lambda_1(E) \leq \lambda_2(E) \leq \cdots \lambda_n(E)$ as their respective ordered spectra. Let $\hat{\lambda}_1 \leq \hat{\lambda}_2 \leq \cdots \hat{\lambda}_n$ be the ordered eigenvalues of $\mathbf{A}+\mathbf{E}$. Then it can be shown that

$$\lambda_1(E) \leq \hat{\lambda}_k - \lambda_k \leq \lambda_n(E) \quad \text{for all } k = 1, 2, \ldots, n. \tag{3}$$

Thus, the variation of the eigenvalues of a perturbed matrix is bounded by the maximum and the minimum eigenvalues of the perturbing matrix. We can use the Hoffman and Wielandt theorem [13, p. 368] to derive the following more comprehensive bound on the perturbations to all the eigenvalues.

$$\left(\sum_k |\hat{\lambda}_k - \lambda_k|^2 \right)^{1/2} \leq \|E\|_2, \tag{4}$$

where $\|E\|_2$ is the Euclidean norm (square root of the sum of the squared entries of E) of the perturbing matrix E.

We verified these theoretical results by considering two types of perturbations, namely, random variations in the graph links and their weights. The random variation of link weights is simulated by considering weights which are uniformly distributed between 0 and 1. We started from a random unweighted graph of sparseness 0.025 and generated 100 weighted (mutually independent) graphs from it by replacing the nonzero links with uniformly distributed weights, over 0 to 1. From Fig. 4a we see that the variation of the eigenvalues is low and tightly bounded. The solid plots in Fig. 4 refer to the mean spectra and the circles mark the minimum and maximum variations in the eigenvalues. The horizontal axes denote the order of the eigenvalues in the graph spectra: the smallest eigenvalue is represented by 1 and the largest eigenvalue is represented by 100. Note that this does not mean that there is little information about the weighting structure in the graph spectrum, but rather that graphs with similar weight distribution will tend to have similar spectra. However, differences in the nature of the distribution of the link weights will tend to be reflected in the eigenvalues, as we shall see in the next section.

To model the random addition and deletion of graph links we randomly generated graph adjacency matrices whose nonzero entries occur with a probability of 0.025, the sparsness index. We constructed 100 such matrices and Fig. 4b shows the variation of the eigenvalues. Note the tight bound.

2. Statistical parameter changes. In practice we would like the eigenvalues to change systematically if the graph links and their weights show systematic change. With progress in construction activity we expect not only the emergence of new relationships between image features but we also expect the relationships to grow stronger. In this section, we study the effect of these two types of change on the eigenvalues. First, we consider an *unweighted* relation graph and study the effect of adding new links. Second, we consider a *weighted* relation graph with a static link structure, but change the link weights. Third, we study the effect of varying both the links and their weights.

An independent, identical, and uniformly distributed random variable is a poor model of the new links added during construction. The emergence of new links is dependent on the existing links. Although the exact statistics depend on the particular type of construction, we can, in general, observe that new construction follows a "crystal growing" pattern. For example, a hangar is not constructed by randomly building parts of it, but by adding new structures around a core. First the access roads are built, the

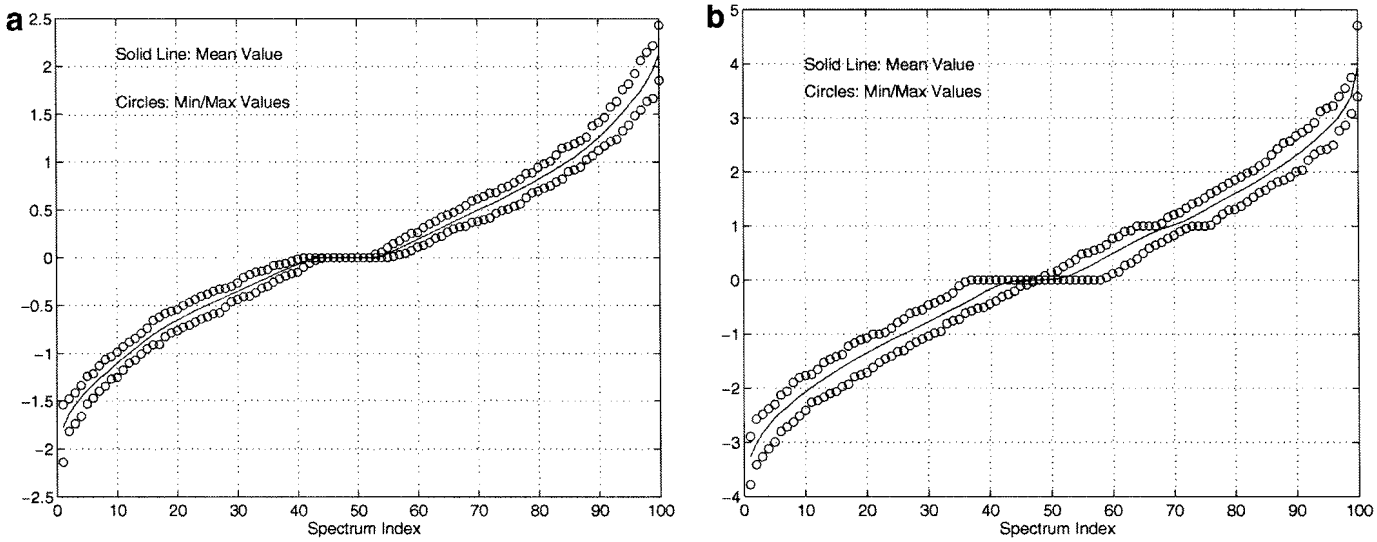

FIG. 4. Variation of eigenvalues with random variation in (a) the link weights and (b) the links. The horizontal axes denote the order of the eigenvalues in the graph spectra: the smallest eigenvalue is represented by 1 and the largest eigenvalue is represented by 100. The solid plot depicts the mean spectrum and the circles denote the maximum and the minimum variation under random perturbations.

foundations are laid, then the walls come up, next the roofs are built, followed by construction of parking lots around the hangar. This sequence of concentrated activities manifests itself as new image features emerging around existing ones, and in organized relationships with them. We model this kind of change by varying the probability of addition of a new link according to the local graph structure.

New links are added to the relation graph according to

$$P(w_{ij} = 1) = \left(\frac{\sum_i w_{ij} + \sum_j w_{ij}}{2n} \right)^2, \qquad (5)$$

where n is the total number of nodes and w_{ij} is the link between nodes i and j. (We assume only unweighted graphs in this simulation, thus, $w_{ij} = 0$ or 1.) The probability of adding a new link between two nodes increases with the valency of the nodes. The square in the expression models the accelerated nature of development. The rate of growth of emergent structures during the development process is not a constant but it increases with time. We start with a random unweighted graph of 100 nodes with a sparsness index of 0.015. We evolve the graph for 50 generations. At each generation we process the whole graph and add links according to the above probabilities. With time, the graph will exhibit increased structure as is reflected in the adjacency matrix entries shown in Fig. 5. Figure 6a shows the change in the ten largest eigenvalues with time. Note that the increase in graph structure is reflected in the increase of the eigenvalues.

Next, we vary the link weights. With construction, we expect the average link weight to increase, signifying an increased image structure. In a later section we explain how we capture the image structure and organization using these weights. For the present experiment, we model the weight increase using a skewed unimodal probability density whose mode can be varied, such as the beta density function. The choice of a beta density is only for experimental purposes. We consider link weights, w_{ij}, distributed according to the following density function:

$$P(w_{ij} = w) = \frac{(n+1)!}{k!(n-k)!} w^k (1-w)^{n-k} \quad 0 \le w \le 1. \qquad (6)$$

The parameters of the distribution are k and n. (This n is not the number of nodes in the graph.) Typical plots of the beta pdf are shown in Fig. 6b. Note the dependence of the mode of the distribution with k; the mode is at k/n. We chose $n = 10$ for our experiments and changed k from 1 through 9 to generate a set of randomly weighted graphs. The graphs with a high value for k will have high average link weight simulating progress in construction. Figure 6c shows the monotonic increase of the ten largest eigenvalues with k. Thus, the eigenvalues do reflect systematic change in the link weights, as well.

In the third study, we combined both kinds of effects; we added new links as well as varied their weights. Thus, with time new links appear, and their weights increase, modeling the emergence of structure in the underlying image. The value of the parameter k in the beta distribution was increased by one every sixth generation to model the increase in the link weights. The variation of the ten largest eigenvalues is shown in Fig. 6d. As expected, we observe a systematic change in the eigenvalues.

In summary, we find the eigenvalues to be stable with respect to modest unstructured perturbations of the graph structure, but responsive to systematic changes in the statistics underlying the graph.

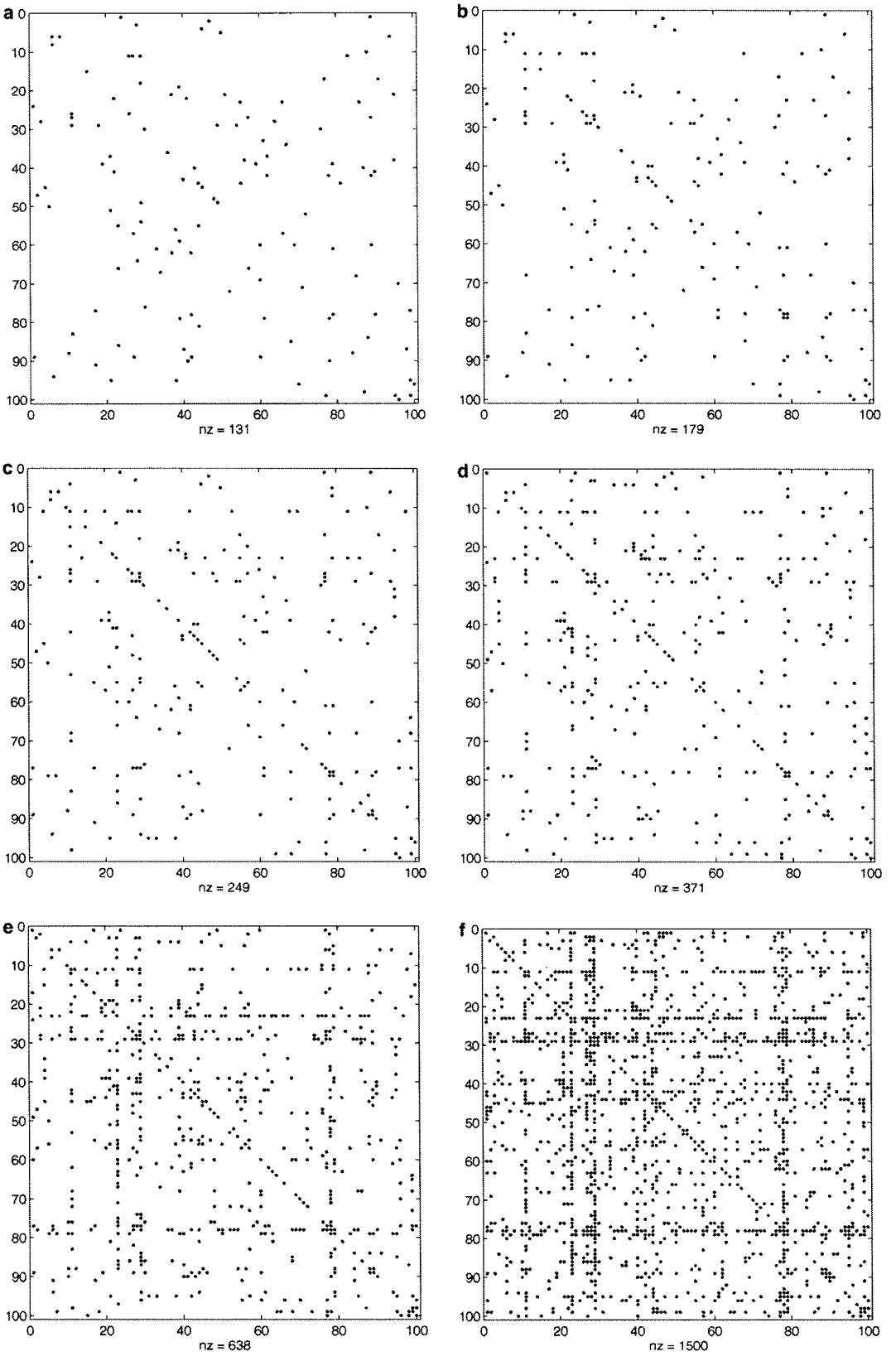

FIG. 5. The adjacency matrices of a graph evolved according to the scheme discussed in the text. (a) The starting adjacency matrix. The adjacency matrices after 10, 20, 30, 40, and 50 generations are shown in (b), (c), (d), (e), and (f), respectively.

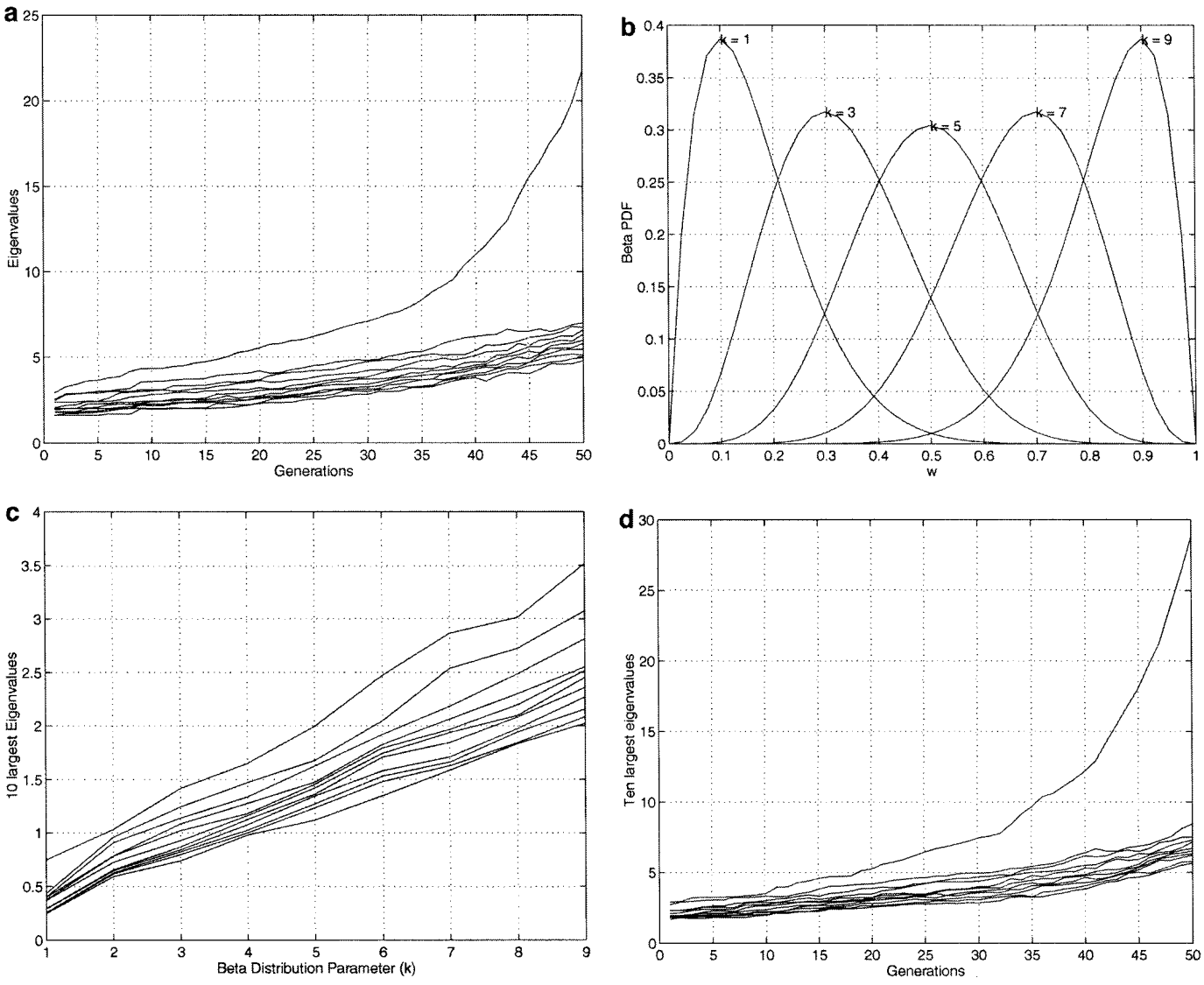

FIG. 6. Variation of the ten largest eigenvalues with systematic variation in (a) the links, (c) their weights, and (d) both the links and their weights. For (a) the links were randomly added with probabilities dependent on the local graph structure. The x-axis in (a) represents the progress of time. For (c) the weights were varied according to a beta probability distribution shown in (b). The x-axis in (c) refers to the parameter k of the beta distribution (see text). For (d) we both added new links and changed their weights according to a beta distribution.

III. THE RELATION GRAPH

We now consider the construction of a graph that captures the underlying image structure and organization. The final objective is to use the eigenvalues of this graph to quantify the image structure. For this purpose, we define a *relation graph* whose nodes denote primitive image features and whose links denote relationships between the image features. In this paper we consider constant curvature edge segments as the primitive image features. The weighted links between the features capture the regular Gestalt-inspired relationships of proximity, parallelism, closure, strands, perpendicularity, and continuity. A closure relationship (N-ary) exists among a set of edge segments if they enclose a region. A strand relationship (N-ary) exists among a set of edge segments if they form a contiguous chain of segments without completely enclosing a region, e.g., any three sides of a rectangle.

A. Gestalt Relations: Computation

We use the bottom-up algorithm described in [14] to find the Gestalt relationships and quantify their significance as described later. The algorithm uses graph operations and voting methods. Regularities among the edge segments based on the Gestaltic principles of proximity, similarity, smooth continuity, and closure are detected by the voting methods, which effect a search procedure over the image tokens and are shown to be superior

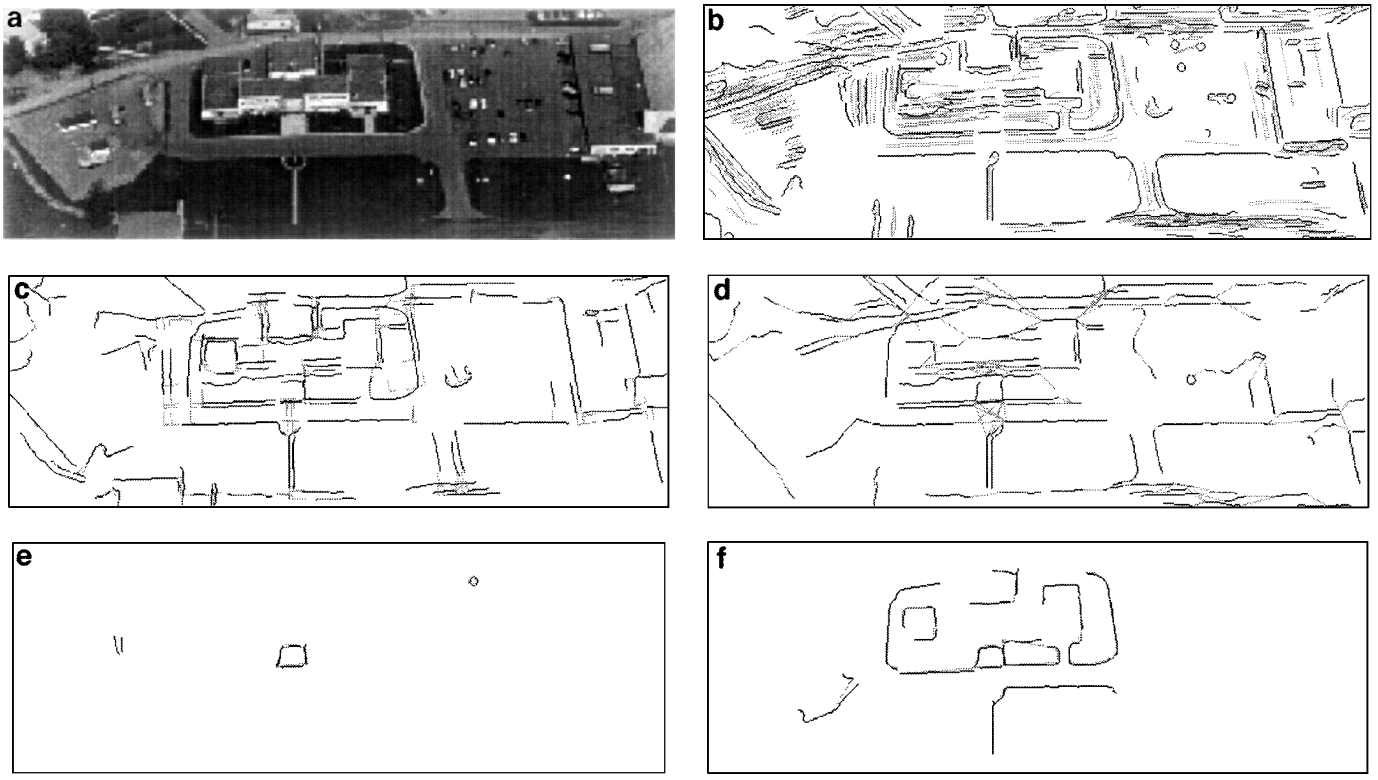

FIG. 7. A sample output of the bottom up organization module. (a) Gray level image. (b) The parallel symmetry axes are drawn in a shade lighter than the participating edge segments. (c) The perpendicular relations between edge segments are shown as small right angles in a shade lighter than the edge segments. (d) The continuous segments are joined with light shaded straight edges. (e) The edges forming a closed boundary. (f) The strands of edges.

to conventional techniques in an average execution time sense. The Gestaltic associations among tokens are represented by a set of Gestalt graphs, which let us apply sophisticated graph theoretic techniques. The outputs are elementary organizations such as parallel symmetries, strands, closures, perpendiculars, and continuous lines. Examples of these structures are shown in Fig. 7. The relationships are represented by a shade lighter than the edge segments. Thus, the parallel symmetry axes are drawn in a lighter shade than the participating edge segments. The perpendicular relationship is represented by a small right angle in light shade. Continuous segments are joined with light shaded straight edges.

B. Gestalt Relations: Quantification

We quantify the Gestalt relationships of proximity, parallelism, closure, strand, perpendicularity, and continuity using measures which incorporate the significance of the constituent edge segments. In the following discussion we denote the ith edge segment by e_i. The relations are illustrated in Fig. 8.

1. *Constant curvature edge segments:* The significance of an edge segment, e_i, is determined by its length, l_i, and the least root mean squared fit error, Err_i (in units of pixels), of a constant curvature segment to it. Mathematically, the significance is quantified by

$$\mathbf{Sig}(e_i) = \frac{l_i}{l_i + Err_i}. \tag{7}$$

The value is one if the fit error is zero and it decreases with increasing fit error. The significance increases with the segment length. See Fig. 8a.

2. *Proximity:* The significance of proximity between two edge segments is quantified using the ratio of the minimum distance ($d_{\min}$) between the end points of two segments to the length of the smaller segment (as shown in Fig. 8b). Note that the minimum distance between *all* points in the two segments is not considered because of increased complexity of computing such a measure. We have found the minimum distance between end points to be sufficient as a measure of proximity for our purposes.

$$\mathbf{Sig_{prox}}(e_i, e_j) = 1 - \min\left(1, \frac{d_{\min}}{\min(l_i, l_j)}\right), \tag{8}$$

where l_i and l_j are the lengths of the two edge segments. Note that the measure ranges from a maximum of 1 for two edge segments without a gap to 0 for two edge segments which are separated by a distance greater than the length of the smaller segment.

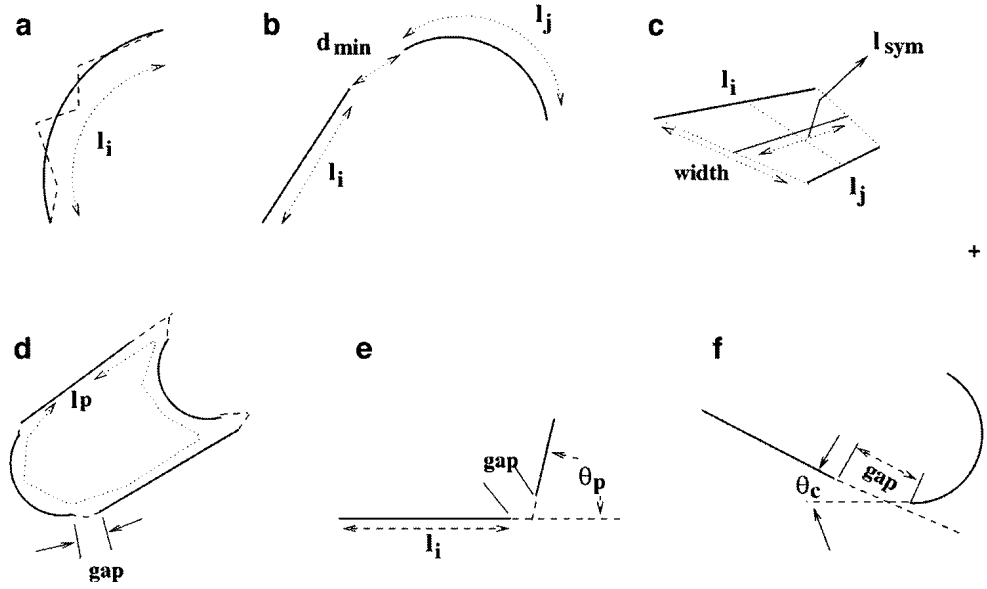

FIG. 8. The Gestalt-like relationships between two edge segments captured in the relation graph. (a) Individual edge segment, (b) proximity, (c) parallelism, (d) closure, (e) perpendicularity, (f) continuity. The figures illustrate the concepts in the corresponding expressions used to capture their significances (see text). The strand relation uses concepts similar to the closure relation.

3. *Parallelism:* The significance of a parallelism relation between two edge segments, e_i and e_j, is determined by the difference in length between the constituting edge segments, l_o, the length of the symmetry axis, l_{sym}, the average width, μ_w, and variation, σ_w, of the parallel strip, and the minimum fit error, Err_{sym}, of a constant curvature segment to the symmetry axis. The mathematical form is given by

$$\mathbf{Sig_{para}}(e_i, e_j) = \min\left(\mathbf{Sig}(e_i), \mathbf{Sig}(e_j), \frac{\mu_w}{\mu_w + \sigma_w}, \frac{l_{\text{sym}}}{l_{\text{sym}} + Err_{\text{sym}}}, \frac{l_{\text{sym}}}{l_{\text{sym}} + l_o}\right). \quad (9)$$

The symmetry axis is computed as described in [15]. One edge segment is warped onto the other with a constant warping factor proportional to the ratio of their lengths. This allows for skew symmetries. The midpoints of the lines joining the mapped points form the symmetry axis and the lengths of the joining lines are the local widths of the symmetry axis (as shown in Fig. 8c). The average value of the local widths is μ_w and their standard deviation is σ_w. The last term in the above expression captures the difference in length between the constituting edge segments; $l_o = |l_i - l_j|$.

Each of the terms on the right side of the equation capture an essential component of the parallelism definition. The overall significance is determined by the minimum value. Also, the above measure works for straight parallel symmetries as well as for curved parallel symmetries.

4. *Closure:* The significance of a closure relationship among a set of edge segments, $\{e_i, \ldots, e_k\}$, is determined by the significance of the constituent edge segments, the convexity of the enclosed region, and the amount of edge support we have for the perimeter of the enclosed region. The convexity of a region is quantified by the ratio of the actual area, a_{actual} to the area of its convex hull (smallest convex polygon circumscribing the region), a_{hull}. The ratio is one for a convex region and decreases with the amount of concave indentations we have along the boundary. The amount of edge support is determined by the length of the perimeter of the region (l_p) enclosed by the edge segments and the total gap (l_g) between the end points of the consecutive edge segments forming the boundary (as shown in Fig. 8d). The mathematical expression capturing the above is

$$\mathbf{Sig_{clo}}(e_i, \ldots, e_k) = \min\left(\mathbf{Sig}(e_i), \ldots, \mathbf{Sig}(e_j), \frac{l_p}{l_p + l_g}, \frac{a_{\text{actual}}}{a_{\text{hull}}}\right). \quad (10)$$

Note that the closure and strand (defined next) are *N*-ary whereas regular graphs capture binary relationships. We will discuss in Section III(C) how we capture *N*-ary relations using regular graphs.

5. *Strand:* Like the closure relationship, the strand relationship among a set of edge segments, $\{e_i, \ldots, e_k\}$, is determined by the significance of the constituent edge segment and by the amount of edge support we have for the boundary. Mathematically, we capture this using

$$\mathbf{Sig_{str}}(e_i, \ldots, e_k) = \min\left(\mathbf{Sig}(e_i), \ldots, \mathbf{Sig}(e_j), \frac{l_p}{l_p + l_g}\right). \quad (11)$$

6. *Perpendicularity:* The perpendicular relation is defined between two *straight* edge segments (as inferred by the contour segmentation algorithm), e_i and e_j. The significance of the

relation is determined by the significance of the individual edge segments, the minimum gap between the end points of the edges (d_{gap}), and the angle, θ_p, between them (as shown in Fig. 8e). We use the following expression

$$\mathbf{Sig_{per}}(e_i, e_j) = \min\left(\mathbf{Sig}(e_i), \mathbf{Sig}(e_j), \frac{l_i + l_j}{d_{\text{gap}} + l_i + l_j}, \sin^2(\theta_p)\right), \quad (12)$$

where l_i and l_j are the lengths of the edge segments. Note that $\sin^2(\theta_p)$ is one for strictly perpendicular lines and gradually falls to zero as the angle between them decreases. The perpendicular relation encompasses both L- and T-junctions. However, the significance of a T-junction, as computed by the above expression, will be lower than for an L-junction; everything else remaining constant.

7. *Continuity:* The continuity relation is defined between two edge segments e_i and e_j. We decide on the continuity between two edge segments by considering the straight line joining the nearest end points of the two segments. We denote the length of joining the straight line by d_{gap}, the gap between the edge segments. We also find the maximum difference in slope between the joining straight line and the slopes at the two end points of the fitted constant curvature segment, θ_c (as shown in Fig. 8f). Ideally, d_{gap} and θ_c should be zero. We use the following expression to quantify the significance.

$$\mathbf{Sig_{con}}(e_i, e_j) = \min\left(\mathbf{Sig}(e_i), \mathbf{Sig}(e_j), \frac{l_i + l_j}{l_i + l_j + d_{\text{gap}}}, \cos^2(\theta_c)\right), \quad (13)$$

where l_i and l_j are the lengths of the two segments. The $\cos^2(\theta_c)$ term is one for perfectly continuous lines and gracefully goes to zero as the lines become discontinuous. Note that the above definition of continuity holds between any combination of edge segment types, straight or curved.

C. Gestalt Relations: Combination

Next, we combine the quantified Gestalt relations to generate the link weights of the *relation* graph. The link weight $w(e_i, e_j)$ between two nodes is given by

$$\begin{aligned} w(e_i, e_j) = \mathbf{Sig_{prox}}(e_i, e_j) \bullet [&C_1\mathbf{Sig_{para}}(e_i, e_j) \\ &+ C_2\mathbf{Sig_{clo}}(e_i, e_j) + C_3\mathbf{Sig_{str}}(e_i, e_j) \\ &+ C_4\mathbf{Sig_{per}}(e_i, e_j) + C_5\mathbf{Sig_{con}}(e_i, e_j)]. \end{aligned} \quad (14)$$

Note that the relations involved in Eq. (14) are binary, except for the closure and strand relationships. However, the link weights, $w(e_i, e_j)$, are defined pairwise. To facilitate this for N-ary relations, we use the following convention: $\mathbf{Sig_{clo}}(e_i, e_j) = \mathbf{Sig_{clo}}(e_{k_1}, \ldots, e_{k_2})$ for every $\{i, j \mid i, j \epsilon (k_1, \ldots, k_2)\}$. That is, all pairs of the edges involved in an N-ary ($N > 2$) relation are assigned a significance value equal to that for the whole set.

Another important aspect of the combination rule in Eq. (14) is that the proximity factor multiplies the sum of the other factors. Thus, if the proximity factor, $\mathbf{Sig_{prox}}$, between two features is zero, the final weight is zero irrespective of the values of the other factors. The multiplication effects ANDing of the proximity term with the other terms which are ORed together using addition. This has the effect of giving priority to *local* relations. The contribution of the individual terms in the expression can be varied using the weights C_i. In the experiments we found that $C_1 = 1.2$, $C_2 = 1.2$, $C_3 = 0.2$, $C_4 = 1.2$, and $C_5 = 1.2$ gave us good results. As we shall see in Section VII(D) perturbation of these weights do not significantly change the performance.

IV. EFFECT OF IMAGING CONDITIONS ON THE EIGENVALUES OF THE RELATION GRAPH

For the eigenvalues of the relation graph to be useful as measures of change they need to exhibit invariance (at least quasi-invariance) with respect to imaging conditions such as viewpoint, resolution, time of day, and weather. The variation of the eigenvalues will depend on the variation of the weights in the adjacency matrix of the *relation* graph. In Section II(C) we saw that the eigenvalues of a graph are stable under random perturbations of the weights. So, as long as the variations of the weights of the graph are small, the eigenvalues will not change significantly.

The variations of the weights of the relation graph depend on the variations of the Gestalt relationships of parallelism, continuity, perpendicularity, and closure. While these Gestalt relations, except for continuity, are not strictly invariant with respect to viewpoint, they are quasi-invariant. The perspective projection of a set of 3D curves satisfying the closure (or strand) relationship will also be closed. While it is possible in general for the attributes (such as the area) of this projected shape to differ with viewpoint, we do not expect this variance to be significant in the present domain. For aerial imagery we can approximate the imaging geometry as being perspective with objects close to the optic axis. Similarly, the attributes (such as the difference in orientation) of a projected 3D (relaxed) parallelism (or perpendicularity) will also vary slightly with small variation in viewpoint. Variations with respect to the time of day might arise because of presence or absence of shadows. However, as we shall see, the effect of shadows on the eigenvalues is not significant. We do not expect to see drastic change in the eigenvalues with mild changes in the weather as long as the salient structures in the image are visible. However, severe weather conditions such as a snow storm will (understandably) affect the eigenvalues.

While it is difficult to theoretically analyze the variance of the eigenvalues with the imaging conditions, we can offer empirical evidence for their stability. We considered 14 aerial images of the same building under different viewpoints, resolutions, times of day, and cloud conditions, as shown in Fig. 9. Note that Fig. 9d was taken in the early morning and has visible shadows. Figure 9j

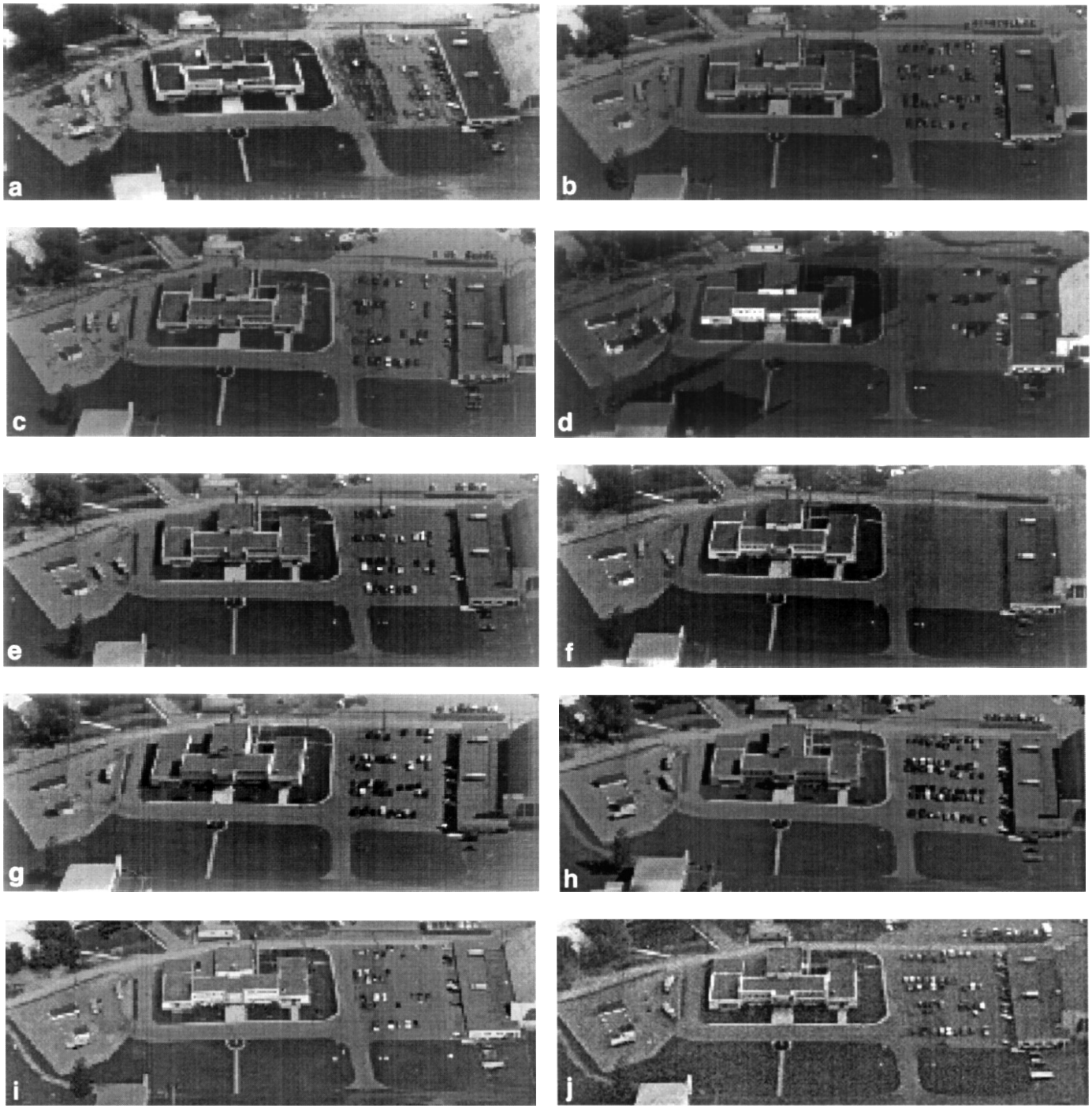

FIG. 9. Database of a *fully* developed site. The image set encompasses variations in viewpoint, resolution, and time of day.

was also taken in the morning but (apparently) under cloud cover since the shadows are not visible. The resolutions of Figs. 9m and 9n are higher than the others and their viewpoints are also different.

For each of the images we computed the edges, the Gestalt graphs, the relation graph, and its associated eigenvalues. The plot in Fig. 10 shows the variation of some of the positive eigenvalues. The horizontal axis denotes the position of the eigenvalues in the spectrum; the first being the largest eigenvalue. We show error bars at one standard deviation on either side of the mean value. The x-axis of the plot corresponds to the ranking of the eigenvalues, the largest eigenvalue being considered first. Note that the variation is modest given the range of imaging conditions we consider. For eigenvalues

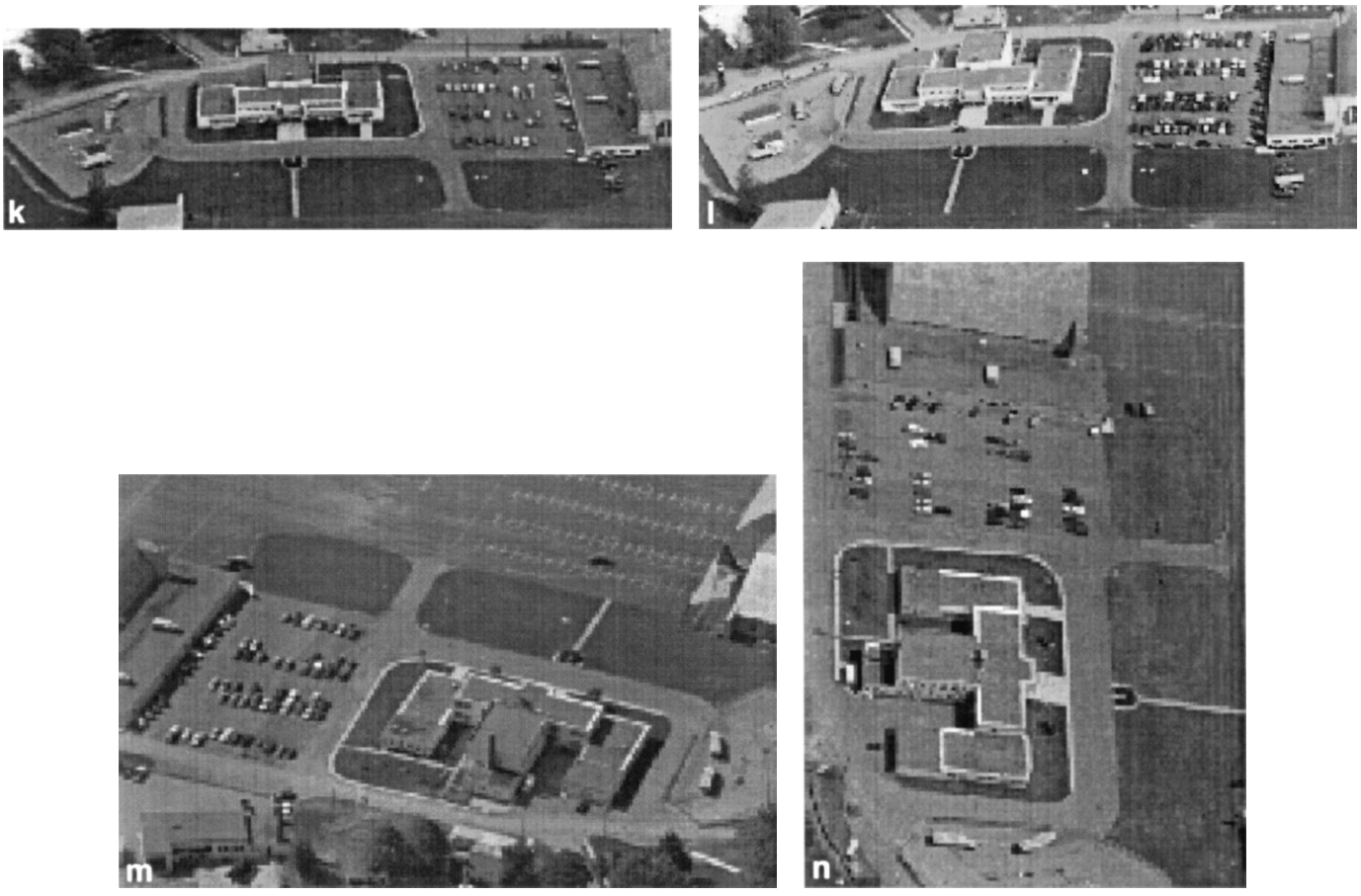

FIG. 9—*Continued*

greater than one, the average ratio of the standard deviation to the mean for each eigenvalue is 0.22. For eigenvalues less than one, the variation ratio is larger at 0.59. In practice, eigenvalues less than one do not represent meaningful underlying clusters.

V. WHAT DO THE EIGENVECTORS GIVE US? EIGENCLUSTERS

Eigenvalues embody important structural aspects of a graph, but what do the eigenvectors give us? As we saw in Section II,

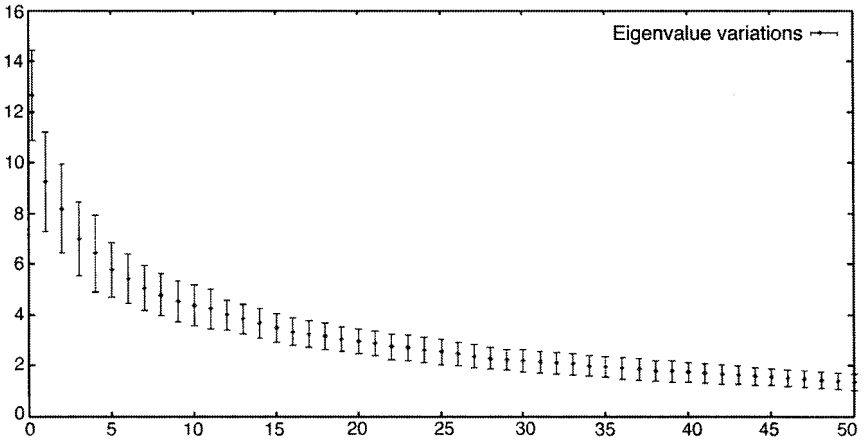

FIG. 10. Variation of the eigenvalues (ordered) corresponding to the site shown in Fig. 9. The center values correspond to the mean eigenvalues and the error bars denote one standard deviation to either side.

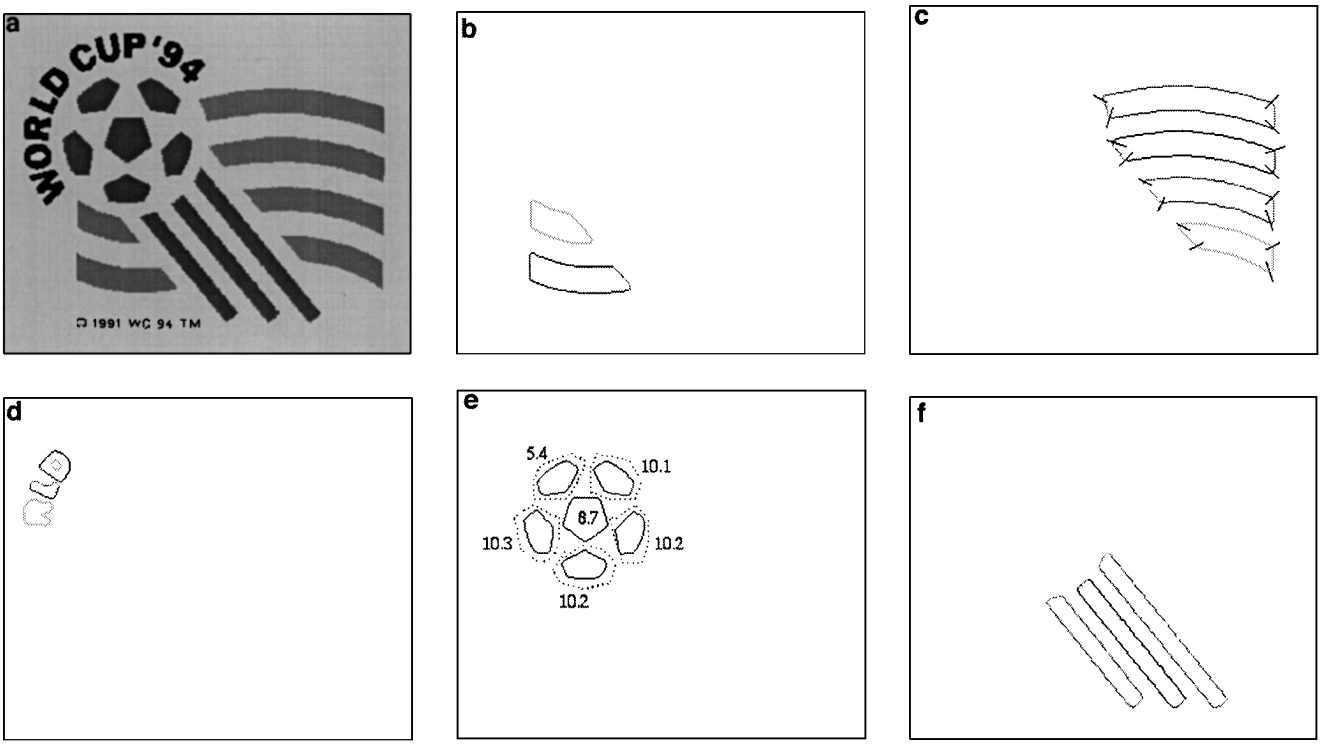

FIG. 11. Examples of eigenclusters. (a) Gray level image. The eigenclusters detected with eigenvalues (b) 12.1, (c) 8.4, (d) 11.3, (e) 10.3, 10.2, 10.2, 10.1, 8.7, 5.4, and (f) 9.3. The intensity of an edge segment is inversely proportional to the weight of the corresponding component in the eigenvector. Thus, a strongly participating edge segment appears dark.

we can consider each eigenvector as representing a weighted clustering of the graph nodes. The greater the magnitude of a component of an eigenvector, the greater the participation of the feature represented by the respective node in the cluster. Strictly, meaningful clusters correspond to positive eigenvectors, those having *all* positive or *all* negative components. We will now relax this definition of positive eigenvectors.

We refer to the components of an eigenvector which account for most (95%) of its total squared value as the *dominant components*. We call the features corresponding to these dominant components the *active* features participating in the eigenvector. The eigenvector is said to be *positive* if its dominant components are all positive or all negative, and if the associated eigenvalue is positive. (Recall that if $\mathbf{x}$ is an eigenvector then so is $-\mathbf{x}$.) We refer to the set of active features corresponding to the dominant components of a positive eigenvector as an *eigencluster*. The relative weight of each active feature in a cluster is determined by the modulus of the corresponding component value in the eigenvector. This weight is a measure of how much a feature "belongs" to an eigencluster; it measures the participation of the part in the whole.

The cohesiveness of an eigencluster is determined by the eigenvalue associated with the *positive* eigenvector. A completely cohesive cluster of n nodes has an underlying graph structure which is complete with all link weights equal to 1 and the associated eigenvalue is the maximum possible, $n - 1$. For a completely incohesive cluster the underlying graph structure will consist only of isolated nodes and the associated eigenvalue will be low.

Figures 11b–11f show some eigenclusters detected in Fig. 11a. Note that the eigenclusters correspond to the salient edge groupings in the image. The gray level encodes the participation of an edge segment in an eigencluster. A strongly participating edge appears dark. The eigenvalues are shown in Fig. 12a.

The eigenvector corresponding to the eigencluster in Fig. 11c is shown Fig. 12b. Note the few nonzero components corresponding to the structures in the cluster. The constant curvature segments have been marked in Fig. 11c. The participation of an edge segment in a cluster is proportional to the corresponding component (marked as dots) in the eigenvector. The four left-most positive components of the eigenvector correspond to the top-most strip of edge segments in Fig. 11c. And the four right-most positive components correspond to the bottom-most strip. Because of the small size of the bottom-most strip, its participation is low in the eigencluster.

Another example is shown in Fig. 13, which displays four eigenclusters detected in the image shown in Fig. 1. Again, the darkness of an edge segment in a cluster is proportional to the participation of the segment. Note that the eigenclusters correspond to salient structures such as buildings and roads.

We also observe that there is overlap between the eigenclusters in the above examples. However in Section II(A), we claimed

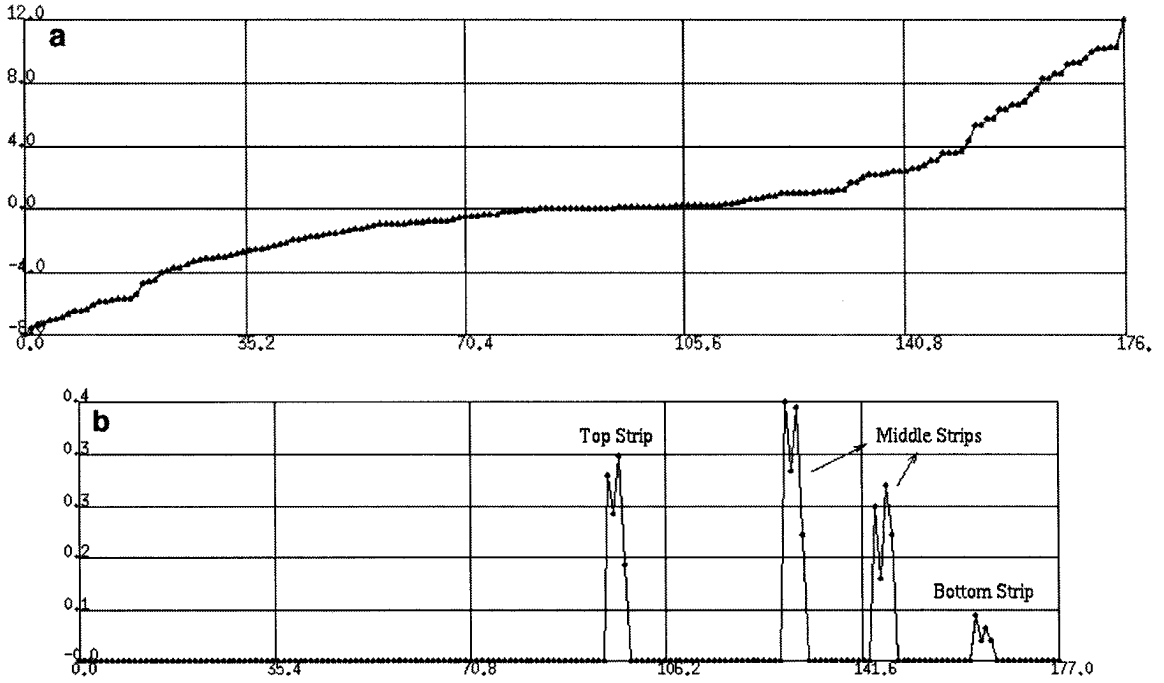

FIG. 12. (a) Spectrum of eigenvalues (ordered) of the relation graph for the World Cup Soccer logo image. (b) Eigenvector corresponding to the eigencluster shown in Fig. 11c.

that since the eigenvectors of a Hermitian matrix are orthogonal, the eigenclusters, which are based on *positive* eigenvectors, will be disjoint. This is not contradictory. The observed overlap between the clusters is because of our relaxed definition of *positive* eigenvectors to include up to 5% negative components. Thus, although any two eigenvectors are still orthogonal, the corresponding eigenclusters may have overlap between them over the nodes corresponding to the negative elements of the eigenvectors. But, the amount of overlap will be restricted to (weighted) 5% of the total cluster weight.

VI. MEASURES FOR CHANGE DETECTION

With the structural change occurring at a site we will observe not only change in the individual image feature attributes, but also in the evolution of relations among these features. We have seen in the previous sections that the eigenvalues of the *relation* graph are a good and stable indicator of its structure. Also, the eigenvectors corresponding to these eigenvalues provide us with information about the salient clusters. We use these eigenvalues and eigenvectors to formulate four measures to monitor change.

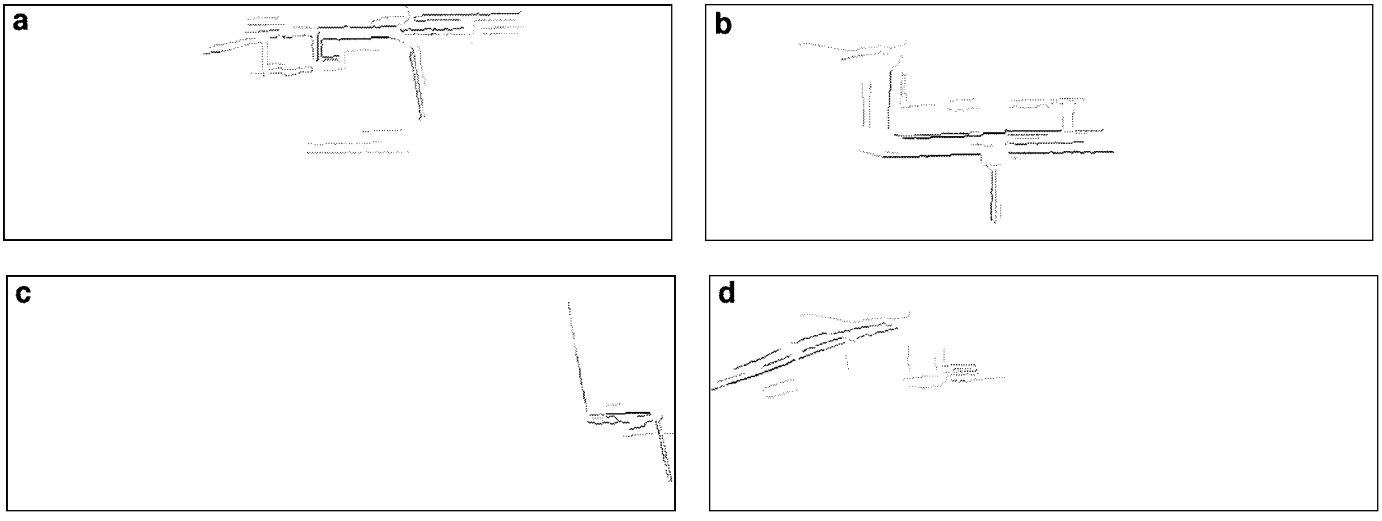

FIG. 13. Eigenclusters detected in Fig. 1. The eigenclusters detected with eigenvalues (a) 5.8, (b) 4.6, (c) 3.9, and (d) 3.5. The intensity of an edge segment is inversely proportional to the weight of the corresponding component in the eigenvector. Thus, a strongly participating edge segment appears dark.

1. l^c_{tot}: This is given by

$$l^c_{\text{tot}} = \left(\frac{\sum_{i=1}^{N_c} \left(\sum_{j=1}^{n_i} l_{ij} \right)}{N_c} \right) N, \tag{15}$$

where l_{ij} is the length of the jth edge segment in the ith eigencluster, normalized by the square root of the image area to account for differences in image resolution. We denote the total number of edge segments by N and the total number of clusters by N_c. The ratio in parentheses will be high for images with small numbers of large clusters as in a fully developed site and will be small for images with large numbers of small clusters as in an undeveloped site. The multiplication by the total number of edge segments in the image takes into account the edge segments which do not participate in any eigencluster. Thus, the measure can capture the difference between two images with similar sized eigenclusters but with different total number of edge segments.

2. N_c: This is the total number of *eigenclusters* in the image. We expect the number of clusters to change with development.

3. λ_+: We consider the sum of the positive eigenvalues in the spectrum of the *relation* graph. This is an indicator of the total amount of *structure* in the image. This should be large for a fully developed site.

4. $\text{Dist}(\mathcal{G}_{t_1}, \mathcal{G}_{t_2})$: This distance measure compares the spectra from two images. Let $\mathcal{G}_{t_1}$ and $\mathcal{G}_{t_2}$ be two *relation* graphs of n_1 and n_2 nodes with $\{\lambda_1, \ldots, \lambda_{n_1}\}$ and $\{\mu_1, \ldots, \mu_{n_2}\}$ as their spectra, respectively. The difference between the spectra provides us with a distance measure between two graphs [16]. However, not all eigenvalues in the spectrum are of equal importance. The positive eigenvalues are of particular importance in our case because they capture the amount of structure in an image. Hence we consider the difference between only the 20 largest eigenvalues from the two images. We normalize this difference as shown below:

$$\text{Dist}\left(\mathcal{G}_{t_1}, \mathcal{G}_{t_2}\right) = \left(\frac{\sum_{i=1}^{20} (\lambda_i - \mu_i)^2}{\min\left(\sum_i \lambda_i^2, \sum_j \mu_j^2 \right)} \right)^{1/2}. \tag{16}$$

The choice of 20 largest eigenvalues is based on empirical studies. We found that there is little new information to be gained from eigenvalues beyond the first 20; the distributions of the smaller eigenvalues have significant overlaps.

We use the above four measures to infer change at a site. We first compute the difference between the measures computed from the images taken at two time instants, t_1 and t_2. These differences are then normalized to formulate four classification features with which we categorize the image pair into two classes: change or no-change. The four classification features are given by:

$$\begin{aligned} f_1(t_1, t_2) &= \frac{\left| l^{t_1}_{\text{tot}} - l^{t_2}_{\text{tot}} \right|}{\min\left(l^{t_1}_{\text{tot}}, l^{t_2}_{\text{tot}} \right)} \\ f_2(t_1, t_2) &= \frac{\left| N_c^{t_1} - N_c^{t_2} \right|}{\min\left(N_c^{t_1}, N_c^{t_2} \right)} \\ f_3(t_1, t_2) &= \frac{\left| \lambda_+^{t_1} - \lambda_+^{t_2} \right|}{\min\left(\lambda_+^{t_1}, \lambda_+^{t_2} \right)} \\ f_4(t_1, t_2) &= \text{Dist}\left(\mathcal{G}_{t_1}, \mathcal{G}_{t_2} \right). \end{aligned} \tag{17}$$

Note that the Dist measure is normalized as formulated. These four classification features form the feature vector which is used in a Bayesian classifier. The Bayesian classifier places each image into one of two classes, change or no-change, based on the assumption that the feature vectors for each class exhibit a multivariate normal distribution. For more details on Bayesian classification we refer the reader to Duda and Hart [17]. We chose Bayesian methods because they constitute a fundamental statistical approach to pattern classification and they suffice to illustrate the change detection strategy of this paper. Other forms of classifiers such as neural networks, decision trees, and nonparametric techniques can be topics of future research.

VII. RESULTS

We first investigate the ability of the proposed eigenvalue based classification features (Eq. (17)) to detect drastic change at a site. We then explore the possibility of detecting coarsely incremental change. One might argue that simple statistics might suffice for change detection and that a costly method based on eigenvalues constitutes overkill. To allay this skepticism we also compare the performance of the eigenvalue based approach against a simple statistical detector based on the number of parallel, continuous, and perpendicular segments. We show that the eigenvalue-based method outperforms the simple change detector.

We formalize the simple statistical detector as follows. Let the number of parallel, continuous, and perpendicular relations be denoted by N_{para}, N_{cont}, and N_{per}, respectively. We denote the total number of edge segments by N. Then, three simple measures of change can be formulated to be the ratios $n_{\text{para}} = \frac{N_{\text{para}}}{N}$, $n_{\text{cont}} = \frac{N_{\text{cont}}}{N}$, and $n_{\text{per}} = \frac{N_{\text{per}}}{N}$, denoting the fraction of edge segments exhibiting the respective relations. A simple statistical change detector can be designed using differences of these ratios between images taken at two time instants, t_1 and t_2.

$$\begin{aligned} F_1(t_1, t_2) &= \frac{\left| n^{t_1}_{\text{para}} - n^{t_2}_{\text{para}} \right|}{\min\left| n^{t_1}_{\text{para}}, n^{t_2}_{\text{para}} \right|} \\ F_2(t_1, t_2) &= \frac{\left| n^{t_1}_{\text{cont}} - n^{t_2}_{\text{cont}} \right|}{\min\left| n^{t_1}_{\text{cont}}, n^{t_2}_{\text{cont}} \right|} \\ F_3(t_1, t_2) &= \frac{\left| n^{t_1}_{\text{per}} - n^{t_2}_{\text{per}} \right|}{\min\left| n^{t_1}_{\text{per}}, n^{t_2}_{\text{per}} \right|} \end{aligned} \tag{18}$$

We use these simple classification features in a Bayesian

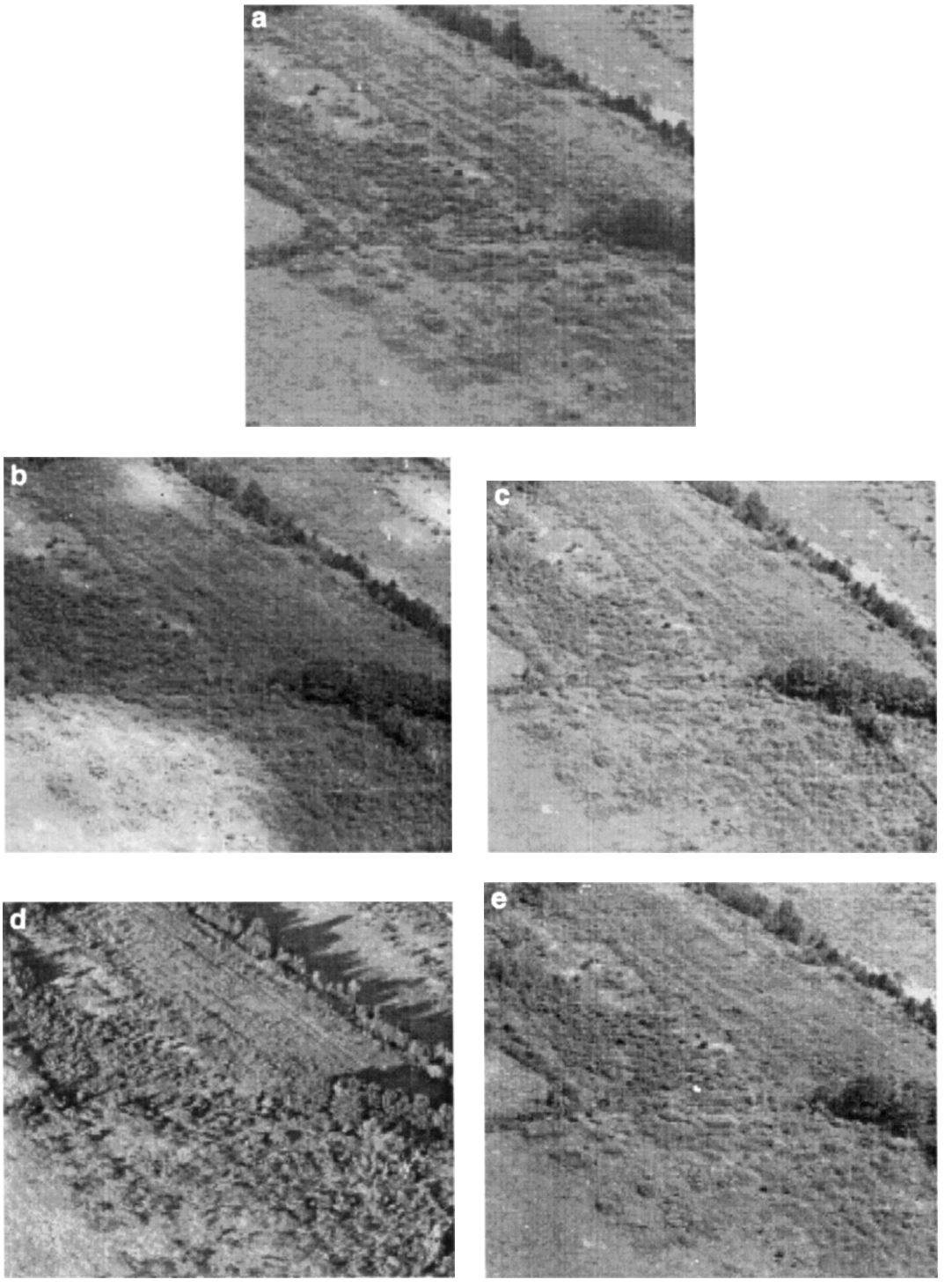

FIG. 14. Database of an *un*developed site at different times and days.

decision scheme to decide whether there is change or no-change between two time instants.

A. Detecting Drastic Change

First, we demonstrate the ability of the proposed measures to detect drastic change from an undeveloped site to a fully developed one. As a test bed we consider the images of the fully constructed site shown in Fig. 9 and the undeveloped forest site images shown in Fig. 14. The image sets account for different viewpoints, different resolutions, different times of the day, and different cloud conditions. Note the partial cloud cover in Fig. 14b, the long shadows in Fig. 14d, and the fog cover in Fig. 14j.

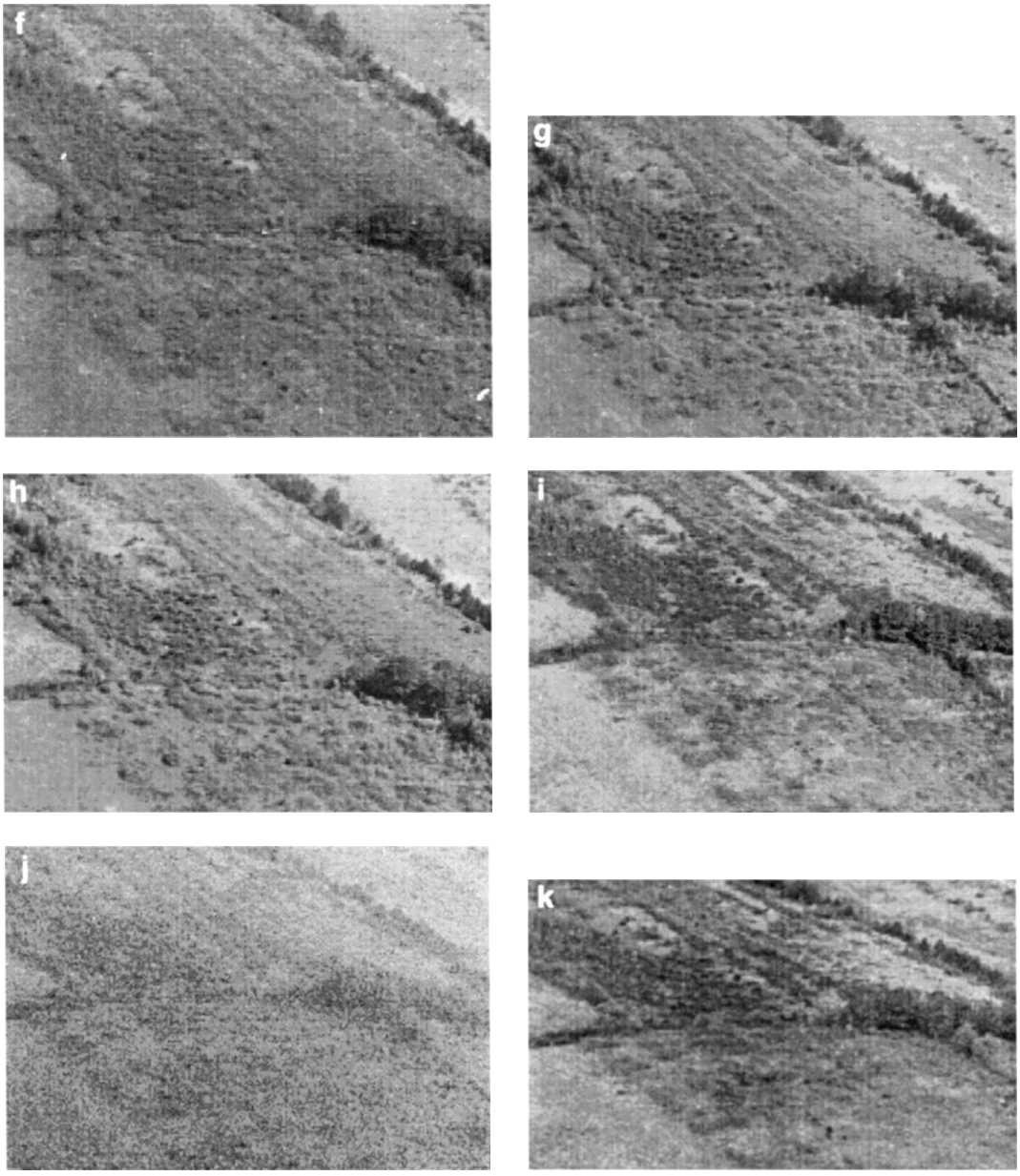

FIG. 14—*Continued*

Since an undeveloped site exhibits significantly less structure than a fully developed site we expect the eigenvalues of the relation graph for an undeveloped site to be significantly different from those for a fully developed site. Figure 15 shows the statistics of 25 largest positive eigenvalues for the two classes. The error bars correspond to one standard deviation. Note that the eigenvalues for the two classes are significantly different from each other. This demonstrates that the eigenvalues can act as basis for discriminating measures. Also, note that the distribution of the eigenvalues in the spectra for each class tend to overlap for smaller values. Thus, the range for the 15th largest eigenvalue overlaps with that for the 16th largest eigenvalue. There is little new information in considering eigenvalues beyond the first 20. This justifies the use of 20 largest eigenvalues in formulating the fourth measure of change, Dist($\mathcal{G}_{t_1}, \mathcal{G}_{t_2}$) in Section VI.

Table 2 lists the values of three of the four proposed measures, the total lengths of the edges in the eigenclusters (l_{tot}), the number of Eigenclusters (N_c), and the sum of the positive eigenvalues (λ_+) for a fully developed site and an undeveloped site, respectively. The tables also list the 95% confidence intervals for the means of the measures. Note that the confidence intervals for the fully developed sites are clearly separated from those of the undeveloped ones.

As we saw in Section IV (Eq. (17)), the indicators for change on the ground are based on fractional changes in the proposed

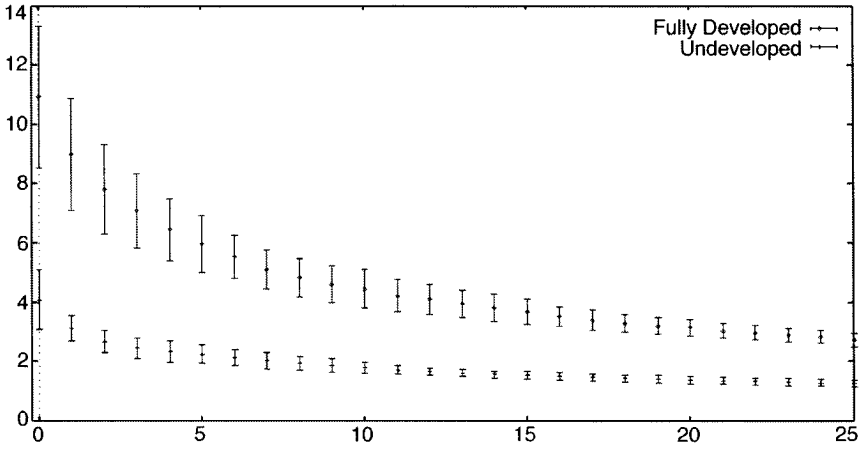

FIG. 15 Variation of the positive eigenvalues over 11 images from the undeveloped site and 14 images from the developed site. The top plot corresponds to the fully developed site and the bottom plot corresponds to the undeveloped forest site.

TABLE 2

The Total Length of the Edges (l^c_{tot}), the Total Number of Clusters (N_c), and the Sum of the Positive Eigenvalues (λ_+) for the Images of (a) the Fully Developed Site Shown in Fig. 9 and (b) the Undeveloped Forest Site Shown in Fig. 14

Images	l^c_{tot}	N_c	λ_+
		a	
(a)	6.614	20.000	273.230
(b)	9.916	16.000	326.814
(c)	6.911	19.000	398.514
(d)	6.666	26.000	263.939
(e)	9.961	14.000	382.496
(f)	11.012	18.000	394.622
(g)	9.625	21.000	337.004
(h)	10.219	15.000	323.417
(i)	7.917	16.000	305.675
(j)	6.333	26.000	319.922
(k)	7.577	20.000	327.342
(l)	5.847	30.000	310.811
(m)	4.309	41.000	256.921
(n)	5.903	32.000	524.623
Mean	7.772	22.429	338.95
95% Conf	6.596–8.949	17.975–26.882	298.92–378.98
Std. Dev.	2.037	7.713	69.33
		b	
(a)	0.280	17.000	62.258
(b)	0.246	12.000	42.326
(c)	0.186	18.000	67.754
(d)	0.171	8.000	48.476
(e)	0.209	20.000	59.717
(f)	0.207	14.000	49.009
(g)	0.200	19.000	83.539
(h)	0.244	16.000	57.053
(i)	0.255	18.000	82.183
(j)	0.175	11.000	51.185
(k)	0.268	16.000	58.869
Mean	0.222	15.364	60.215
95% Conf	0.196–0.248	12.863–17.865	51.291–69.139
Std. Dev.	0.038	3.72	12.284

measures. So next we investigate the distribution of these classification features, $f_1(t_1, t_2)$, $f_2(t_1, t_2)$, $f_3(t_1, t_2)$, and $f_4(t_1, t_2)$ for the two classes: change and no-change. The distribution of the classification features are shown as 3D scatter plots in Fig. 16. The red crosses represent change and the blue circles denote no-change. Note that the distributions are very well separated.

From Fig. 16 we see that the means of the classification features for the two classes, change and no-change, are well separated. As mentioned before (Section VI), we signal change between two images using a Bayesian classifier. If we assume that the classification features are from a multivariate Gaussian distribution, the Bayesian classifier results in a hyperquadric discriminant surface in the feature space [17]. Discriminant surfaces partition the feature space into regions belonging to each class.

To analyze the performance of the change detector we estimate the parameters of the Bayesian classifier on a subset of the data and test the classifier on another data subset. This method of estimating performance is also known as cross-validation [18]. Cross-validation maximizes the use of the available data by reusing data subsets for training and testing. For example, in our case we have 14 images of a fully developed site and 11 images of an undeveloped site. Thus, we have 300 pairs of images of which 154 are pairs of images with change. We randomly divide these 300 image pairs into 5 *disjoint* subsets of roughly equal size. We estimate the parameters of the Bayesian classifier using four of these subsets and test the performance on the one left out. Since there are five different ways of choosing the four training subsets, we test the performance five times. The average over the five tests captures the classifier performance. This method is also known as 5-way cross-validation.

We characterize the performance of the Bayesian classifier in terms of the false alarm and true detection rates. A false alarm occurs when a change is flagged in the presence of a no-change. A true detection occurs when a change is correctly flagged in the presence of an actual change. For each test data point the

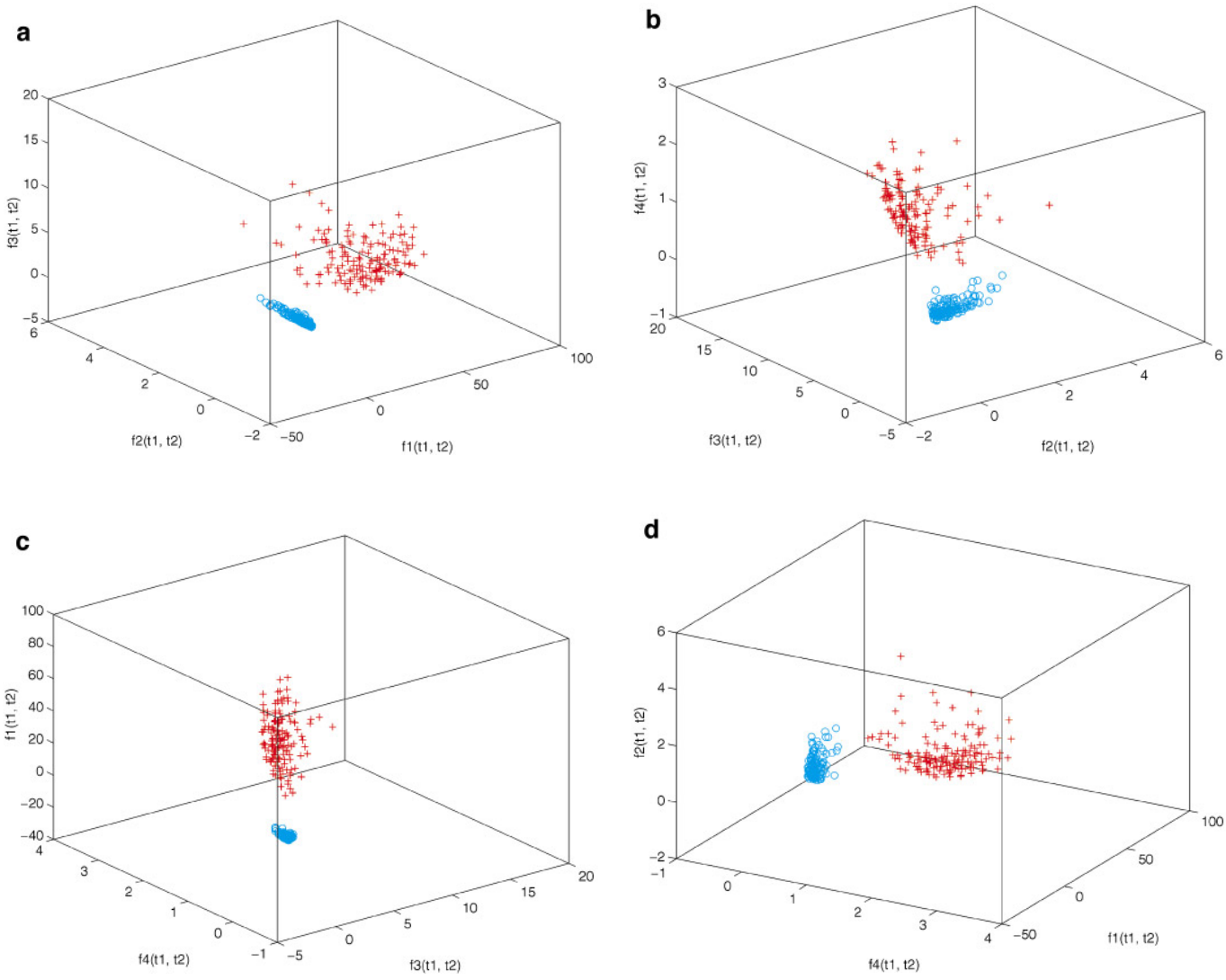

FIG. 16 The joint distribution of the classification features $f_1(t_1, t_2)$, $f_2(t_1, t_2)$, $f_3(t_1, t_2)$, $f_4(t_1, t_2)$ (Eq. (17)) for the two classes, change and no-change, are shown above. The red crosses denote measurements for the change-class and the blue circles denote the no-change class. The individual scatter plots are for: (a) $f_1(t_1, t_2)$, $f_2(t_1, t_2)$, and $f_3(t_1, t_2)$, (b) $f_2(t_1, t_2)$, $f_3(t_1, t_2)$, and $f_4(t_1, t_2)$, (c) $f_3(t_1, t_2)$, $f_4(t_1, t_2)$, and $f_1(t_1, t_2)$, (d) $f_4(t_1, t_2)$, $f_1(t_1, t_2)$, and $f_2(t_1, t_2)$.

Bayesian classifier computes the probability of the data being in a target class. We threshold this probability with different values to generate different false alarm and detection rates. We plot these rates to form the receiver operating characteristics (ROC). The area under these ROCs characterizes the performance of an algorithm and can form a basis for comparing two algorithms. Ideally, the area under the ROC should be one. The optimal ROC curve consists of the lines corresponding to a zero false alarm rate and a detection rate of one. In the worst case, the area is 0.5 corresponding to the straight ROC curve with slope 1 passing through the origin.

The ROCs of the eigenvalue based approach and the simple statistical approach are shown in Fig. 17. The left plot corresponds to the eigenvalue based approach and the right plot is for the simple statistical approach. Note that these ROCs are the averages of the five different cross-validations. We see that the eigenvalue based approach clearly outperforms the simple statistical approach for drastic change detection at a site. The area under the ROC for the eigenvalue approach is 1, whereas, the area under the ROC for the simple statistical approach is 0.972.

B. Detecting Coarsely Incremental Changes

In the previous section we saw that the proposed measures can be used to detect drastic changes from no to full development. Here we explore the ability to discriminate between coarsely incremental change. We define coarse change as change which affects at least 5 to 10% of the image. This is as opposed to fine change which affects just a small fraction of the image such as the changing door states of a big warehouse building.

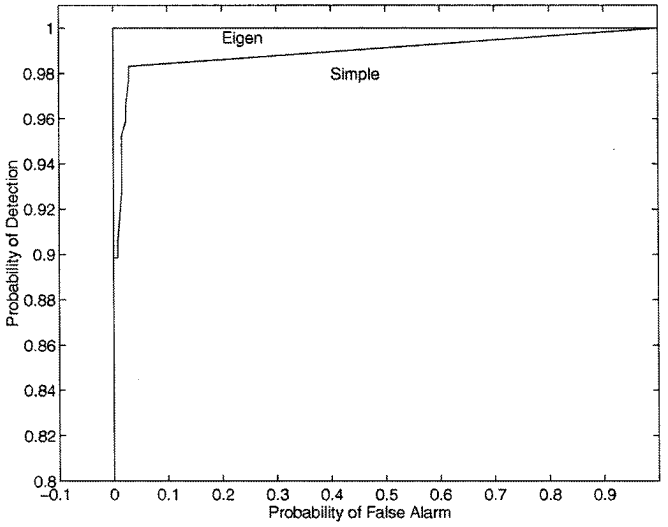

FIG. 17 Receiver operating characteristics for the proposed eigenvalue-based approach and the simple statistical approach in the context of *drastic* change detection. The vertical axis is for the true detection rate and the horizontal axis represents the false alarm rate.

We expand the data set from 25 to 104 images. This expanded set includes real images of construction activity in the publicly available RADIUS imagery and manually altered images. Figure 18 shows images of a construction site on different days. The total duration is 42 days. Figure 18a corresponds to the ground clearing stage and Fig. 18k shows the same site when the construction has advanced to the laying of the foundation walls. This set of images covers the preliminary construction stage. Due to the paucity of real images of a site undergoing change, we also generated sets of images by incrementally "erasing" buildings and road in images of fully constructed sites. Figure 19 shows such a sequence of changes. Such changes might occur because of construction activity or demolition. The set of 104 images also includes three other fully developed sites viewed on different days and under different weather conditions. In total, the database consists of images from 11 different sites.

We compute the ROCs on these 104 images using 5-way cross validation as described earlier. The ROCs are shown in Fig. 20. The left plot is for the proposed eigenvalue-based approach and the right plot is for the simple statistical approach. Note that the performance of the eigenvalue approach is clearly better than the simple statistical approach. The area under the ROC for the eigenvalue approach is 0.954 and that for the simple statistical approach is 0.845. For a given detection rate, the eigenvalue approach will result in far fewer cases to be falsely classified as changes. This is not surprising. The eigenvalues capture the global structure more comprehensively than the statistics of local relations. Thus, the low false alarm rate justifies the added expense of computing eigenvalues.

C. Variation with Edge Detection Parameters

The input to the change detection algorithm consists of edge segments. This led us to question the sensitivity of the algorithm with respect to the choice of the edge detection parameters. All the results shown so far in this paper are with the Canny detector for a σ of 1.0. In this section we study the effect of the variation of this smoothing parameter over a range of values: 1.0, 1.2, 1.5, 2.0, 2.5, 3.0, 5.0, 6.0, 7.0, and 10.0. For each value we compute the ROC on the test image database of 104 images. Figure 21 shows the various ROCs for different σs. We see that in general the performance decreases with increasing edge detector scale. However, the variation in performance over the range 1.0 to 3.0 is small. Performance degradation for large edge detector scales is not surprising. The proposed eigenvalue-based change detector uses edge structure built out of relations such as perpendicularity and parallelism as cue for change. And it is a well-known result that at larger scales the Canny edge detector tends to "round off" corners and other sharp 2D features. This, in turn, leads to reduced organized geometric structure (of the kind we are looking for) in the edge image, hence, the decreased performance. However for a reasonable range of edge detector scales (1.0 to 3.0) we can conclude that the performance of the eigenvalue approach is stable.

D. Variation with Relation Graph Parameters

As we saw in Section III(C), the link weights of the *relation* graph are weighted combinations of the significance of the individual Gestalt relations (see Eq. (14)). For all the experiments presented so far the weights are chosen to be: $C_1 = 1.2$, $C_2 = 1.2$, $C_3 = 0.2$, $C_4 = 1.2$, and $C_5 = 1.2$. To test the robustness of the algorithm with respect to these weights we considered 10 different combinations. In Fig. 22 we present the ROCs for five of these 10 weights to illustrate the range of variation. The ROCs of the other five images were within the variation shown; we omit them in the plot for clarity. Note that the performance of the eigen approach is stable with respect to the change in the C_is. The worst performance (lower-most ROC curve) was with the second combination shown in Table 3 where the cycle, strand, and perpendicular relations are weighted more than parallelism and continuity.

E. Limitations of the Approach

As we have seen, the proposed eigenvalue-based measures can very effectively detect change, both drastic and coarsely incremental, at a site. However, it is unable to detect fine incremental changes such as those observed between Figs. 18j and 18k where there are some trailers in the lower left corner of one

TABLE 3
Various Combinations of the Weights, C_is, of Eq. (14)

	C_1	C_2	C_3	C_4	C_5
Set1	1.2	1.2	0.2	1.2	1.2
Set2	0.2	1.8	1.8	0.2	1.0
Set3	1.0	1.0	1.0	1.0	1.0
Set4	1.8	1.8	0.2	0.2	1.0
Set5	1.8	0.2	1.8	0.2	1.0

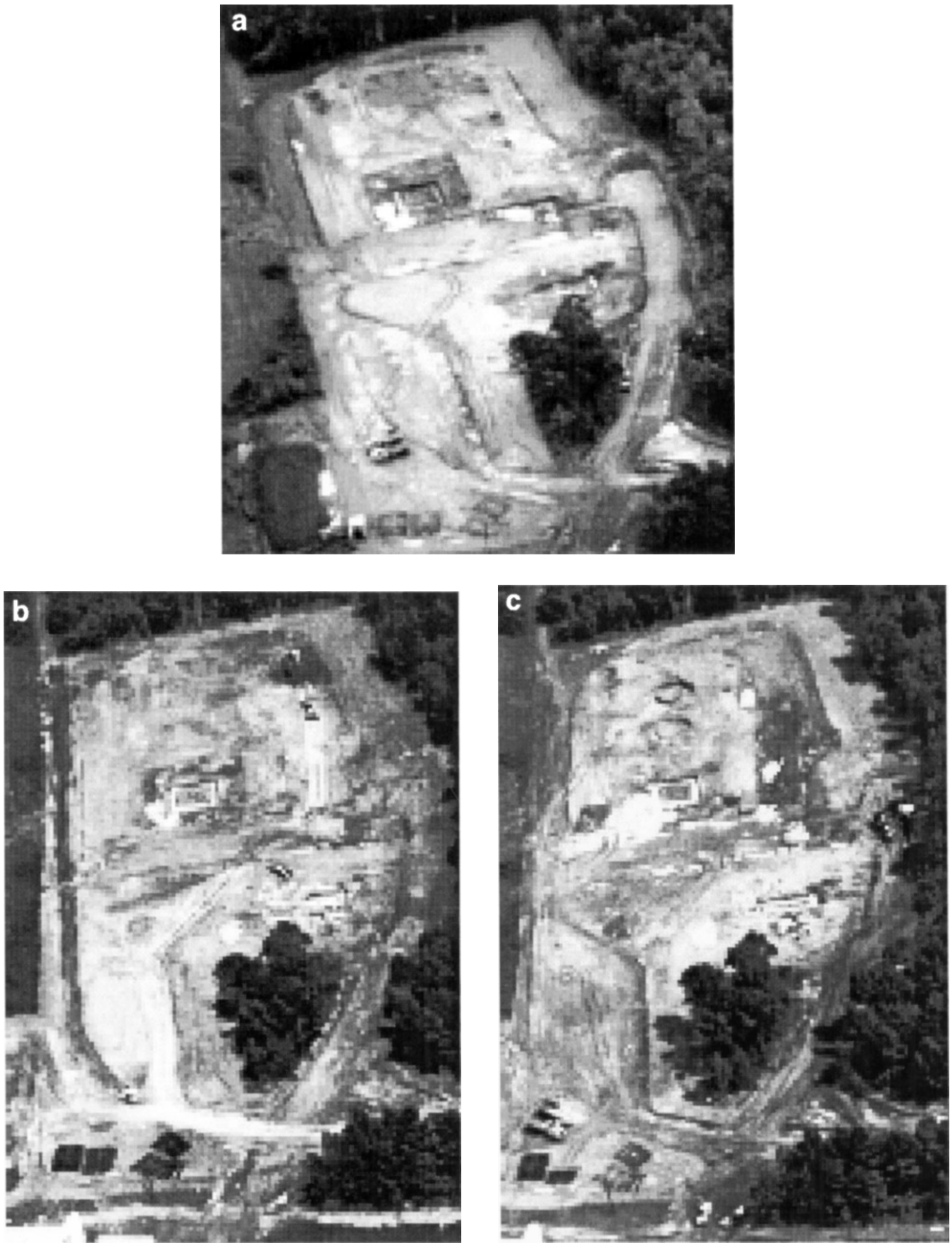

FIG. 18 A site under construction at different times and days: (a) 8/24 at 1700 h, (b) 8/26 at 1300 h, (c) 8/30 at 1150 h, (d) 8/31 at 810 h, (e) 9/2 at 1121 h, (f) 9/3 at 1230 h, (g) 9/7 at 945 h, (h) 9/9 at 1046 h, (i) 9/29 at 1426 h, (j) 9/30 at 845 h, and (k) 10/5 at 1122 h.

image and not in the other. This is not surprising because the proposed measures capture *global* change in the image. To monitor incremental change we have to do fine-scale feature matching. Another scenario where the proposed algorithm might fail is when we replace old objects with new ones having same general structure as the old ones. For example, we demolish a building and build a new one with a different style but preserving the similar characteristics, i.e., preserving the gross rectilinear structure. Both the old and new building would tend to have the same gross structural characteristics. Again, in this case to solve the problem we need to conduct fine-scale feature correspondences.

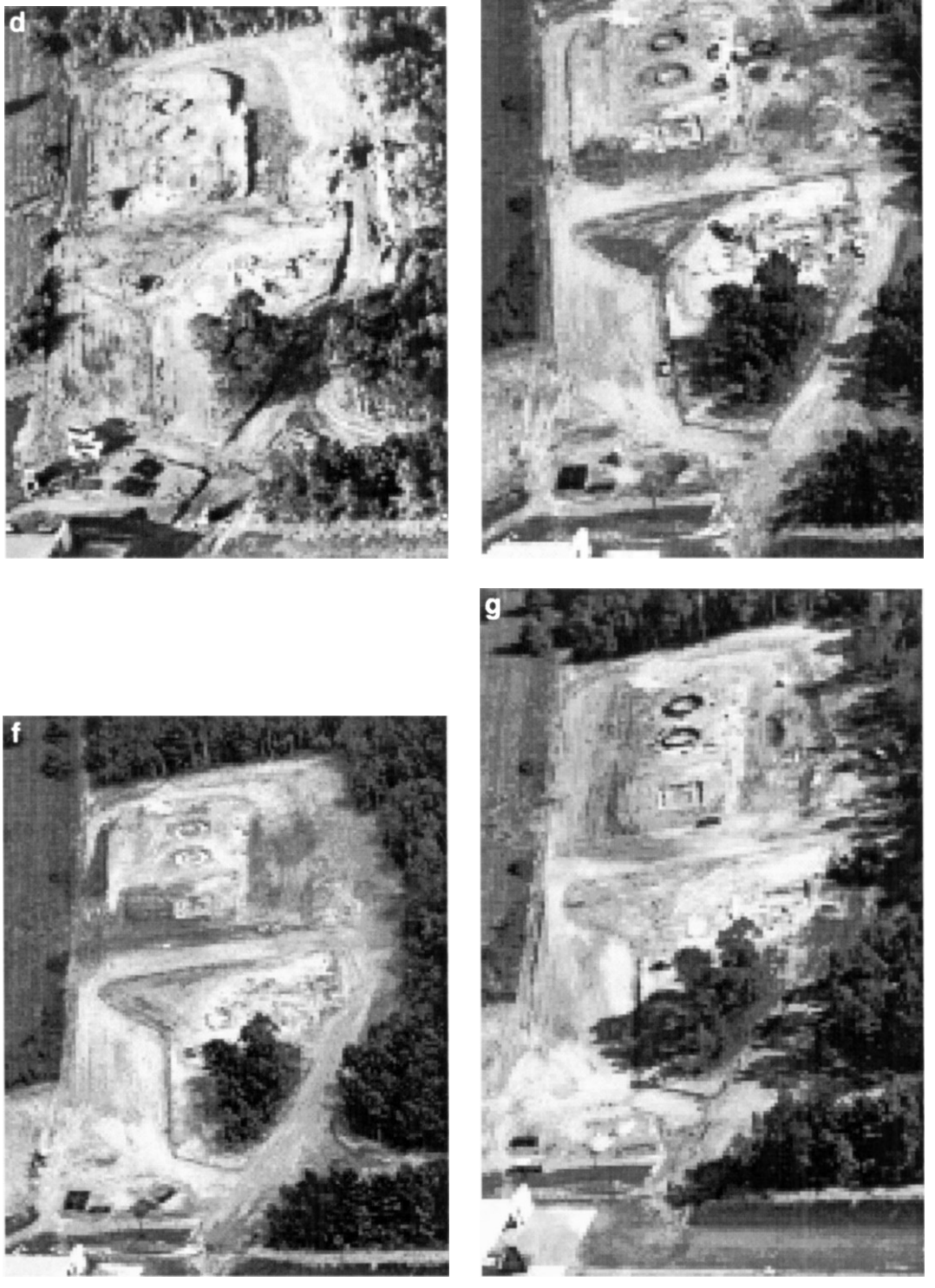

FIG. 18—*Continued*

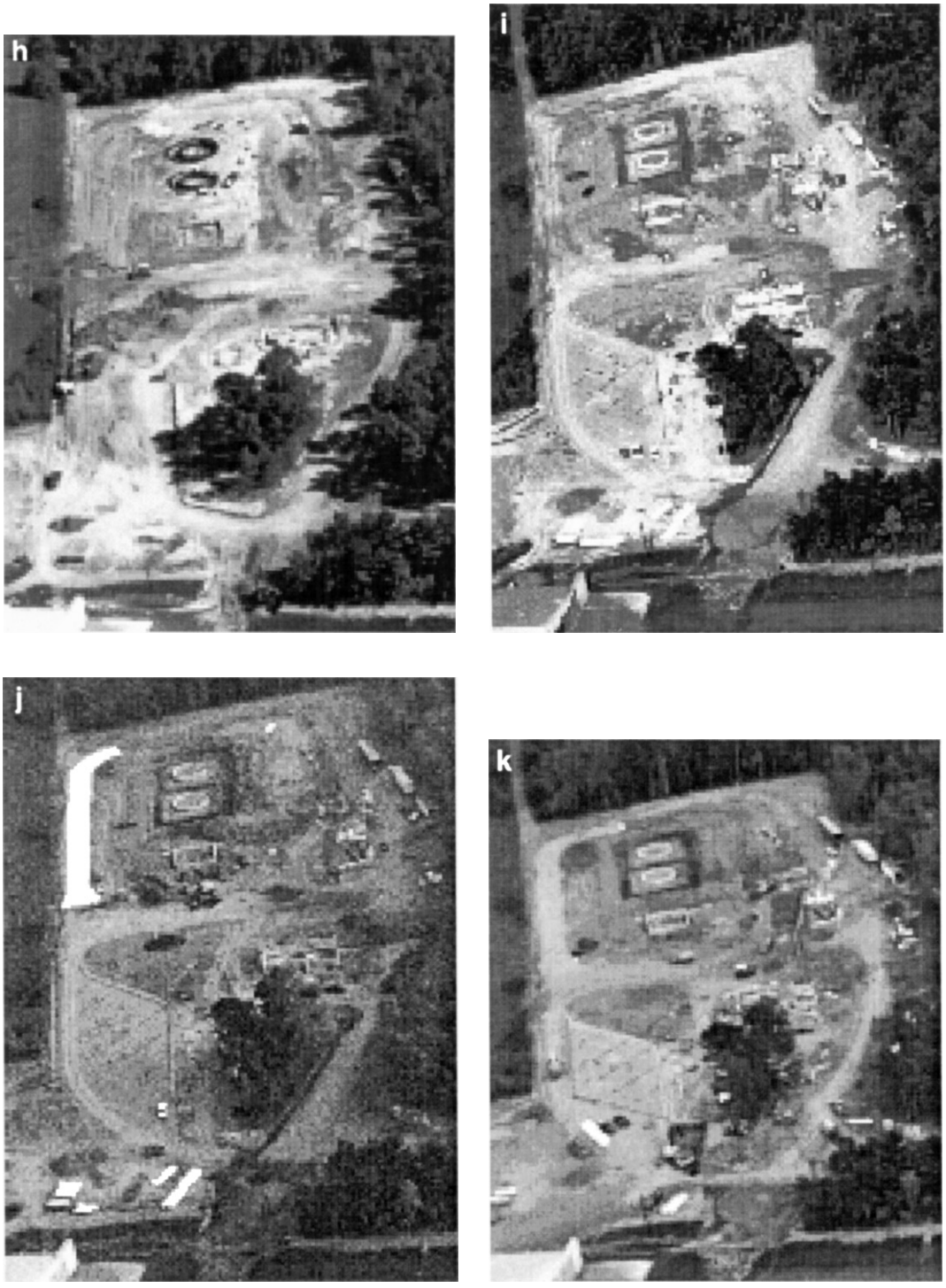

FIG. 18—*Continued*

F. Execution Times

The algorithm is implemented in C and has been tested on a Sun Ultra Sparc workstation. The average (total) execution time over all the images in this paper was 50 s including file I/O. The average (total) CPU time was 40 s. Of this, an average of 6 s was spent on edge detection and segmentation, and 11 s of CPU time was spent on preattentive organization.

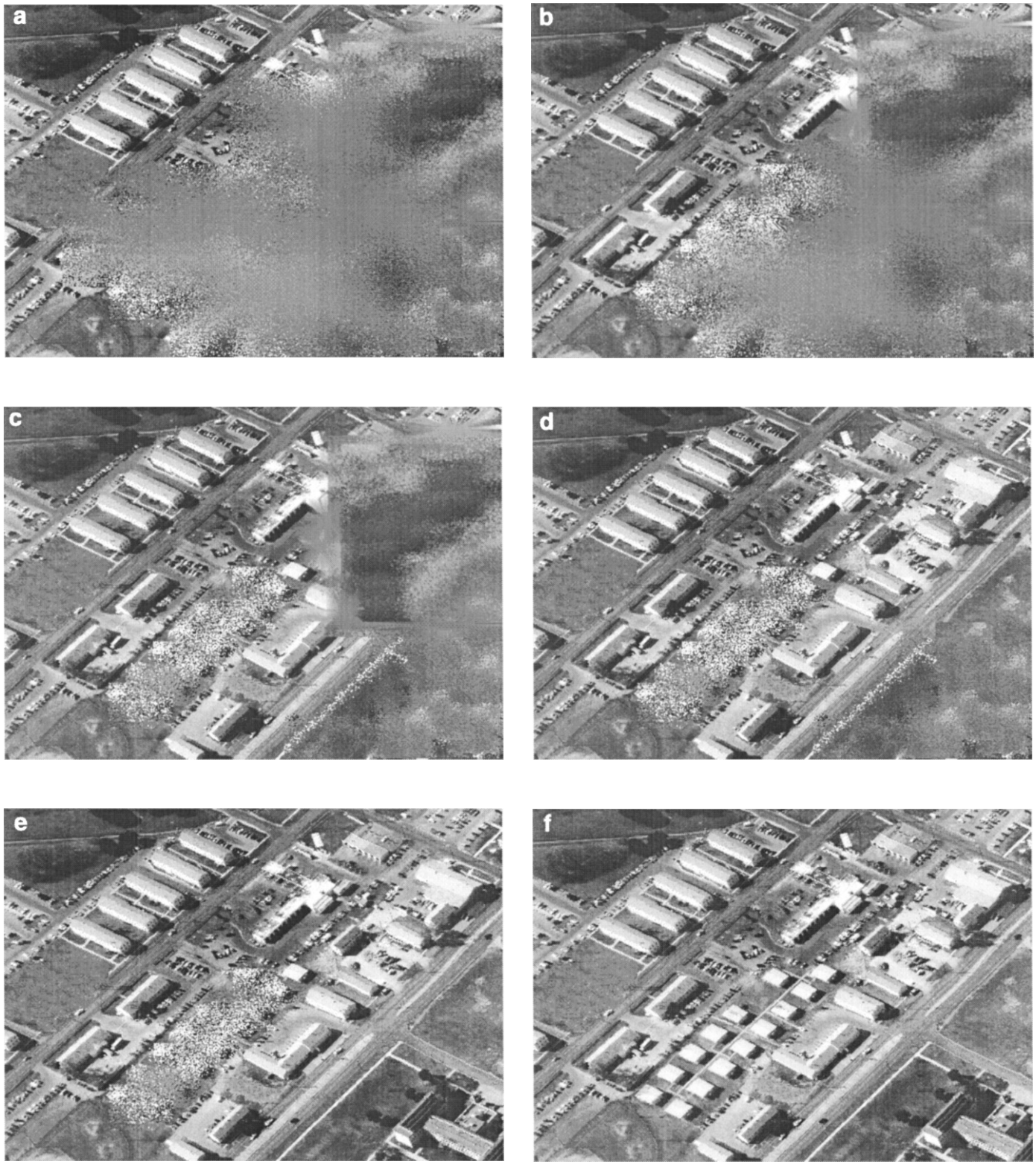

FIG. 19 A site with simulated change.

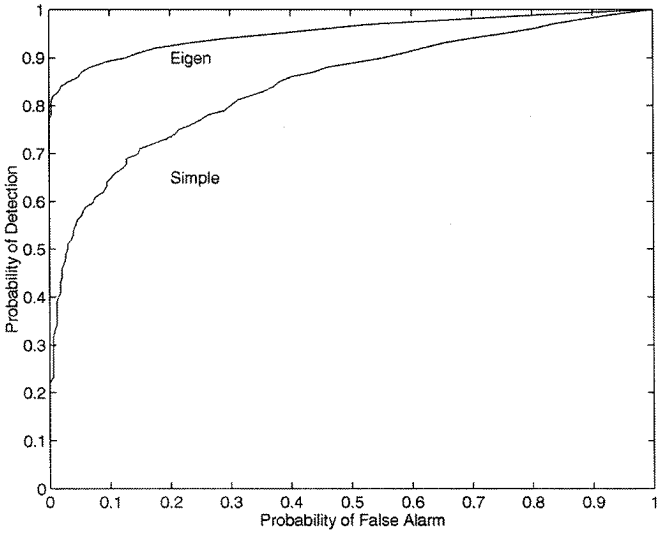

FIG. 20 Receiver operating characteristics of the proposed eigenvalue-based approach and the simple statistical approach in the context of *incremental* change detection. The vertical axis is for the true detection rate and the horizontal axis represents the false alarm rate.

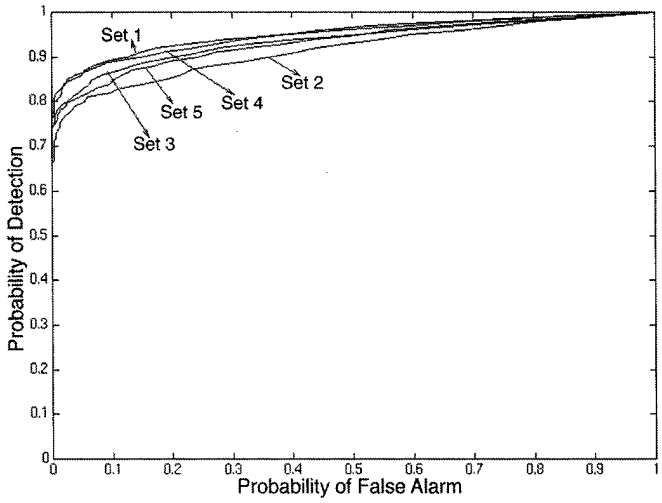

FIG. 22 Receiver operating characteristics of the proposed eigenvalue-based approach for different Relation graph weight combinations shown in Table 3. The vertical axis is for the true detection rate and the horizontal axis represents the false alarm rate.

The average CPU time spent computing the eigenvalues and eigenvectors was 23 s. The software to compute the eigenvalues and eigenvectors was obtained from the archives of the Cornell–IBM Joint Study on Computing for Scientific Research project. The statistical study was conducted using the SAS software.

VIII. CONCLUSION

In this paper we proposed four measures for image organizational change which can be used to monitor site change. The measures are based on the change in the relationship among image features. These relationships are captured by the eigenvalues and eigenvectors of the relation graph embodying the organization among the image features. We demonstrated the ability of the measures to distinguish between drastic and coarsely incremental change at a site using a Bayesian classifier. We also found the eigenvalue approach to be stable with respect to edge detector scales and algorithm parameters. The performance of the eigenvalue approach was also shown to be superior to simple statistical approaches.

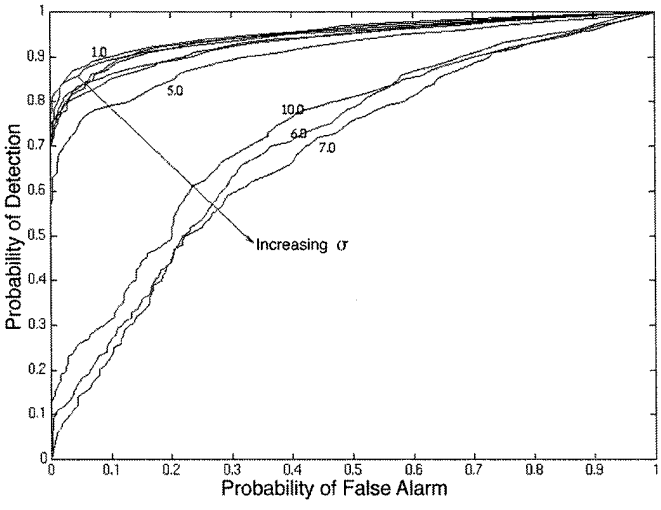

FIG. 21 Receiver operating characteristics of the proposed eigenvalue-based approach for different values of the edge detection smoothing parameter, σ. The vertical axis is for the true detection rate and the horizontal axis represents the false alarm rate.

REFERENCES

1. A. H. Tewfik and M. Deriche, An eigenstructure approach to edge detection, *Trans. Image Process.* **2**, 1993, 353–368.
2. D. B. Goldgof, H. Lee, and T. S. Huang, Matching and motion estimation of three dimensional point and line sets using eigenstructure without correspondences, *Pattern Recognition* **25**, 1992, 271–286.
3. M. A. Turk and A. P. Pentland, Face recognition using eigenfaces, in *Proceedings of the Conference on Computer Vision and Pattern Recognition, 1991*, pp. 586–591.
4. T. Lei and W. Sewchand, Eigenstructure approach to region detection and segmentation, *Int. Conf. Image Process.* **3**, 1994, 456–459.
5. L. S. Shapiro and J. M. Brady, Feature-based correspondence: An eigenvector approach, *Image Vision Comput.* **10**, 1992, 283–288.
6. K. Sengupta and K. L. Boyer, Organizing large structural modelbases, *IEEE Trans. Pattern Anal. Mach. Intelligence*, **17**, 1995, 321–332.
7. K. Sengupta and K. L. Boyer, Creating random structural descriptions of CAD models and determining object classes, in *Proceedings of the Second CAD based Vision Workshop, 1994*, pp. 38–45.
8. C. Huang, J. Mundy, and C. Rothwell, Model supported exploitation: Quick look, detection and counting, and change detection, in *Proceedings of the Second IEEE Workshop on Applications of Computer Vision, 1994*, pp. 144–151.

9. M. Benjamin, A. Huertas, G. Medioni, and R. Nevatia, Model validation for change detection, in *Proceeding of the Second IEEE Workshop on Application of Computer Vision, 1994*, pp. 160–167.
10. R. Lillestrand, Techniques for change detection, *IEEE Trans. Comput.* **21**, *1972*, 654–659.
11. M. S. Ulstad, An algorithm for estimating small scale differences between two digital images, *Pattern Recognition* **3**, 1973, 323–333.
12. H. S. D. M. Cvetkovic and M. Doob, *Spectra of Graphs*, Academic Press, New York, 1979.
13. R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridg, UK, 1990.
14. S. Sarkar and K. L. Boyer, A computational structure for preattentive perceptual organization: Graphical enumeration and voting methods, *IEEE Trans. Systems Man Cybernet.* **24**, Feb. 1994, 246–267.
15. R. Mohan and R. Nevatia, Perceptual organization for scene segmentation and description, *IEEE Trans. Pattern. Anal. Intell.* **14**, June 1992, 616–635.
16. S. Umeyama, An eigendecomposition approach to weighted graph matching problems, *IEEE Trans. Pattern Anal. Mach. Intelligence* **10**(5), 1988, 695–703.
17. R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.
18. M. Stone, Cross-validatory choice and assessment of statistical predictions, *Proc. Roy. Statist. Soc.* **B36**, 1974.