Statistics and Computing (2000) **10,** 167–182

3D object recognition: Representation and matching

ANIL K. JAIN* and CHITRA DORAI†

**Department of Computer Science, Michigan State University, East Lansing, Michigan 48824*
jain@cps.msu.edu
†*IBM T.J. Watson Research Center, P.O. Box 704, Yorktown Heights, New York 10598*
dorai@waston.ibm.com

Received January 1998 and accepted March 1999

Three-dimensional object recognition entails a number of fundamental problems in computer vision: representation of a 3D object, identification of the object from its image, estimation of its position and orientation, and registration of multiple views of the object for automatic model construction. This paper surveys three of those topics, namely representation, matching, and pose estimation. It also presents an overview of the free-form surface matching problem, and describes COSMOS, our framework for representing and recognizing free-form objects. The COSMOS system recognizes arbitrarily curved 3D rigid objects from a single view using dense surface data. We present both the theoretical aspects and the experimental results of a prototype recognition system based on COSMOS.

Keywords: three-dimensional objects, representation, recognition, localization, registration, view integration, automatic 3D object modeling, free-form objects, sculpted surfaces, intensity data, range images, digital interferometry

1. Introduction

The growing importance of computer vision is evident from the fact that it was identified as one of the "Grand Challenges" (Executive Office of the President, Office of Science and Technology Policy 1989) and also from its prominent role in the National Information Infrastructure (Weld 1995). The central goal of computer vision is to build a system that can automatically *interpret* a *scene*, given a snapshot (image) of the scene in terms of an array of brightness or depth values. A *scene* is defined as an instance of a 3D world consisting of one or more 3D objects. An *interpretation* of an image of a scene is defined as a determination of *which* 3D objects are *where* in the scene. Deriving an interpretation of a scene involves solving two interrelated problems. The first is *object identification*, in which a label must be assigned to an object in the scene, indicating the category to which it belongs. The second problem involves the *estimation* of the *pose* (position and orientation) or *localization* of the recognized object with respect to some global coordinate system attached to the scene. The term "object recognition" is used in computer vision to describe the entire process of automatic identification and localization of objects from the sensed images of scenes in the real world.

Three-dimensional object recognition is a topic of active interest motivated by a desire to build computers with "human-like" visual capabilities, and also by a pragmatic need to aid numerous real world applications such as robot bin-picking, autonomous navigation, automated visual inspection and parts-assembly tasks. The dominant paradigm in 3D object recognition (Roberts 1965) processes the sensed data in two stages: First, an internal representation of the scene is derived from the input image data that may have been obtained from one or more sensors such as CCD cameras and range scanners. In the second stage, the derived representation is matched against stored representations or models of known objects. Automating this *model-based* scene analysis is indeed an ambitious goal since a recognition system has to make sense out of a large number of pixels of an image array which by themselves contain very little information. Some knowledge of 3D object structures and how they appear in 2D images is necessary to make any assertion about the scene. *Object modeling* is the process that attempts to make this knowledge explicit. For example, objects can be modeled as volumes, or sets of bounding surfaces, or represented just as lists of salient local features. The two-dimensional image formation process can be characterized using either perspective or orthographic projection of a 3D scene into a 2D image array.

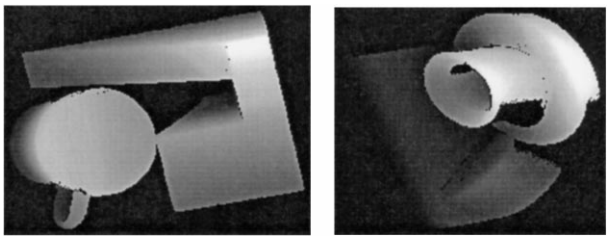

Fig. 1. *Range images containing occluded objects*

The resulting image has to be then processed to extract connected entities or "blobs" and to represent them in a way that is indicative of the 3D objects and their spatial interrelations in the scene. This process is confounded by practical difficulties such as sensor inaccuracies, variations in ambient illumination, shadows and highlights, background clutter, and occlusion due to objects being close to or sometimes overlapping one another in the scene, as shown in Fig. 1.

Recognition is the processing step that compares a derived description from the image which is often incomplete with stored representations of objects in order to identify what is present in the scene. A recognition module has to search among the possible candidate object representations to identify the best match and then determine whether the object label assigned to the entity in the scene is indeed *correct* or *incorrect* through a pose estimation step. This search procedure can be very time-consuming because the space containing possible feature-correspondences and view-transformations is very large. The time complexity of object matching critically depends on the number of stored objects, the degree of complexity and details of the stored representations, and the organization of the representations. In addition, the matching process has to deal with missing information in the derived representation due to occlusion, and sometimes spurious additional information resulting from incorrect merging of connected "blobs". Figure 2 illustrates the interplay of the three important complexity parameters in different application domains: (i) number of distinct objects possible in the scene; (ii) scene complexity; and (iii) object complexity. A typical machine inspection system is designed to operate under a controlled environment and under reasonable assumptions, e.g., that (a) an image of a part that needs to be inspected contains a single unoccluded view of the part and (b) the number of parts to be inspected are in the range of a few thousand per hour. On the other hand, a general object recognition system is expected to handle tens of thousands of objects of various classes, and the images of objects may contain views of multiple objects occluding one another. In practice however, a recognition system, rather than operating at the far extreme on most or all of these three dimensions, attempts to exploit the domain characteristics and performance requirements of a given application as much as possible in order to be successful.

Figure 3 presents the important stages in the design and development of a recognition system. All the processing steps in an automatic recognition system have to be performed reliably and efficiently in order to be employed in many challenging real-world applications such as automated part inspection for defects, face recognition for security analysis, and autonomous navigation of robots. Reconstruction and recognition of various aspects of shape and other physical properties of the objects in the real world are some of the fundamental steps during the creation of vision-augmented virtual environments (Kalawsky 1993, Breen *et al.* 1996). Medical diagnosis from X-ray, ultrasound, and MRI data (Sallam and Bowyer 1994) and image-guided surgical planning (Grimson *et al.* 1994) are examples of emerging applications in which automatic object recognition can make a significant impact and be beneficial to humanity.

2. Challenges in 3D object recognition

A number of representational and matching themes have been recognized by several researchers as challenging and important in object recognition (Flynn and Jain 1994). These themes are related to the complexity, speed, and generality issues that need to be addressed by a recognition system.

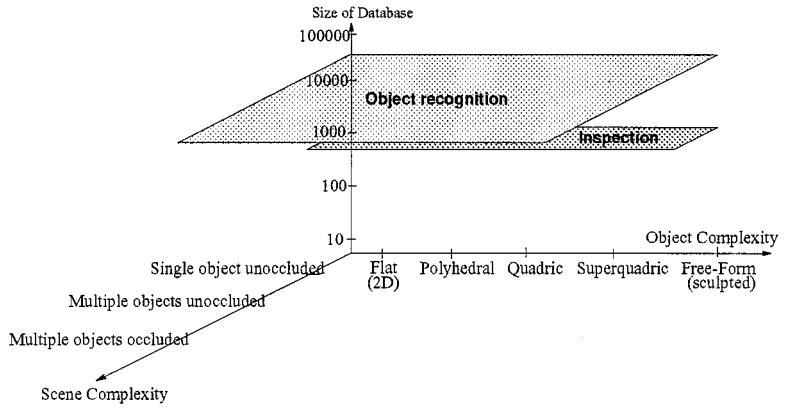

Fig. 2. *Complexity of a recognition system. A general purpose object recognition system differs from a specific machine inspection system in terms of the number of objects to be handled, scene complexity, and object shape complexity*

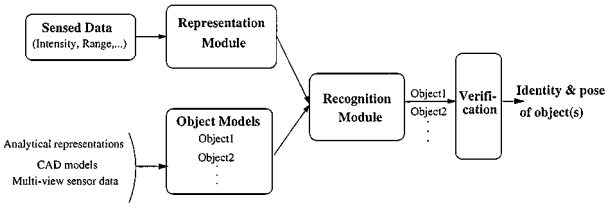

Fig. 3. *Key components of a 3D object recognition system*

- *Object shape complexity*: 3D object recognition has so far dealt mainly with geometric entities in two and three dimensions, such as points or groups of points, planar patches and normals, straight edges and polylines, polyhedral and quadric surfaces, and superquadrics. The success of several existing object recognition systems can be attributed to the restrictions they impose on the classes of geometrical objects that can be handled. However, there has been a notable lack of systems that can handle arbitrary surfaces with very few restrictive assumptions about their geometric shapes. Interest has recently emerged in matching arbitrarily curved surfaces that cannot be modeled using volumetric primitives and that may or may not have easily detectable landmark features such as vertices and edges of polyhedra, vertices of cones and centers of spheres. Complex real-world applications of computer vision systems such as content-based search of image and video databases, digital reconstruction of artistic masterpieces and works of architecture, and object-based coding for interactive video have recently begun to stimulate the development of general 3D recognition systems that can handle arbitrarily curved objects.
- *Size of object model database*: In real-world applications, one typically finds the need to handle databases containing a large number of complex object models. By "large" we mean typically a thousand or more models. As the number of objects to be recognized by a system increases, the computational time to perform recognition of even a simple input image becomes discouragingly large. This is primarily due to the fact that in most systems, the input representation is matched against all the object models in the database. There is an increased awareness of this computational cost owing to recent interest in search and retrieval tasks related to image and video data on the Internet and corporate intranets. It has resulted in methods to prune the number of matches needed either by using focus or salient object features or by indexing a hash table based on invariant features during recognition. In addition, approaches that can organize the representations in a hierarchical fashion (to eliminate unlikely matches quite early during recognition and subsequently present only a few candidate objects for final verification of their identity and pose) have begun to be considered seriously.
- *Learning*: As the requirement for adaptation and flexibility in vision systems grows, there is an increasing need for systems that incorporate at least some aspect of learning. A recognition system may perform well within the scope of its knowledge; but any slight deviation such as noisy image segmentation or representation outside the narrow expertise of the system causes the performance to deteriorate rapidly. Many application domains of computer vision systems are by and large unstructured. Less controlled environments, therefore, mandate construction of systems that can automatically build models of "unexpected" objects. It is desirable to have the recognition system learn the description of an unknown object so that future encounters with instances of the object will not result in system breakdowns (Poggio and Edelman 1990). The ability to learn to recognize new inputs as well as to remember the previous instances of objects is known as the adaptation or *plasticity* property. This property is lacking in current recognition systems which are thus rendered quite brittle. It is also vital to be able to learn generalized representations of an object from multiple training instances in order to recognize a degraded or noisy instance of the object in an image. Further, learning enforces the use of a large variety of real data in performance evaluation of recognition systems, due to the need for feedback that is necessary for valid generalizations.
- *Individual and generic object categories*: A recognition system can represent and recognize either individual objects or store and match descriptions of generic classes of objects. For example, descriptions can be stored for individual chairs and/or the generic class of chairs. Recognition of the latter category is a more difficult problem since there is no unique structural description that characterizes the entire class of chairs although a functional description of the class of chairs is available and simple. Functionality is tied to the description of geometric features that need to be present in an object for it to be recognized as an instance of a generic category. Although function-based representations are appropriate for constructing "generic" models of some classes of objects, for other objects, their function cannot be readily translated into their appearance. This has resulted in a push for a hierarchy of description levels with a slew of models that allow for symbolic reasoning about object form and function at the coarse level and for more detailed geometry based reasoning such as graphs or interpretation trees at the level of object instances.
- *Non-rigidity of objects*: Most object recognition systems assume that the objects under consideration are rigid. Flexible objects such as organs (for example, heart and lung) in a human body are those whose shape need not remain constant with time. A deformable object is one which is entirely non-rigid while an articulated object has movable rigid parts. Recent work has begun to investigate whether there exist primitives of fixed volume or of fixed surface shape at suitable scales to describe non-rigid objects. A family of shapes for a single object can be specified by parameterizing the point patterns representing the shape (Umeyama 1993). Applications in medical imaging have spurred some of the research in deformable shape representations such as deformable superquadrics and finite-element methods.
- *Occlusion*: Another major issue in the design of a recognition system is how to reliably recognize partial data of objects in the scene. Recognition systems that deal with multiple

objects present within the field of view of a sensor need to be able to recognize objects that may be partly occluded by others (Turney, Mudge and Volz 1985). Self occlusion is also possible with objects that are complex-shaped. The stored object representation should be such that recognition of objects is possible even from partial information. The presence of "salient" local features in object representations would certainly be a crucial and deciding criterion in recognizing the object. Even if an object is partially seen, a total lack of distinguishing features may render its recognition impossible.

- *Viewpoint-dependency*: Approaches to 3D object representation can be categorized as either *viewpoint-independent* (object-centered) or *viewpoint-dependent* (viewer-centered). A viewpoint-independent representation attaches a coordinate system to an object; all points or object features are specified with respect to this system. The description of the object thus remains canonical, independent of the vantage point. Although, this approach has been favored by Marr and Nishihara (1978) and others, it is difficult to derive an object-centered representation from an input image. A unique coordinate system needs to be first identified from the input images and this becomes difficult when the object has many natural axes. Practical implementation becomes a complicated task as only 2D or 2.5D information is usually available from a single image and perspective projection effects have to be corrected in the image, before building the representation. Note that this approach is well suited to simple 3D objects that can be specified by analytic functions. A viewer-centered approach, on the other hand, describes an object relative to the viewer and as one does not have to compensate for the viewpoint, object view representations can be easily computed from images. A major disadvantage is that a large number of views needs to be stored for each object, since different views of an object are in essence treated as containing distinct objects. However, representing an object with multiple views is quite useful in view-based matching, alleviating the need for expensive 3D model construction. An important research issue in view-based recognition is: which and how many of these object views are actually necessary and useful for *recognition*?

3. Object representations and recognition strategies

Research in three-dimensional object recognition addresses a number of significant issues related to the themes just described. This section presents a brief survey of the previous work in three important topics: representation, matching, and pose estimation of a 3D object. Besl and Jain (1985), Chin and Dyer (1986), Suetens, Fua and Hanson (1992), and Arman and Aggarwal (1993) also present comprehensive surveys of 3D object recognition systems. The spectrum of 3D object recognition problems is discussed in Bajcsy and Solina (1987), Connell and Brady (1987), and Jain and Flynn (1993). Sinha and Jain (1994) also provide an overview of geometry-based representations derived from range data of objects. Model-based 3D object recognition systems differ in terms of a number of factors, namely: (i) the type of sensors used, (ii) the kinds of features extracted from an image, (iii) the class of objects that can be handled, (iv) the approaches employed to hypothesize possible matches from the image to object models, (v) the conditions for ascertaining the correctness of hypothesized matches, and (vi) the techniques to estimate the pose of the object. We discuss these various design issues in our descriptions of the popular schemes prevalent for representation and recognition.

3.1. *Sensors*

Among the various types of image sensors used for 3D object recognition, the two most commonly used are: (a) *intensity* sensors that produce an image (a 2D array) containing brightness measurement $I(x, y)$ at each pixel location (x, y) of the image, and (b) *range* sensors which provide the range or the distance $z(x, y)$ of a visible surface point (pixel) on the object from the sensor. The brightness measurement of a scene at a location in an image is a function of surface geometry, reflectance properties of the surfaces, and the number and positions of the light sources that illuminate the scene. Range sensors are calibrated to result in images with spatial coordinates that are directly comparable with the coordinates used in object representations. A big advantage with range data over intensity data is that it explicitly represents surface information. This makes it easier to extract and fit mathematical surfaces to the data. Range data is also usually insensitive to ambient lighting. Regions of interest belonging to objects can be separated from the background easily in range images based on the distinctive background depth values provided by the sensor, as opposed to intensity images which can have complex backgrounds with varying grey-levels that are similar to those of the objects themselves. Surface depth of objects can be measured using a number of techniques such as time-of-flight lasers, structured light, sonar, and radar. Algorithms from computer vision to determine object depth from shading, texture, stereo, motion, and focus are also useful to acquire surface data. A taxonomy and detailed descriptions of range sensing methods are found in Jarvis (1993). Figure 4(a) shows a range image of a view of a wooden Buddha sculpture where the depth is rendered as pseudo intensity displaying the relative orientation of the surfaces; points oriented almost vertically are shown in darker shades. The corresponding three-dimensional surface plot is shown in Fig. 4(b). Full-field optical surface profilers such as projected fringe interferometers are also employed when high accuracy is needed in surface height measurements (Mercer and Beheim 1990). A recent surge of interest in medical applications has also motivated the use of magnetic resonance imaging (MRI) and other types of 3D data obtained through medical imaging modalities.

3.2. *3D object representations and models*

Table 1 presents an overview of some of the key representation schemes and applicable object domains. Design of an

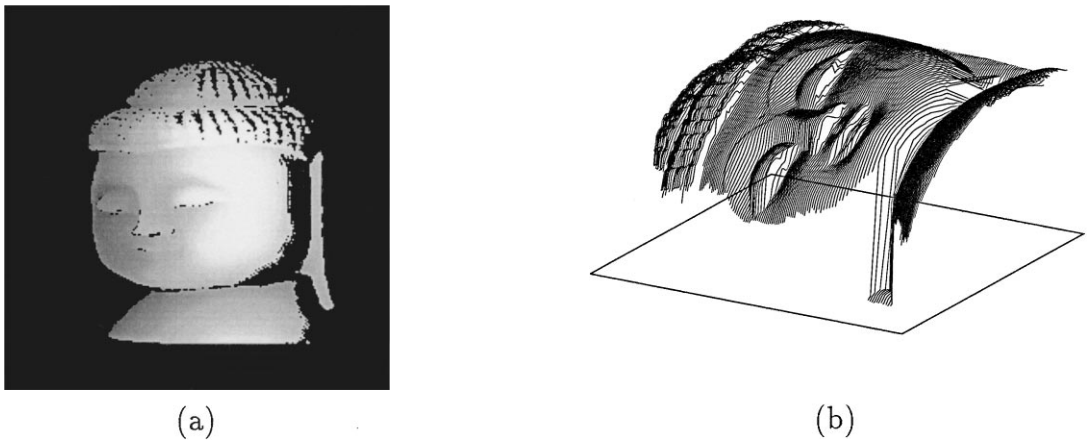

Fig. 4. *View of a wooden Buddha sculpture: (a) A range image of the object where the depth is rendered as pseudo intensity; (b) its three-dimensional surface plot*

appropriate representation scheme for 3D objects is a crucial factor that influences the ease with which they are recognized. Most of the successful object representation schemes have adopted some form of surface or volumetric parametric models to characterize the shapes of objects. Current volumetric representations rely on representing objects in terms of spatial occupancy, generalized cylinders, superquadrics, or set-theoretic combinations of volume primitives as in constructive solid geometry (CSG) (Samet 1990, Brooks 1983, Pentland 1986, Solina and Bajcsy 1990, Chen and Lin 1994). However, objects with free-form surfaces, in general, may not have simple volumetric shapes that can be easily expressed with, for example, a superquadric primitive, even though it may contain eight (including bending and tapering) parameters. Further, the difficulty with recognizing an object via matching volumetric descriptions is that many views of the object must be used because there is uncertainty in the extent of the object in a direction parallel to the line of view. However, humans can identify objects even from a partial view. This aspect is a motivating factor for matching objects using "surface-based" representations that describe an object in terms of the surfaces bounding the object and their properties such as the surface normals, curvatures, etc. (Bolles and Horaud 1986, Grimson and Lozano-Pérez 1987, Fan, Medioni and Nevatia 1989, Faugeras and Hebert 1986, Flynn and Jain 1991, Jain and Hoffman 1988, Besl 1988). Surface representations are more commonly employed for recognition since they directly correspond to features easily derived from the sensed image of the scene. In addition, matching only the observed surface patches with those in a stored representation can aid in recognition of occluded objects.

Surface representations are mostly based on a small set of analytical surface primitives that either exclude sculpted objects from their domains, or allow free-form surfaces at the expense of approximating the object with a very large number of simple primitives such as quadric and bicubic surface patches. Such an approximation tends to be coarse or fine depending on the number of primitives used. If it is coarse, then it may not capture the shape of the object accurately and hence can be ambiguous. If it is too fine, the number of primitives will be too large, leading to a loss of global shape. Therefore, global representations such as the extended Gaussian image (EGI) (Horn 1984) and other orientation-based descriptors (Nalwa 1989, Kang and Ikeuchi 1993, Liang and Taubes 1994, Matsuo and Iwata 1994) describe 3D objects in terms of their surface-normal distributions on the unit sphere, with appropriate support functions. They handle convex polyhedra efficiently, but arbitrarily curved objects have to be either approximated by planar patches or divided into regions based on the Gaussian curvature. Part-based representations (Biederman 1987, Raja and Jain 1994, Dickinson, Pentland and Rosenfeld 1992) have also become important as they capture structure in object descriptions but there is a lack of consensus in deciding the general set of part primitives that need to be used and in justifying why they are necessary, sufficient, and appropriate. In addition, computation of parts from a single view of an object is difficult.

If a 3D object can be specified using a few parametric equations that capture its volumetric or surface characteristics, then an object-centered and viewpoint-independent representation serves as an efficient description. However, in describing complex objects whose shapes cannot be captured by a single analytical form, or by a set of equations compactly, viewer-centered representations play a more important role. The aspect graph approach (Koenderink and van Doorn 1979) attempts to group what could possibly be a set of infinite 2D views of a 3D object into a set of meaningful clusters of appearances. Some view-sensitive appearance-based descriptions (Murase and Nayar 1995, Swets 1996) also exploit photometric information in describing and recognizing objects, unlike geometric systems. A major drawback of view-centered representations is a lack of terseness. Their ability to generalize from object instance descriptions to classes remains to be demonstrated.

With recent interest in building content-based image retrieval systems, there is a greater focus on approaches that learn broad

Table 1. *An overview of popular object representation schemes*

Representation scheme	Type of shape descriptor	Object domain	Sensing modality	Viewpoint dependency
Points (corners and inflection points along edge contours) (Huttenlocher and Ullman 1990)	Local	Objects with well-defined local features	Intensity	Stable over changes in viewpoint
Straight line segments (Lowe 1987)	Local	Polyhedra	Intensity	Viewpoint-invariant over wide ranges
Points, planar faces and edges (Grimson and Lozano-Pérez 1987)	Local	Polyhedra	Range	Viewpoint-independent
Silhouettes of 2D views (Basri and Ullman 1988, Ullman and Basri 1991, Chen and Stockman 1996)	Global	Curved	Intensity	Viewpoint-dependent
Circular arcs, straight edges, cylindrical and planar surfaces (Bolles and Horaud 1986)	Local	Planes and cylinders	Range	Viewpoint-independent
Planar and quadric surface patches (Fan, Medioni and Nevatia 1989, Faugeras and Hebert 1986, Flynn and Jain 1991)	Local	Planes, quadric surfaces	Range	Viewpoint-independent
Convex, concave and planar surfaces (Jain and Hoffman 1988)	Local	Arbitrarily curved objects	Range	Object-centered
Gaussian and mean curvatures-based patches (Besl 1988)	Local	Curved	Range	Viewpoint-invariant
Generalized cylinders (GC) (Brooks 1983)	Global	Generalized cylinders	Intensity	Object-centered
Superquadrics (Pentland 1986, Solina and Bajcsy 1990)	Global	Curved objects	Range	Object-centered
Constructive surface geometry (simple volumetric primitives) (Chen and Lin 1994)	Local	Curved objects	Range	Object-centered
Geons (Raja and Jain 1994, Dickinson, Pentland and Rosenfeld 1992)	Parts-based	Curved including articulated objects	Range	Object-centered
Extended Gaussian image (EGI) (Horn 1984)	Global	Convex objects	Intensity	Object-centered
Algebraic polynomials (Taubin *et al*. 1992)	Global	Curved	Range	Object-centered
Splash and super (polygonal) segments (Stein and Medioni 1992)	Local	Arbitrarily curved	Range	Viewpoint-independent
Aspect graphs (Koenderink and van Doorn 1979, Plantinga and Dyer 1990, Eggert and Bowyer 1993)	Global	Convex polyhedra and a class of curved-surfaces	Intensity	Viewpoint-sensitive
Eigen faces (Murase and Nayar 1995, Swets 1996)	Global	General 3D objects	Intensity	Viewpoint-dependent (appearance based)

object categories without the help of prespecified models. This interest has spurred research in more general shape representation schemes that attempt to capture a variety of details about objects: viewing an object as a composition of superquadrics and geons, thus utilizing both qualitative and quantitative features to achieve recognition of object classes (Borges and Fisher 1997); representing articulatedness of an object as a parameterization of relative movement between the parts that comprise the objects; modeling non-rigid objects that change shape with time using deformable superquadrics (Terzopoulos and Metaxas 1991), finite element analysis (Horowitz and Pentland 1991), and statistical shape models such as point distribution models (Lanitis, Taylor and Cootes 1997) and their extensions using polar coordinates (Heap and Hogg 1995); and employing function (utility)-based attributes to describe a generic category of objects (Stark and Bowyer 1991).

In summary, a search for a uniform representation scheme that can handle all object categories and instances is unlikely to succeed. Object representations are most effective when they are task-dependent and when they encapsulate a range of descriptions, of varying type and level of detail. Characterizing an object with a set of many models rather than with a single prototype description is useful for dealing with shape variations, positional uncertainty, and occlusion; endows a recognition system with a generalization capability; and leads to organization of learned object categories.

3.3. *Matching strategies*

Object recognition is achieved by matching features derived from the scene with stored object model representations. Each successful match of a scene feature to a model feature imposes a constraint on the matches of other features and their locations in the scene. A consistent set of matches is referred to as a *consistent scene interpretation*. Approaches vary in terms of how the match between the scene and model features is achieved, how a consistent interpretation is derived from the scene-model feature matches and how the pose is estimated from a consistent interpretation. In the following discussion, "scene" and "image" are used interchangeably and so are "model" and "object model". A "model" indicates a stored representation. The popular and important approaches to recognition and localization of 3D objects are the following: (i) hypothesize-and-test, (ii) matching relational structures, (iii) Hough (pose) clustering, (iv) geometric hashing, (v) interpretation tree (I.T.) search, and (vi) iterative model fitting techniques.

In the **hypothesize-and-test** paradigm, a 3D transformation from the object model coordinate frame of reference to the scene coordinate frame of reference is first hypothesized to relate the model features with the scene features. A system of equations is solved to provide the transformation that minimizes the squared error which characterizes the quality of match between the model and scene features. The transformation is used to verify the match of model features to image features, by aligning the model features (typically points and edge segments) to the scene features. The hypothesized match is either accepted or rejected depending on the amount of matching error. Lowe (1987), Huttenlocher and Ullman (1990), Seales and Dyer (1992), and Chen and Stockman (1996) have presented representative work in this paradigm. Other examples include earlier recognition systems such as 3DPO (Bolles and Horaud 1986) and RANSAC (Fischler and Bolles 1981).

Representations using **relational structures** attempt to capture the structural properties of objects by describing both scene and object models using attributed-relational graphs (ARGs), where each node in the ARG stands for a primitive scene or model feature and the arc between a pair of nodes represents a relation between the two features. Matching of a scene ARG with a model ARG is carried out using graph-theoretic matching techniques such as maximal clique detection, sub-graph isomorphism, etc. (Barrow and Burstall 1976). Recognition schemes using relational graphs have been explored extensively (Brooks 1983, Vemuri and Aggarwal 1987, Fan, Medioni and Nevatia 1989, Wong, Lu and Rioux 1989).

In the **pose clustering** approach, also referred to as the generalized Hough transform, evidence is first collected for possible transformations from image-model matches and then clustered in the transformation space to select a pose hypothesis with the strongest support. Each scene feature is matched with each possible model feature; matches are then eliminated based on local geometric constraints such as angle and distance measurements. A geometric transformation is computed from each successful match and stored as a point in the Hough (transformation parameter) space. The Hough space is six-dimensional if we deal with 3D objects with six degrees of freedom whereas it is three-dimensional for 2D planar objects with three degrees of freedom. Clustering of points in the Hough space results in a globally consistent pose hypothesis of the object present in the scene. Stockman (1987), Krishnapuram and Casasent (1989), and Silberberg, Davis and Harwood (1984) are representative of recognition using pose clustering.

In **geometric hashing**, also known as structural indexing, feature correspondence determination and model database search are replaced by a table look-up mechanism. Invariant features are computed from an image that can be used as indices into a table containing references to the object models. The pioneering work by Lamdan and Wolfson (1988, 1990) uses a two-stage methodology: (i) creation of a model hash table using invariants computed from model features and (ii) indexing into the table using the image invariants. The model entry that has the maximum support is used to compute a rigid transformation from the model to the scene coordinate system. Stein and Medioni (1992) and Flynn and Jain (1992) have employed geometric hashing for 3D object recognition.

Interpretation tree (I.T.) search, or constrained search, is a very popular recognition scheme and has been the subject of active work over the past ten years. An I.T. consists of nodes that represent a potential match between a scene feature and a model feature. During search, a scene feature is paired with a model feature and a node at level n of the tree characterizes a partial interpretation, i.e., the path from the root to a node at level n specifies an assignment of model features to the first n scene features. Instead of searching the tree exponentially for a complete and consistent interpretation, local geometric constraints such as pairwise angle and distance measurements between features are used to discard or prune inconsistent matches between scene features and model features. A global transformation is computed to determine and verify the pose of the object when a path of sufficient length is found. I.T. search was formulated and well explored by Grimson (1984, 1987). Chen and Kak (1989), Flynn and Jain (1991), Ikeuchi (1991), and Vayda and Kak (1991) have also employed constrained search of the interpretation tree for CAD-based object recognition.

Iterative model fitting is used when 3D objects are represented by parametric representations wherein the parameters are used to specify both the shape and the pose of the objects. There is no feature detection and correspondence determination between model and scene features. Object recognition and pose estimation reduce to estimating the (pose) parameters of the model from the image data, and matching with stored parametric representations. If a sufficient number of image data points is available, the estimation of parameters can be done by solving a system of over-constrained linear or nonlinear equations for a solution that is best in the minimum-sum-of-squared-error sense. Solina and Bajcy (1990), Gupta, Bogoni and Bajcsy (1989),

Gupta and Bajcsy (1992) and Pentland (1990) have modeled 3D objects as superquadrics with local and global deformations for recognition purposes. Implicit equations (Ponce, Hoogs and Kreigman 1992) have also been used to fit observed image features to the 3D position and orientation of object models.

In addition to these popular approaches which can be used to classify a majority of the existing matching schemes, there are a few others that merit mention. One of them is a **rule-based** approach proposed by Jain and Hoffman (1988) to recognize 3D objects based on evidence accumulation. Instead of matching object features to scene (image) features, they construct an evidence rule base that stores salient information about surfaces and use it to compare the similarity between scene and model features. Another approach views matching as a **registration** problem (Besl and McKay 1992) and matches sets of surface data with one another directly without any appropriate surface fitting. In this approach, the distance between two point-sets obtained from surfaces is computed and minimized to find the best transformation between the model and scene data.

In summary, matching strategies such as the Hough clustering compare global features or shapes, and are relatively fast. However, they are error-prone when there is occlusion in images. Local feature-based matching schemes can handle occlusion but tend to be computationally expensive. Recognition systems have to be made more robust and faster (e.g., by parallelizing segmentation and matching algorithms) in order to handle large databases of complex objects.

4. 3D free-form object recognition

The success of many of the object recognition systems can be attributed to the restrictions they impose on the geometry of objects. However, there has been a notable lack of systems that can handle objects with arbitrarily curved surfaces. These complex free-form surfaces may not be modeled easily using volumetric primitives and may not have easily detectable landmark (salient) features such as edges, vertices of polyhedra, vertices of cones and centers of spheres. A free-form surface S is defined to be a smooth surface such that the surface normal is well defined and continuous almost everywhere, except at vertices, edges, and cusps (Besl 1990). Since there is no other restriction on S, it is not constrained to be polyhedral, piecewise-quadric or superquadric. Discontinuities in the surface normal or curvature may be present anywhere on a free-form object and similarly, discontinuities in the surface depth may be present anywhere in a projection of the object. The curves that connect these points of discontinuity may meet or diverge smoothly. Some representative objects with free-form surfaces are human faces, cars, boats, airplanes, sculptures, etc. Free-form surfaces are extensively used in the design of smooth objects in automotive, aerospace, and ship building industries. Recognition of free-form objects is essential in automated machining of complex parts, inspection of arbitrarily curved surfaces, and path planning for robot navigation.

Figure 5 shows a set of range images of some 3D free-form objects, obtained using a laser range scanner that produces depth data in an X-Y grid. The figure shows surface depth as pseudo intensity.

Recent approaches using algebraic polynomials (Keren, Cooper and Subrahmonia 1994, Ponce *et al.* 1993) splash and super (polygonal) segments (Stein and Medioni 1992), simplex angle image (Delingette, Hebert and Ikeuchi 1993), 2D silhouettes with internal edges (Chen and Stockman 1996), and point sets based registration (Besl and McKay 1992) have specifically sought to address the issue of representing complex curved free-form surfaces. They suffer from one or more limitations relating to object segmentation issues, bounding constraints, surface fitting convergence, restricting objects to be topologically equivalent to a sphere, local minima, and sensitivity to noise when using low-level surface features.

We have developed a new approach to automated representation and recognition of 3D free-form rigid objects using dense surface data. Our computer vision system recognizes arbitrarily curved 3D rigid objects from a single view when (a) the viewpoint can be arbitrary and (b) the objects may vary in shape and complexity. Our surface representation scheme, COSMOS (Dorai and Jain 1997a) yields not only a meaningful and rich description useful for the recoverability of several classes of 3D objects, but also provides a set of powerful matching primitives for object recognition. We present a recognition strategy which consists of a multi-level matching mechanism employing shape spectral analysis and features derived from the COSMOS representations of objects for both fast and efficient object identification and pose estimation. Figure 6 shows the various modules that comprise our recognition system. All theoretical aspects of this work have been experimentally validated via a prototype system, which has been tested on a large database of range images of several different complex objects acquired using a laser range scanner.

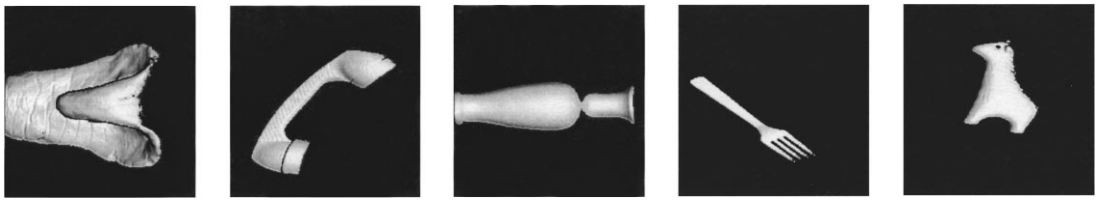

Fig. 5. *Range images of 3D free-form objects*

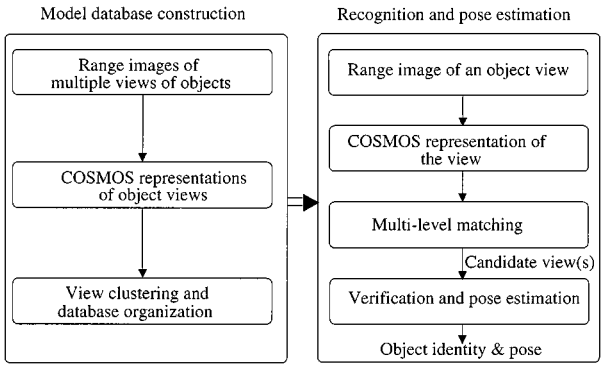

Fig. 6. *Overview of our 3D object recognition system (Dorai 1996)*

4.1. *The representation scheme*

COSMOS (*C*urvedness-*O*rientation-*S*hape *M*ap *O*n *S*phere) is a general scheme capable of representing free-form surfaces and objects with holes as it does not rely on analytical surface primitives for object modeling. The novelty of the scheme lies in its description of an object as a smooth composition or arrangement of regions of arbitrary shapes that can be detected regardless of the complexity of the object. Each of the local and global attributes used in the COSMOS scheme captures a specific geometric aspect of the object and is defined using differential geometry based concepts such as surface normals, curvature, shape index and curvedness. Since the data obtained from a single-view range imaging sensor typically takes the form of a graph surface, the surface parameterization assumes a simple form: $\vec{x}(u, v) = [u, v, f(u, v)]^T$ where T indicates the transpose. The COSMOS-based recognition system works with graph surfaces as well as with any collection of (x, y, z) points on which the fundamental notions of metric, tangent space, curvature and natural coordinate frames can be suitably defined.

4.1.1. COSMOS of a free-form object

We employ a modified definition of *shape index*, originally proposed by Koenderink for graphical visualization of surfaces (Koenderink and van Doorn 1992), to identify the shape category to which each surface point on an object belongs. Our approach makes use of the shape index, S_I which is a scalar ranging from [0, 1] computed from the principal curvatures of a surface to describe complex objects. Every distinct surface shape corresponds to a unique value of S_I, excepting the planar shape. For computational purposes in our implementation, a symbolic label (or sometimes, a shape index value of 2.0) is used to indicate surface planarity. Since Gaussian curvature is intrinsic to a surface, bending a surface without stretching preserves the Gaussian curvature, although the "shape" is modified by this action. Therefore, *both* Gaussian and mean curvatures are necessary for characterizing the notion of "extrinsic shape" (Koenderink and van Doorn 1992). On the other hand, a single shape index suffices for the same task.

Fundamentally, COSMOS concisely describes an object in terms of a set of maximally sized surface patches of constant shape index. For example, a spherical surface has a single constant shape maximal patch (CSMP) of spherical cap (convex) shape, whereas a truncated cylinder bounded by hemispherical caps at its ends has three CSMPs, one with cylindrical ridge shape and the other two of spherical cap shape. The COSMOS representation of an object segmented into a set of homogeneous patches (CSMPs) is comprised of two sets of functions: the Gauss patch map and the surface connectivity list $\langle G_0, V \rangle$, and two support functions $\langle G_1, G_2 \rangle$ (Dorai 1996). Given an object O segmented into a set of CSMPs $\mathcal{P}_O$, the Gauss patch map G_0 maps each CSMP, P, on O to a point on the unit sphere whose normal corresponds to the *orientation* (mean surface normal) of the patch P. This spherical mapping of the maximal patches thus results in a description of the object's orientation in 3D space. The surface connectivity list V associates each patch $P \in \mathcal{P}_O$ with the set of patches $V(P) = \{Q\} \subseteq \mathcal{P}_O$ that are adjacent to P, and thus represents connectivity information about the segmented object. It can be seen that the traditional *region adjacency graph* data structure can be easily derived from the set of CSMPs, $\mathcal{P}_O$, and the surface connectivity list V of an object, where each CSMP, P, on the object serves as a node in the graph and $V(P)$ provides information about the edges (or the connectivity) that link the nodes in the region adjacency graph.

The support functions $\langle G_1, G_2 \rangle$, defined on the unit sphere $\mathcal{S}^2$, summarize at each point on the sphere, local geometric information such as the area and the curvedness of all the patches that have been mapped by G_0 to the point. The average curvedness of a surface patch specifies whether it is *highly* or *gently* curved; the surface area quantifies its extent or spread in 3D space. G_1 integrated over a region on $\mathcal{S}^2$ results in a summary of the surface areas of all the mapped CSMPs in each shape category. Similarly, G_2 when integrated over a region around a point on the unit sphere and normalized by the area of the mapped patches provides the mean curvedness of the patches mapped into the region. The definitions of both G_1 and G_2 make use of the notion of shape spectral function (Dorai and Jain 1997a) and a suitable transform of the support functions results in high level feature summaries of the object.

We have proposed a novel concept called the *shape spectrum* that can be derived from the support function G_1 of the object (Dorai 1996). The shape spectrum of an object characterizes its shape content by summarizing the areas of the surfaces on the object at each shape index value. It qualitatively describes "which shape categories are present and how much of each shape category is present in an object." As has been signaled by the term "spectrum" itself, we have shown that the shape spectrum is the Fourier transform of the integral of G_1 over the entire unit sphere (Dorai 1996). Unlike schemes based on histograms of surface area versus the curvatures, our shape spectrum has many advantages including the decoupling of size and shape so that a single number, S_I suffices to capture the latter (Koenderink and van Doorn 1992), and obviation of the need for the (arbitrary) assignment of a principal direction. The shape spectrum-based

features allow free-form object views to be grouped meaningfully in terms of the shape categories of the visible surfaces and their surface areas.

The main strength of COSMOS is the integration of local and global shape information that can be computed easily from sensed data and is reflective of the underlying surface geometry. The representation is compact for many classes of objects that contain only a few distinguishable surface patches of constant shape index, i.e., whose surface shapes do not change rapidly over large regions of the object. While shape spectra of objects can be used to distinguish between categories, supplemental information needed to perform more specific model-based object recognition is incorporated in COSMOS representations of objects using CSMPs and their attributes. We refer the reader to (Dorai and Jain 1997a) for a comprehensive discussion of the properties of COSMOS.

4.1.2. Deriving COSMOS from range data

The COSMOS representation of an object derived from its *complete* 3D surface data is viewpoint-independent. In practice, especially with a free-form surface, we may not have a complete object model or complete surface data. Hence, we employ a set of COSMOS representations derived from range images of multiple views to represent an object. We have designed and implemented a procedure (Dorai 1996) to compute the COSMOS representation from the range data from a single view of an object.

Range images of objects obtained using the laser range scanner available in our laboratory provide depth data of surfaces visible to the camera. Principal surface curvatures are computed at each pixel (object point) in an image by estimating them using a bicubic approximation of the local neighborhood. In our implementation, we use a neighborhood of 5×5 pixels to locally approximate the underlying surface. Then an iterative curvature smoothing algorithm based on the curvature consistency criterion (Ferrie, Mathur and Soucy 1993) is applied to improve the reliability of the estimated curvatures. The range data is processed by running the curvature smoothing algorithm for 15 iterations. Once the curvatures are reliably estimated at each pixel, the shape index values are computed. The next step is to obtain as few maximally connected patches of constant shape index as possible while retaining sufficient information about the different shapes that may be present. Since the range data are of finite resolution, and since curvature estimates can be noisy, we need to take into account the possibility of noise in the shape index values of surface points belonging to the same shape category. Connected points whose shape indices are similar have to be merged to obtain a CSMP.

Our segmentation algorithm groups image pixels with the following objectives: (i) minimize the total number of CSMPs in the image (to avoid fragmentation into hundreds of small patches); and (ii) minimize some measure of the spread of shape index values of the pixels within a CSMP. Since these two objectives obviously conflict, global information is required to achieve objective (i) subject to some constraints. Our region growing technique constructs maximal patches by repeatedly merging smaller patches, starting with very small (pixel-sized) patches. Since in each step it merges the two (adjacent) patches that would result in the least merged *shape diameter* (the difference between the minimum and maximum shape index values within the patch) of all the feasible (connected) pairs of patches, the spread of shape indices of the pixels within a patch is minimized. In addition, since the algorithm is applied until the maximum merged shape diameter would exceed the constrained value for every remaining pair of patches, it minimizes the total number of patches. Using the shape index in this way improves the segmentation over methods that use only fixed-width and fixed-threshold quantized bins.

In our experiments with different objects, a shape diameter of 0.25 yielded good CSMPs in most images. An increased shape diameter resulted in bigger patches in the cases of the complex objects such as the cobra-head. This parameter can be adaptively adjusted depending on the size of the smallest CSMP that is detected in a given image. Since our current segmentation algorithm does not explicitly detect edges prior to region growing, our experimental results show that the detected CSMPs blend into neighboring CSMPs without a sharp intervening boundary. Future work should integrate edge detection schemes with the region growing algorithm to obtain stable region boundaries and also prevent CSMPs from leaking across the discontinuities to their surrounding regions. Another promising direction is experimenting with statistical properties such as the standard deviation of a patch's S_I as a basis for determining the CSMPs instead of the shape diameter.

Once the CSMPs in the image are obtained, the Gauss patch map is constructed using their surface normals and the surface connectivity list is built based on their adjacency. The surface attributes, area and curvedness, stored by the coefficients in the support functions are computed at the mapped points on the unit sphere. The shape spectrum of the object view is obtained by constructing a histogram $H(h)$ of the shape index values – we used 0.005 as the bin width – and accumulating all the object pixels that fall into each bin. Figure 7 shows the components of the COSMOS representation of a view of a vase (referred to as Vase-2) derived from its range image.

4.2. *COSMOS based recognition*

In model-based object recognition, the identity of a 3D object in an input scene is determined by matching features derived from the sensed data of a scene against stored object model representations. We have proposed a novel multi-level matching strategy to address the following important issues: (i) What sort of indexing (model selection) mechanism should be used for identifying a subset of candidate object views in a model database that can then be matched in detail with the input view? (ii) What recognition strategy should be used to match the

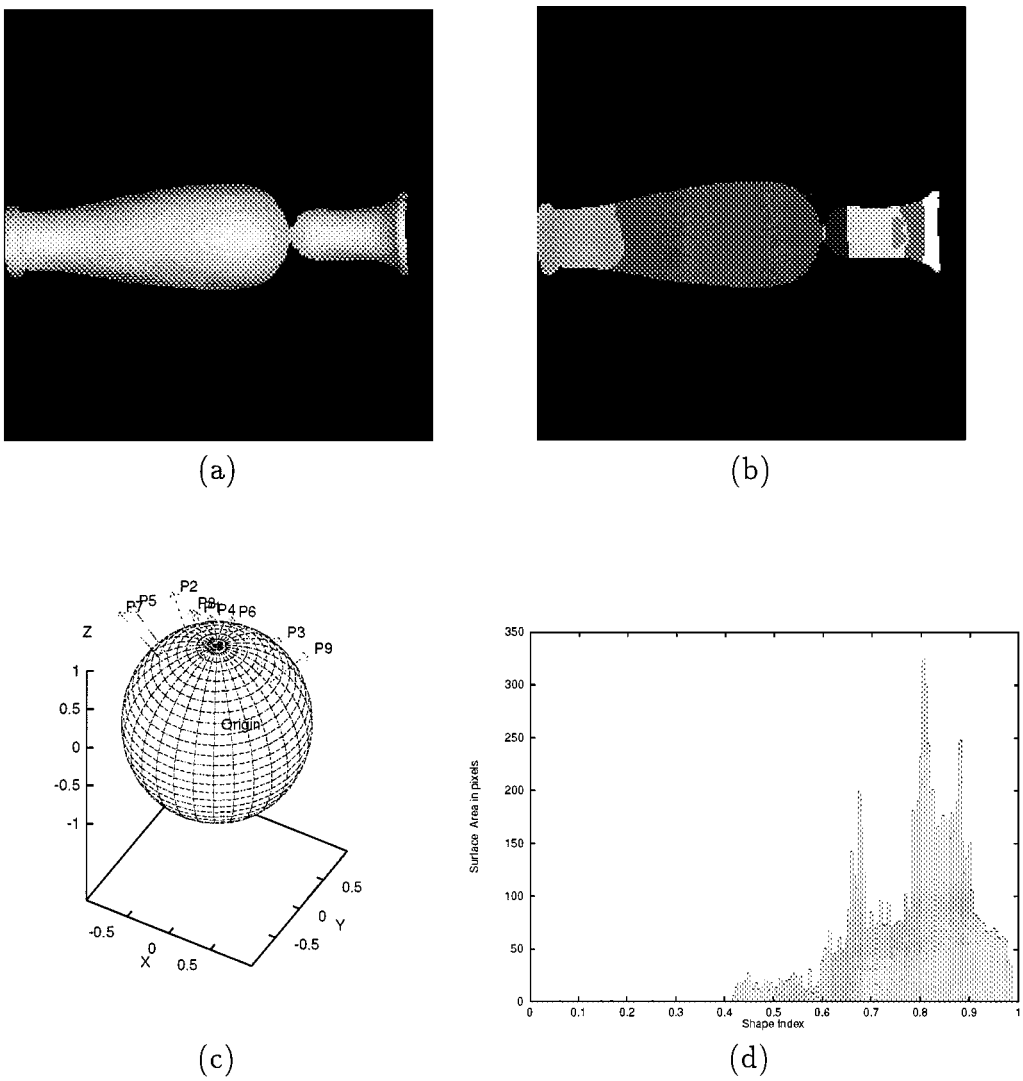

Fig. 7. *COSMOS of a view of a vase: (a) Range image; (b) constant shape maximal patches; (c) Gauss patch map; (d) shape spectrum*

COSMOS representation derived from an input view of an object with the stored representations?

Given a range image of an uncluttered view (allowing self-occlusion) of an object as input, the first stage of the proposed matching scheme matches the input view with the model object views efficiently on the basis of shape spectral information. As demonstrated in Dorai and Jain (1997b), shape spectra of object views can be used to sort a large model database into structurally homogeneous subsets, leading to a meaningful organization of the database and it can also be used for rapid pruning of the view database to select a small set of candidate model views that best match with the input. In the second level of matching, the detailed COSMOS representations of object views are exploited to determine the image-model feature correspondences using a combination of search methods, and thus identify and localize the object in the input view.

4.2.1. Shape spectrum based model selection

The COSMOS representation of each view of every object in the model database is computed and stored along with its object label and its pose vector, if already known. A feature vector is computed for the jth view of the ith object, O_j^i, $1 \leq i \leq N$, $1 \leq j \leq M$, based on the first ten moments of its shape spectrum, which has been normalized with respect to the visible object surface area (Dorai and Jain 1997b). This moment-based representation emphasizes the spread characteristics (variance) of the spectral distribution. The problems associated with bin quantization, that cause poor performance if direct comparison of two histograms is used, are also reduced. These moment features are best understood if we observe the likeness between the shape spectrum of an object view and a probability density function of a random variable.

In order to provide a meaningful categorization of views of an object O^i, views are clustered based on their pair-wise dissimilarities $\mathcal{D}(\mathbf{R}^i_j, \mathbf{R}^i_k)$ where $\mathcal{D}(\mathbf{R}^i_j, \mathbf{R}^i_k)$ is the Euclidean distance between the view moment vectors, $\mathbf{R}^i_j$ and $\mathbf{R}^i_k$, using a hierarchical clustering scheme such as the complete-link algorithm (Jain and Dubes 1988). The partition $\mathcal{P}^i$ is obtained by splitting the hierarchical grouping of O^i at a specific level of dissimilarity in the dendrogram. The split level is chosen at the dissimilarity value of 0.1 or less to result in a set of compact and well-separated clusters for each object. The database is then organized into a two-tiered structure: at the first level is the collection of view clusters obtained from each object, and at the second level each cluster contains views that have been judged to be similar based on the dissimilarity between the corresponding moment features of the shape spectra of the views. A summary representation for each view cluster is abstracted from the moment vectors of its constituent views by computing the centroid of the view cluster.

Given an input view, a small set of object views that are most similar to the input is determined quickly and accurately in two stages: first compare the moment vector of the input view with all the cluster summary representations and select the K best matched clusters; then match it with the moment vectors of the views in these best-matched clusters and select the top m closest views. This shape spectrum-based first-level matching stage results in a set of probable model object views that are similar to the input in terms of visible surface shapes. Experimental results on a database of 6,400 views of 20 free-form objects show that when tested with 2,000 independent views, our model selection technique examined, on the average, only 20% of the database for correct classification of the test views (Dorai 1996).

4.2.2. COSMOS based view matching

The function of this view verification stage is to determine a *correct* object match among the selected model view hypotheses. Given the COSMOS representations of two object views, the objective is to establish *correspondences* between the CSMPs in the views to determine the identity and the pose of the sensed object accurately. The matching algorithm however, must deal with noise, missing patches, and spurious information due to imperfect segmentation. Scene-model feature matching has been formulated as a subgraph isomorphism problem, exploiting the *region adjacency graph* data structure that can be abstracted from the COSMOS representation of an object view. In essence, our solution is to merge some patches into "patch-groups" (which along with their connectivity information define a "patch-group graph"), construct a consistent correspondence between the two patch-group graphs, and compare the resulting graphs, iteratively until they become isomorphic. This step constructs a correspondence between the views that maximizes a measure of goodness in terms of similarities in the shape index, mean curvedness, area and the neighbors' attributes between matched patch-groups in the graphs, given the topological constraints imposed by the surface connectivity information, and obtains the finest grain mapping possible between the patch-groups in the views. Once we ascertain the stored view in the model database that matches the best with the input view, with the highest goodness measure among all hypotheses, we estimate the coarse pose of the input object by aligning the surface normals of *corresponding* patch-groups. Once a coarse pose is obtained, it is refined using an optimal range image registration algorithm (Dorai 1996).

4.3. *Experimental results*

We examined the performance of our COSMOS based matching algorithm on range images of several pairs of views obtained from different objects. Figure 8 shows the correspondences between the CSMPs detected in the two views of Phone. Observe that since our matching algorithm does not model symmetry explicitly, the correspondence shown in Fig. 8(c) inversely matches the symmetrical structures in the two views.

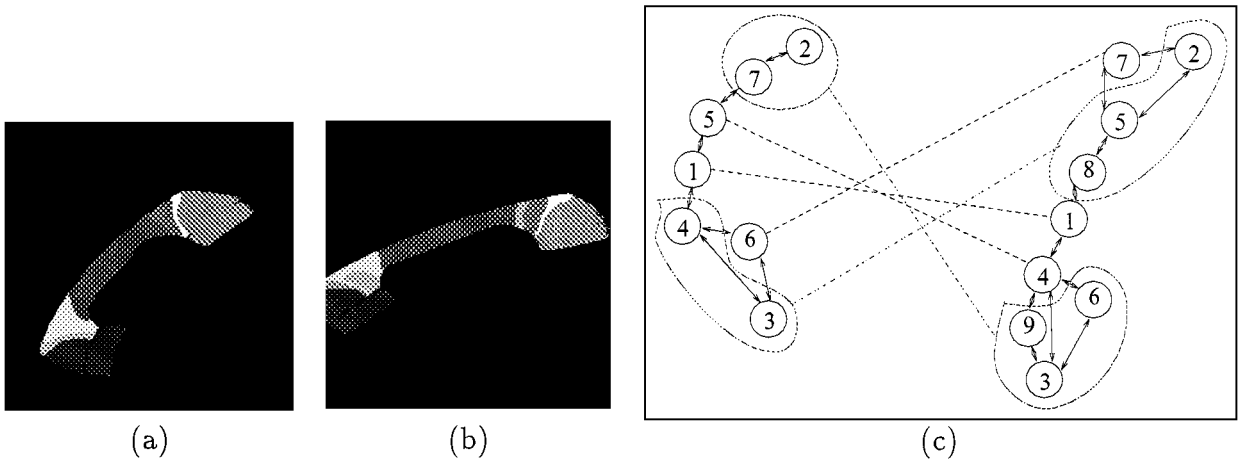

Fig. 8. *Correspondence between the views of Phone: (a) View 1 of Phone; (b) view 2 of Phone; (c) correspondence established between the CSMPs visible in the views*

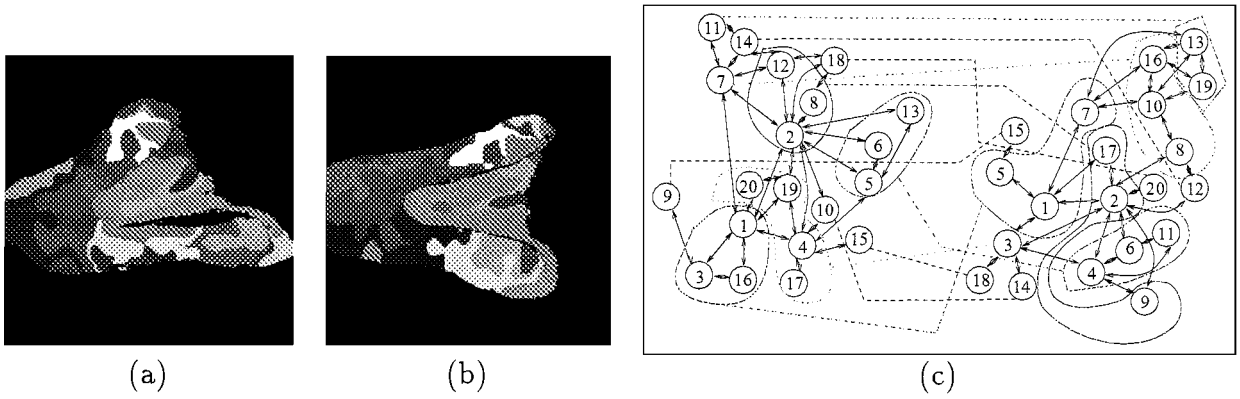

Fig. 9. *COSMOS-based matching: (a) CSMPs on the test view (view 1); (b) CSMPs on the model view with the highest matching score (view 2); (c) scene-model patch correspondence established by the algorithm between the views of Cobra*

The whole object recognition system was tested using 50 independent test range images on a database containing 50 model images obtained from ten different free-form objects. With the shape spectrum based model view selection scheme, 47 of the 50 test views were able to retrieve at least one model view from their correct object classes among the selected five view hypotheses, thus resulting in a view classification accuracy of 94%. Then the patch-group graph matching scheme was used to determine the best object view match among the five candidate view hypotheses short-listed using the shape spectral analysis for each test view. The recognition system was able to identify 82% of the fifty test views correctly by returning a model view from the correct object class with the highest matching score. Out of the nine test views incorrectly matched, three views did not have any view from the correct object class present among the five hypotheses that were examined using the detailed matching scheme. The remaining six errors were mainly due to errors in the surface connectivity information introduced by noisy small patches. Figure 9(a) and (b) show the segmentation of the test view and the stored model view with the highest matching score of 0.447. Observe that despite the differences in the segmentation results of these two views, our matching algorithm was able to successfully merge patches overcoming the imperfections arising from segmentation, and provide a structurally correct correspondence between the scene and model patches as shown in Fig. 9(c).

Since the system may fail to correctly recognize the object in the input scene when only model views from incorrect object classes are presented as hypotheses to the COSMOS-based detailed matching stage, we can enforce a reject option using a threshold on the dissimilarity levels of the hypotheses to prevent some matches from being further examined. The current version of our matching algorithm does not tolerate any violation of connectivity relationships between matched patch-groups, and as observed in our experiments, noisy small patches can introduce errors in the adjacency relationships between the patches thus affecting the recognition accuracy. The algorithm can be improved by allowing violations of the connectivity to a small degree depending on the strength of the adjacency as determined by the number of boundary pixels that are shared between a pair of patches.

5. Summary and future directions

We have surveyed 3D object representation and recognition schemes and highlighted the inadequacy of a number of prevalent representation techniques in handling free-form objects. We have presented an overview of our research effort, specifically attempting to solve the free-form surface matching problem using a novel 3D object representation scheme called COSMOS. We described a novel multi-level matching strategy that employs shape spectral analysis and features derived from the COSMOS representations of objects for fast and accurate recognition of sculpted objects. Experiments on a database of 100 range images of several complex objects acquired using a laser range scanner have demonstrated the strengths of our COSMOS based recognition system.

Several research trends in the area of 3D object recognition can be identified. The first involves careful evaluation and comparison of the performance of various approaches using larger sets of images, in specific application contexts (Ponce, Hebert and Zisserman 1996). Secondly, representation of objects, besides being the core area of object recognition research, has become a prominent issue in several application domains. One active area is that of building models of physical objects for which CAD models may be unavailable. Accurate models of free-form objects are required in emerging applications such as object animation and visualization in virtual museums, vision augmented reality, and on-line services such as e-business and Internet shopping. Processing and transmitting images generated from 3D object models for interactive viewing poses another challenge. The explosion of multimedia applications such as object-based coding and content-based image and video retrieval has provided new impetus to defining generic models of 3D objects. Advances in medical imaging have resulted in demand

for systems that perform 3D data registration for image-guided surgeries. Vision-based user interfaces incorporating gaze control and gesture recognition are seen as the next generation of systems for human-computer interaction. This continual emergence of new technologies and applications that build upon basic research in object recognition promises an exciting future.

References

Arman F. and Aggarwal J.K. 1993. Model-based object recognition in dense range images—A review. ACM Computing Surveys 25(1): 5–43.

Bajcsy R. and Solina F. 1987. Three-dimensional object representation revisited. In: Proc. First IEEE International Conference on Computer Vision, London, pp. 231–240.

Barrow H.G. and Burstall R.M. 1976. Subgraph isomorphism, matching relational structures and maximal cliques. Information Processing Letters 4: 83–84.

Basri R. and Ullman S. 1988. The alignment of objects with smooth surfaces. In: Proc. Second IEEE International Conference on Computer Vision, Tarpon Springs, FL, pp. 482–488.

Besl P.J. 1988. Surfaces in Range Image Understanding, Springer Series in Perception Engineering. Springer-Verlag.

Besl P.J. 1990. The free-form surface matching problem. In: Freeman H. (Ed.), Machine Vision for Three-Dimensional Scenes. Academic Press, pp. 25–71.

Besl P.J. and Jain R. 1985. Three-dimensional object recognition. ACM Computing Surveys 17: 75–145.

Besl P.J. and McKay N.D. 1992. A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2): 239–256.

Biederman I. 1987. Recognition-by-components: A theory of human image understanding. Psychological Review 94(2): 115–147.

Bolles R. and Horaud P. 1986. 3DPO: A three-dimensional part orientation system. International Journal of Robotics Research 5(3): 3–26.

Borges D.L. and Fisher R.B. 1997. Class-based recognition of 3D objects represented by volumetric primitives. Image and Vision Computing 15: 655–664.

Breen D., Whitaker R., Rose E., and Tuceryan M. 1996. Interactive occlusion and automatic object placement for augmented reality. In: Proc. Eurographics '96, Poitiers, France. Elsevier Science Publishers, B.V., pp. 11–22.

Brooks R.A. 1983. Model-based three-dimensional interpretations of two-dimensional images. IEEE Transactions on Pattern Analysis and Machine Intelligence 5: 140–150.

Chen C. and Kak A. 1989. A robot vision system for recognizing 3-D objects in low-order polynomial time. IEEE Transactions on Systems, Man, and Cybernetics 19(6): 1535–1563.

Chen T.-W. and Lin W.-C. 1994. A neural network approach to CSG-based 3-D object recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(7): 719–726.

Chen J.-L. and Stockman G.C. 1996. Determining pose of 3D objects with curved surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 18: 52–56.

Chin R.T. and Dyer C.R. 1986. Model-based recognition in robot vision. ACM Computing Surveys 18(1): 67–108.

Connell J.H. and Brady M. 1987. Generating and generalizing models of visual objects. Artificial Intelligence 31: 159–183.

Delingette H., Hebert M., and Ikeuchi K. 1993. A spherical representation for the recognition of curved objects. In: Proc. Fourth IEEE International Conference on Computer Vision, Berlin, pp. 103–112.

Dickinson S.J., Pentland A.P., and Rosenfeld A. 1992. 3-D shape recovery using distributed aspect matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 14: 174–198.

Dorai C. 1996. COSMOS: A framework for representation and recognition of 3D free-form objects. PhD Thesis, Department of Computer Science, Michigan State University, East Lansing.

Dorai C. and Jain A.K. 1997a. COSMOS—A representation scheme for 3D free-form objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(10): 1115–1130.

Dorai C. and Jain A.K. 1997b. Shape spectrum based view grouping and matching of 3D free-form objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(10): 1139–1146.

Eggert D. and Bowyer K. 1993. Computing the perspective projection aspect graph of solids of revolution. IEEE Transactions on Pattern Analysis and Machine Intelligence 15: 109–128.

Executive Office of the President, Office of Science and Technology Policy. 1989. The Federal High Performance Computing Program. Washington, D.C.

Fan T.-J., Medioni G., and Nevatia R. 1989. Recognizing 3-D objects using surface descriptions. IEEE Transactions on Pattern Analysis and Machine Intelligence 11(11): 1140–1157.

Faugeras O. and Hebert M. 1986. The representation, recognition, and locating of 3-D objects. International Journal of Robotics Research 5(3): 27–52.

Ferrie F.P., Mathur S., and Soucy G. 1993. Feature extraction for 3-D model building and object recognition. In: Jain A.K. and Flynn P.J. (Eds.), Three-Dimensional Object Recognition Systems. Elsevier Science Publishers, B.V., Amsterdam, The Netherlands, pp. 57–88.

Fischler M. and Bolles R. 1981. Random consensus: A paradigm for model-fitting with applications in image analysis and automated cartography. Communications of the ACM 24: 381–395.

Flynn P.J. and Jain A.K. 1991. BONSAI: 3D object recognition using constrained search. IEEE Transactions on Pattern Analysis and Machine Intelligence 13(10): 1066–1075.

Flynn P.J. and Jain A.K. 1992. 3D object recognition using invariant feature indexing of interpretation tables. CVGIP: Image Understanding 55(2): 119–129.

Flynn P.J. and Jain A.K. 1994. Three-dimensional object recognition. In: Young T.Y. (Ed.), Handbook of Pattern Recognition and Image Processing, Vol. 2. Academic Press, ch. 14, pp. 497–541.

Grimson W.E.L. and Lozano-Pérez T. 1984. Model-based recognition and localization from sparse range or tactile data. International Journal of Robotics Research 3: 3–35.

Grimson W.E.L. and Lozano-Pérez T. 1987. Localizing overlapping parts by searching the interpretation tree. IEEE Transactions on Pattern Analysis and Machine Intelligence 9: 469–482.

Grimson W., Lozano-Perez T., Wells W.M. III, Ettinger G., White S., and Kikinis R. 1994. An automatic registration method for frameless stereotaxy, image guided surgery, and enhanced reality visualization. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Seattle, Washington, pp. 430–436.

Gupta A. and Bajcsy R. 1992. Surface and volumetric segmentation of range images using biquadrics and superquadrics. In: Proc. 11th

International Conference on Pattern Recognition, The Hague, The Netherlands, pp. 158–162.

Gupta A., Bogoni L., and Bajcsy R. 1989. Quantitative and qualitative measures for the evaluation of the superquadric models. In: Proc. IEEE Workshop on Interpretation of 3D Scenes, Austin, pp. 162–169.

Heap A.J. and Hogg D.C. 1995. Automated pivot location for the Cartesian-polar hybrid point distribution model. In: Proc. 6th British Machine Vision Conference, Birmingham, U.K., pp. 97–106.

Horn B.K.P. 1984. Extended Gaussian image. In: Proceedings of the IEEE 72: 1671–1686.

Horowitz B. and Pentland A.P. 1991. Recovery of non-rigid motion and structure. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Maui, Hawaii, pp. 288–293.

Huttenlocher D.P. and Ullman S. 1990. Recognizing solid objects by alignment with an image. International Journal of Computer Vision 5(2): 195–212.

Ikeuchi K. and Hong K.S. 1991. Determining linear shape change: Toward automatic generation of object recognition programs. CVGIP: Image Understanding 53(2): 154–170.

Jain A.K. and Dubes R.C. 1988. Algorithms for Clustering Data. NJ, Prentice Hall, Englewood Cliffs.

Jain A.K. and Flynn P.J. (Eds.). 1993. 3D Object Recognition Systems. Elsevier Science Publishers, B.V., Amsterdam, The Netherlands.

Jain A.K. and Hoffman R.L. 1988. Evidence-based recognition of 3-D objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 10: 783–802.

Jarvis R. 1993. Range sensing for computer vision. In: Jain A.K. and Flynn P.J. (Eds.), Three-dimensional Object Recognition Systems. Elsevier Science Publishers, B.V., Amsterdam, The Netherlands, pp. 17–56.

Kalawsky R.S. 1993. The Science of Virtual Reality and Virtual Environments. Addison Wesley.

Kang S.B. and Ikeuchi K. 1993. The complex EGI: A new representation for 3-D pose determination. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(7): 707–721.

Keren D., Cooper D., and Subrahmonia J. 1994. Describing complicated objects by implicit polynomials. IEEE Transactions on Pattern Analysis and Machine Intelligence 16: 38–53.

Koenderink J.J. and van Doorn A.J. 1979. Internal representation of solid shape with respect to vision. Biological Cybernetics 32(4): 211–216.

Koenderink J.J. and van Doorn A.J. 1992. Surface shape and curvature scales. Image and Vision Computing 10: 557–565.

Krishnapuram R. and Casasent D. 1989. Determination of three-dimensional object location and orientation from range images. IEEE Transactions on Pattern Analysis and Machine Intelligence 11(11): 1158–1167.

Lamdan Y., Schwartz J.T., and Wolfson H.J. 1990. Affine invariant model-based object recognition. IEEE Transactions on Robotics and Automation 6(5): 578–589.

Lamdan Y. and Wolfson H. 1988. Geometric hashing: A general and efficient model-based recognition scheme. In: Proc. Second IEEE International Conference on Computer Vision, Tarpon Springs, Florida, pp. 238–249.

Lanitis A., Taylor C.J., and Cootes T.F. 1997. Automatic interpretation and coding of face images using flexible models. IEEE Transactions on Pattern Analysis and Machine Intelligence 19: 743–756.

Liang P. and Taubes C.H. 1994. Orientation-based differential geometric representations for computer vision applications. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(3): 249–258.

Lowe D.G. 1987. Three-dimensional object recognition from single two-dimensional images. Artificial Intelligence 31: 355–395.

Marr D. and Nishihara H.K. 1978. Representation and recognition of the spatial organization of three-dimensional shapes. Proc. Royal Society, London, ser. B 200: 269–294.

Matsuo H. and Iwata A. 1994. 3-D object recognition using MEGI model from range data. In: Proc. 12th International Conference on Pattern Recognition, Jerusalem, Israel, pp. 843–846.

Mercer C.R. and Beheim G. 1990. Fiber-optic projected-fringe digital interferometry. In: Hologram Interferometry and Speckle Metrology Proceedings, Bethel, CT. Society for Experimental Mechanics, pp. 210–216.

Murase H. and Nayar S.K. 1995. Visual learning and recognition of 3-D objects from appearance. International Journal of Computer Vision 14(1): 5–24.

Nalwa V.S. 1989. Representing oriented piecewise C^2 surfaces. International Journal of Computer Vision 3: 131–153.

Pentland A.P. 1986. Perceptual organization and the representation of natural form. Artificial Intelligence. 28: 293–331.

Pentland A.P. 1990. Automatic extraction of deformable part models. International Journal of Computer Vision 4: 107–126.

Plantinga W.H. and Dyer C.R. 1990. Visibility, occlusion, and the aspect graph. International Journal of Computer Vision 5: 137–160.

Poggio T. and Edelman S. 1990. A network that learns to recognize three-dimensional objects. Nature 343: 263–266.

Ponce J., Hebert M., and Zisserman A. 1996. Report on the 1996 international workshop on object representation in computer vision. In: Ponce J., Zisserman A., and Hebert M. (Eds.), Proc. Intl. Workshop on Object Representation in Computer Vision II, pp. 1–8.

Ponce J., Hoogs A., and Kreigman D.J. 1992. On using CAD models to compute the pose of curved 3D objects. CVGIP: Image Understanding 55(2): 184–197.

Ponce J., Kriegman D.J., Petitjean S., Sullivan S., Taubin G., and Vijayakumar B. 1993. Representations and algorithms for 3D curved object recognition. In: Jain A.K. and Flynn P.J. (Eds.), Three-Dimensional Object Recognition Systems. Elsevier Science Publishers, B.V., Amsterdam, The Netherlands, pp. 17–56.

Raja N.S. and Jain A.K. 1994. Obtaining generic parts from range images using a multi-view representation. CVGIP: Image Understanding 60: 44–64.

Roberts, L. 1965. Machine perception of three-dimensional solids. In: Tippett J.T., Berkowitz D.A., Clapp L.C., Koester C.J., and Alexander Vanderburgh J. (Eds.), Optical and Electro-Optical Information Processing. MIT Press, Cambridge, Massachusetts, pp. 159–197.

Sallam M. and Bowyer K. 1994. Registering time sequences of mammograms using a two-dimensional image unwarping technique. In: Second International Workshop on Digital Mammography, pp. 121–130.

Samet H. 1990. The Design and Analysis of Spatial Data Structures. Addison-Wesley.

Seales W.B. and Dyer C.R. 1992. Viewpoint from occluding contour. CVGIP: Image Understanding 55(2): 198–211.

Silberberg T.M., Davis L., and Harwood H. 1984. An iterative Hough procedure for three-dimensional object recognition. Pattern Recognition 17(6): 621–629.

Sinha S.S. and Jain R. 1994. Range image analysis. In: Young T.Y. (Ed.), Handbook of Pattern Recognition and Image Processing: Computer Vision, Vol. 2. Academic Press, ch. 14, pp. 185–237.

Solina F. and Bajcsy R. 1990. Recovery of parametric models from range images: The case for superquadrics with global deformations. IEEE Transactions on Pattern Analysis and Machine Intelligence 12: 131–147.

Stark L. and Bowyer K.W. 1991. Achieving generalized object recognition through reasoning about association of function to structure. IEEE Transactions on Pattern Analysis and Machine Intelligence 13: 1097–1104.

Stein F. and Medioni G. 1992. Structural indexing: Efficient 3-D object recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2): 125–145.

Stockman G.C. 1987. Object recognition and localization via pose clustering. Computer Vision, Graphics, and Image Processing 40: 361–387.

Suetens P., Fua P., and Hanson A.J. 1992. Computational strategies for object recognition. ACM Computing Surveys 24: 5–61.

Swets D.L. 1996. The self-organizing hierarchical optimal subspace learning and inference framework for object recognition. PhD Thesis, Michigan State University, Department of Computer Science, East Lansing, Michigan.

Taubin G., Cukierman F., Sullivan S., Ponce J., and Kreigman D.J. 1992. Parametrized and fitting bounded algebraic curves and surfaces. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Champaign, Illinois, pp. 103–108.

Terzopoulos D. and Metaxas D. 1991. Dynamic 3D models with local and global deformations: Deformable superquadrics. IEEE Transactions on Pattern Analysis and Machine Intelligence 13: 703–714.

Turney J.L., Mudge T.N., and Volz R.A. 1985. Recognizing partially occluded parts. IEEE Transactions on Pattern Analysis and Machine Intelligence 7(4): 410–421.

Ullman S. and Basri R. 1991. Recognition by linear combination of models. IEEE Transactions on Pattern Analysis and Machine Intelligence 13: 992–1006.

Umeyama S. 1993. Parameterized point pattern matching and its application to recognition of object families. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(2): 136–144.

Vayda A. and Kak A. 1991. A robot vision systems for recognition of generic shaped objects. CVGIP: Image Understanding, 54: 1–46.

Vemuri B. and Aggarwal J. 1987. Representation and recognition of objects from dense range maps. IEEE Transactions on Circuits and Systems CAS-34: 1351–1363.

Weld D.S. (Ed.). 1995. The role of intelligent systems in the National Information Infrastructure. AI Magazine 16(3): 45–64.

Wong A.K.C., Lu S.W., and Rioux M. 1989. Recognition and shape synthesis of 3D objects based on attributed hypergraphs. IEEE Transactions on Pattern Analysis and Machine Intelligence 11(3): 279–290.