A New Signature-Based Method for Efficient 3-D Object Recognition

Salvador Ruiz-Correa†, Linda G. Shapiro† and Marina Meliă‡
†Department of Electrical Engineering
‡Department of Statistics
University of Washington, Seattle, WA 98105
{sruiz@u, shapiro@cs, mmp@stat}.washington.edu

Abstract

This paper considers the problem of shape-based recognition and pose estimation of 3-D free-form objects in scenes that contain occlusion and clutter. Our approach is based on a novel set of discriminating descriptors called spherical spin images, which encode the shape information conveyed by classes of distributions of surface points constructed with respect to reference points on the surface of an object. The key to this approach is the relationship that exists between the l_2 metric, which compares n-dimensional signatures in Euclidean space, and the metric of the compact space on which the class representatives (spherical spin images) are defined. The connection allows us to efficiently utilize the linear correlation coefficient to discriminate scene points which have spherical spin images that are similar to the spherical spin images of points on the object being sought. The paper also addresses the problem of a compressed spherical-spin-image representation by means of a random projection of the original descriptors that reduces the dimensionality without a significant loss of recognition/localization performance. Finally, the efficacy of the proposed representation is validated in a comparative study of the two algorithms introduced here that use uncompressed and compressed spherical spin images versus two previous spin image algorithms reported recently in the literature. The results of 2012 experiments suggest that the performance of our proposed algorithms is significantly better with respect to accuracy and speed than the performance of the other algorithms tested.

1. Introduction

With the increased availability and decreased prices for 3-D scanners, the use of range data for recognizing and locating 3-D objects in complex scenes is a feasible option. Shape-based recognition systems that use surface signatures to represent the shape of the object are particularly attractive, because they can handle a wide-variety of objects, including those whose shape cannot be easily approximated with parametric models (often called "free-form" objects). A surface signature at a given point on the surface of an object is a descriptor that encodes the geometric properties measured in a neighborhood of the point. Curvature is one of the oldest and most basic local descriptors of shape. In early work, Besl and Jain [1] characterized surface points according to the signs of their mean and Gaussian curvatures, which could then be used to classify points into symbolic categories, such as peaks, pits, ridges, and valleys. Faugeras and Hebert [6] used curvature for detecting primitive features (points, lines, planes, and quadric patches) in range data scenes. In recent years, more complex surface-signature representation schemes have been reported in the literature. They include the splash representation of Stein and Medioni [13], the point signatures of Chua and Jarvis [2], the shape spectrum scheme of Dorai and Jain [4], the surface signatures from simplex meshes of Yamany *et al.*, [14], the harmonic shape images of Zhang and Hebert [15], and the spin-image representation introduced by Johnson and Hebert [9].

The problem of shape-based 3-D object recognition in complex scenes is difficult for two principal reasons. In the first place, real range data scenes generally contain multiple objects. The clutter due to the presence of surface points that are not part of the object being sought can cause confusion in the recognition process. In the second place, scenes also contain varying levels of occlusion, so there is only partial information pertaining to the object of interest.

This paper addresses the problems described above by proposing a simple and general representation of shape that is amenable for effectively recognizing and locating objects in complex 3-D scenes. The *spherical-spin-image* representation (related to the spin-image approach introduced in [9]) is a general representation of shape based on a collection of descriptors (*spherical spin images*) that are robust to scene clutter and occlusion. The paper also considers reducing the dimensionality of the spherical spin images by

0-7695-1272-0/01 $10.00 © 2001 IEEE

means of a random projection to a subspace of lower dimension, thus accomplishing a compact representation of shape we call *compressed spherical-spin images*. Random projections are transformations that are much faster than traditional transforms like the Karhunen-Loeve Transform (KLT), since they do not depend on the data to be compressed. They were introduced in [7] and used subsequently in [3] for density estimation and pattern recognition applications. Finally, the paper presents two algorithms that use uncompressed and compressed spherical spin images to recognize and locate objects in range data scenes. A performance evaluation is conducted to compare these two algorithms to their counterpart spin image algorithms given in [9]. The results of 2012 experiments suggest that the spherical spin-image representation significantly improves the recognition/localization performance and dramatically reduces the matching time of the spin-image algorithm. It also suggest that the compressed spherical spin-image representation outperforms the recognition/localization rates of its compressed spin-image counterpart.

The rest of our paper is structured as follows. We describe the original spin-image representation in Section 2, since it motivated our representation of shape. Section 3 is devoted to the spherical-spin image representation and Section 4 to our recognition algorithms. Section 5 discusses the results of the comparative study, and Section 6 concludes the paper.

2. Spin Images

In [9] Johnson and Hebert introduce an elegant and powerful representation for surface matching. The *spin image* representation comprises a set of descriptive images associated with the oriented points on the surface of an object. These images are created by constructing a pose-invariant 2-D coordinate system at an oriented point (3-D point with normal vector) on the surface, and accumulating the coordinates (α, β) of other points in a 2-D histogram called a spin image as explained in Figure 1. The representation assumes that surfaces are approximated by regular polygonal surface meshes, of a given mesh resolution, defined as the median length of all edges in the mesh. The spin image representation is robust to scene clutter and occlusion and therefore suitable for object recognition posed as a surface-matching problem.

Matching surfaces with spin images consists of finding correspondences between surface points of two instances of the same object, from which a rigid transformation that registers the surfaces is calculated. Correspondences are found by comparing spin images from points of one surface (usually the model) with spin images from points of a second surface (usually the scene) using a loss function of the linear correlation coefficient as a measure of similarity, and

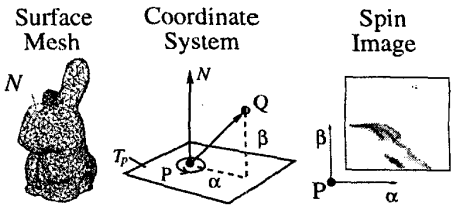

Figure 1. The spin image for point P is constructed by accumulating in a 2-D histogram the coordinates α and β of a set of contributing points (such as Q) on the mesh representing the object . Contributing points are those that are within a specified distance of P and for which the surface normal forms an angle of less than a specified size with the surface normal N of P. The coordinate α is the distance from P to the projection of Q onto the tangent plane T_p; β is the distance from Q to this plane.

selecting the correspondence pairs with the highest similarity that are geometrically consistent. The authors of [9] showed that the method described above must be modified in order to extend surface matching using spin images to object recognition in complex scenes. Finding correspondences using the correlation coefficient is computationally expensive [9], and therefore, a different way of managing the information conveyed by the spin images is needed. In order to make the process of matching efficient, dimensionality reduction was achieved by projecting spin images represented as n-tuples to a space of dimension $d < n$, using principal component analysis (PCA). This compressed representation allows a significant improvement in computational efficiency that justifies an acceptable reduction of the recogition performance.

In the compressed spin-images representation, correspondences are found by using an efficient closest-point search structure that looks for the best-matching model spin image to a given scene spin image in a search space of given dimensions and shape. Thus, finding closest tuples in a subspace of R^d replaces correlating spin images [9]. One drawback of this compressed representation is that the magnitude of the d-tuples becomes dependent on the resolution of the mesh. In principle it is possible to control the resolution by using resampling algorithms, but in some cases data cannot be set to a specific resolution without losing important information related to their geometrical properties. In the sections that follow we show that the spherical spin image solves this problem and improves the overall performance of the spin-image representation in the context of object recognition/localization tasks.

3. Spherical Spin Images (SSI)

Like spin images, spherical spin images are signatures associated with the vertices of a polygonal mesh of a given resolution that approximates the surface of an object. Spherical spin images are points of the unit sphere in R^n that identify the equivalence classes of spin images induced by the equivalence relation derived from the linear correlation coefficient; a pair of spin images is equivalent if their correlation coefficient equals one or, in practice, are highly correlated. Figure 2 illustrates the relationship between spin images represented as vectors in R^n and spherical spin images on the unit sphere S^{n-1} and shows several examples of spherical spin images for selected points of a surface mesh.

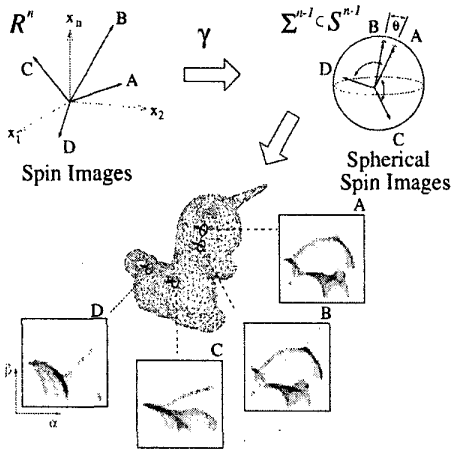

Figure 2. Spherical spin images for four oriented points (3-D points plus normal vectors) A, B, C and D on the polygonal mesh of a model. They are obtained from spin images via the mapping γ defined in Section 3.1. The correlation coefficient is calculated as the cosine of the angle between spherical spin images. The angle θ for the elements labeled as A and B on the sphere is small indicating that the spherical spin images are highly correlated. The angle between points A and C (D) is large, indicating lack of positive correlation. Spherical spin images are n-dimensional unit vectors that can be represented as images of n pixels, as shown in the figure.

The mathematical tools needed for creating spherical spin images are described in this section and summarized in Figure 3. There are three main geometric spaces involved and three mappings that relate each space to the others. The function γ assigns spin images to spherical spin images. The function g maps equivalence classes of spin images to spherical spin images, and the function e maps each spin image to its equivalence class. These functions are related by the equation $\gamma = g \circ e$, where $\circ$ denotes function composition. We start by introducing our notation.

Throughout this paper we let R^n denote the set of all ordered sets of n-tuples of real numbers. If x is an element of R^n the coordinates of x will be denoted by x_i; hence, $x = (x_1, \ldots, x_n)$. The letter o denotes the origin $(0, \ldots, 0)$ of R^n. We treat R^n as a vector-space structure with componentwise addition and scalar multiplication. That is, the n-tuples are also considered to be vectors of n components. If $u, v \in R^n$ we let $u \cdot v$ denote the inner product, and $\mid u \mid$ the Euclidean norm. We let S^{n-1} denote the unit sphere centered at o, that is $S^{n-1} = \{x = (x_1, \cdots, x_n) \mid x_1^2 + \cdots + x_n^2 = 1\}$.

We let ρ denote the *linear correlation coefficient*, which measures the degree of linear relationship between pairs of n-tuples in R^n. This means that for two elements x and y of R^n, $\rho(x, y) = 1$ if and only if all (x_i, y_i) pairs lie on a straight line with positive slope, and $\rho(x, y) = -1$ if and only if all (x_i, y_i) pairs lie on a straight line with negative slope.

There is a well-known geometrical interpretation of the linear correlation coefficient, as the cosine of the angle between two unit vectors, that is most convenient for our purposes. To introduce this notion and simplify our notation, we need to define two mappings: μ and χ. The function μ maps elements of R^n to n-tuples of the form $(m(x), \cdots, m(x))$, where $m(x) = \frac{1}{n}\sum_{i=1}^{n} x_i$. Therefore, $\mu(x)$ is an n-dimensional vector, all of whose elements are the mean of the n components of x. We let χ denote the function defined as

$$\chi(x) = \frac{x - \mu(x)}{\mid x - \mu(x) \mid}. \tag{1}$$

The function χ produces an n-tuple (the unit difference vector) $\chi(x)$ whose ith component is the normalized difference between the ith component of x and the mean of the components of x. The range of χ denoted by Λ^{n-1} consists of points that lie on S^{n-1}. This map allows us to write the correlation coefficient as

$$\begin{aligned} \rho(x, y) &= \sum_{i=1}^{n} \frac{x_i - m(x)}{\mid x - \mu(x) \mid} \times \frac{y_i - m(y)}{\mid y - \mu(y) \mid} \\ &= \chi(x) \cdot \chi(y) = \cos\theta_{\chi(x)\chi(y)}, \end{aligned}$$

the cosine of the angle between the unit vectors $\chi(x)$ and $\chi(y)$. Notice that $\rho(x, y) = 1$ if and only if χ maps x and y to the same unit vector; that is, if and only if $y = ax + (b, \cdots, b)$, $a, b \in R$ and $a > 0$. Also note that the elements in the domain of χ belong to the complement of the set

$$\mathcal{N} = \{x \mid x \in R^n, \mid x - \mu(x) \mid = 0\}.$$

The collection of spin images represented as n-tuples can be defined as the set $\Gamma^n \subset \mathcal{N}^c$. It consists of elements of

$\mathcal{N}^c$ that have non-negative components. This means that

$$\Gamma^n = \{x \mid x \in \mathcal{N}^c, x_i \geq 0, \forall i = 1, \cdots, n\}. \quad (2)$$

This definition is consistent with the original spin image formulation of Johnson. The elements of Γ^n produce well-defined values for the correlation coefficient, and they have nonnegative components, since they are histograms.

3.1 Equivalence Classes of Spin Images.

In this section we construct the set of spherical spin images in two steps. First we define the set of equivalence classes of spin images induced by the equivalence relation given by the linear correlation coefficient. Then, we define the function γ that induces the same equivalence relation and allows us to define the spherical spin images as the elements of its range and the function g that allows us to identify the spherical spin images as "labels" attached to the equivalence classes.

The linear correlation coefficient defines an equivalence relation on the set of spin images. Two spin images x, y are equivalent if and only if their correlation coefficient equals one. This equivalence relation partitions the set of spin images into *equivalence classes* that constitute the set Γ^n/E; the equivalence class under E of any element $x \in \Gamma^n$ is formally defined as

$$[x] = \{y \mid y \in \Gamma^n \text{ and } xEy\},$$

and the set of all possible equivalence classes (or quotient set) of E, as

$$\Gamma^n/E = \{C \mid C \subset \Gamma^n \text{ and } C = [x] \text{ for some } x \in \Gamma^n\}.$$

Here the notation xEy indicates that the spin images x and y are equivalent.

A standard theorem in algebra shows that given the equivalence relation E on Γ^n and a function $\gamma : \Gamma^n \mapsto \Sigma^{n-1}$ such that xEy implies $\gamma(x) = \gamma(y)$, there exists exactly one function $g : \Gamma^n/E \mapsto \Sigma^{n-1}$ for which $\gamma = g \circ e$, with $e(x) = [x]$ for each $x \in \Gamma^n$. Furthermore, if γ is a surjection (1-1), then g is a bijection (1-1 and onto). The equivalence relation E is called the *equivalence kernel* of γ.

The key points of the theorem are: 1) the function γ defined on Γ^n implies the same equivalence relation E defined by the correlation coefficient; and 2) for surjective γ the function g is bijective, so the elements of Σ^{n-1} "label" the set of equivalence classes. We define the surjective function γ such that E is its equivalence kernel; that is, γ is a restriction of the unit-difference-function $\chi : \mathcal{N}^c \mapsto \Lambda^{n-1}$ such that $\gamma(x) = \chi(x)$ for each $x \in \Gamma^n$, with $\Gamma^n \subset \mathcal{N}^c$ and $\Sigma^{n-1} \subset \Lambda^{n-1}$. (See Figure 3).

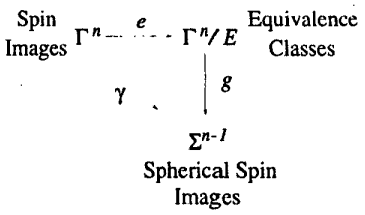

Figure 3. Commutative diagram that shows the relation between the sets of spin images Γ^n, the set of equivalence classes of spin images Γ^n/E and the set of spherical spin images $\Sigma^{n-1} \subset S^{n-1}$.

Definition 1 Spherical spin images *are elements of the set* $\Sigma^{n-1} \subset S^{n-1}$, *which label the elements of the quotient set of spin images induced by the equivalence kernel of* γ.

Some remarks are of importance. 1) Spherical spin images are points on the unit sphere, but since we are assuming a vector space structure in R^n, spherical spin images can also be defined as vectors of unit length. In what follows we refer to spherical spin images as elements of S^{n-1} or vectors of unit length, interchangeably. 2) The value of the linear correlation coefficient between spin images equals the cosine of the angle between the corresponding spherical spin images. That is, for each pair $x, y \in \Gamma^n$ $\rho(x, y) = \cos\theta_{\gamma(x)\gamma(y)}$. 3) The angle between spherical spin images u and v is related to the Euclidean distance between them by the formula $\mid u - v \mid = 2\sin(\theta_{uv}/2)$. This means that $\mid u - v \mid = 2\sin(\cos^{-1}(\rho(u, v))/2)$, where $\rho(u, v) = cos\theta_{uv}$. Σ^{n-1} is a metric space. 4) Spin images and spherical spin images are characterized by four parameters when represented as square matrices: bin size, image width, support width and support angle. See [9] for details. 5) The approach used here for representing shape in terms of equivalence classes generalizes to any kind of surface signatures that can be discriminated by the correlation coefficient.

3.2 Compressing Spherical Spin Images

Compression of spherical spin images is achieved by means of a random projection technique which offers a nice distance preservation property [7] and does not depend on the data set to be compressed. It consists of projecting a given point in R^n to a random d-dimensional hyperplane spanned by an orthonormal set of column vectors $U_1, U_2, \cdots, U_d$, whose joint distribution is invariant to rotations of the coordinate system. More precisely, $U_1, U_2, \cdots, U_d$ have the same joint distribution as $GU_1, GU_2, \cdots, GU_d$ for every orthogonal linear transform G which leaves the origin fixed. The following algorithm

generates a random projection matrix $U = [U_1 U_2 \cdots U_d]$. First, each entry of a $n \times d$ matrix is set to an i.i.d. $N(0,1)$ value. Then the d columns of the matrix are made orthogonal by using the Gram-Schmidt algorithm. Finally, the rows are normalized to unit length. This takes $O(d^2 n)$ time overall compared with the PCA, which takes $O(n^3)$. Compressed spherical spin images are obtained from a *fixed* matrix U, generated with the algorithm just described, as follows. For each $\gamma(x) \in \Sigma^{n-1}$, $x \in \Gamma^{n-1}$, the corresponding compressed instance is $\gamma(x^T U) \in \Sigma^{d-1}$ a d-dimensional unit vector. (Notice the abuse of notation in this expression in which we use the same symbol γ to denote different functions.) This compressed representation is *independent* of the resolution of the surface mesh of the models as opposed to the PCA representation.

Principal component analysis is an extremely important tool for data analysis and dimensionality reduction, however a recent comparative study between PCA and the random projection technique has shown that random projections guarantee a certain level of performance regardless of the individual data set, while the usefulness of the PCA varies from data set to data set [3]. We observed this phenomenon in the comparative study presented in Section 5. Our proposed algorithm for recognizing and locating 3-D objects in complex scenes based on random projections outperforms in terms of (offline and online) speed and accuracy an algorithm based on PCA.

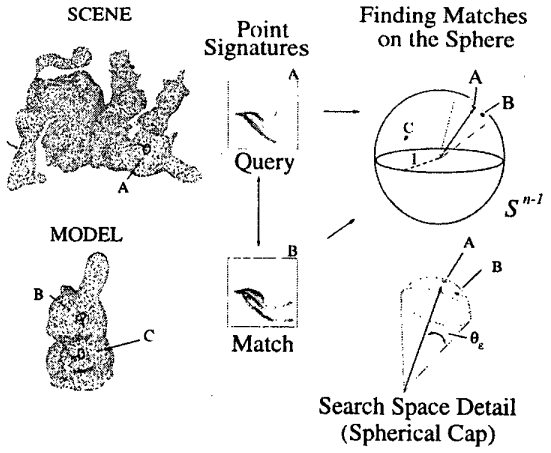

Figure 4. Finding potential matches for the scene point A consists of searching for model spherical spin images lying on a spherical cap of geodesic radius θ_ε and pole located at A. The radius determines the minimum expected level of linear correlation ρ_ε for the potential matches, since $\rho_\varepsilon = \cos\theta_\varepsilon$; the smaller the radius the more correlated the model spherical spin images inside the cap will be to the scene query point. The spherical spin image labeled as B is an example of a matching point. Notice that the geodesic distance between two points on the unit sphere equals the angle between the unit vectors associated with the points.

4. Object Recognition with SSI

Recognizing and locating objects in complex 3-D scenes using spherical spin images consist of finding matches from a set of candidate correspondences between surface scene points and surface model points, that allow us to calculate a rigid transformation that best aligns both surfaces. Figure 4 illustrates the idea of finding correspondences for the scene point labeled A using the spherical-spin-image representation. Points lying inside the spherical cap are potential correspondences for the query point A, since the respective spherical spin images are similar (highly correlated). This means that finding correspondences consists of finding closest points to the query on a subset of S^{n-1}.

Finding closest spherical spin images is implemented by using a modified version of the closest-point search algorithm proposed in [11], which has complexity that is linear in the number of points for structured data [11]. In order to account for the noise present in the scene, our algorithm searches for a group of neighboring points located on the surface of the sphere within distance $\varepsilon = 2\sin(\cos^{-1}(\rho_\varepsilon)/2)$ from the query point; ρ_ϵ is a user-supplied parameter that determines the largest angle allowed between query vector and potential candidate vectors. Let ρ_{max} be the highest correlation coefficient of all neighbors in the group. The neighbors that have a correlation value above $\kappa\rho_{max}$ $(0 < \kappa < 1)$ are kept as the candidate correspondences for the query. Details of the geometry of our algorithm and its relationship to the approach presented in [11] are depicted in Figure 5.

The idea of searching for closest-points for matching is not new. It can be found in the work of Murase and Nayar [10] and Johnson [9]. Our contribution consists of redefining the geometry of the search space in order to utilize the correlation coefficient as a similarity measure in an efficient fashion.

Surface matching using spherical spin images proceeds by storing (offline) the model spherical spin images (unit vectors of dimension $n = 400$) in a closest point search structure and randomly selecting (online) a percentage of the points from the scene, which are compared individually to the model in two steps. First, a scene point's spherical spin image is created using the model's spherical spin image parameters and the scene data; and second, the scene point is used as a query point to the closest-point search algorithm described above, which returns a list of candidate correspondences for the query. The procedure is repeated for various points of the scene, in order to obtain a list of geometrically consistent correspondences, which generates

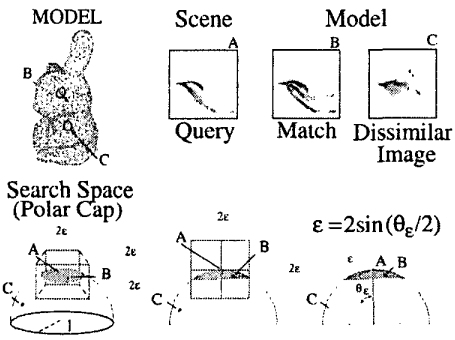

Figure 5. Geometric details of the search space (shaded area) used to find correspondences with the spherical-spin-image representation. The point labeled as B is considered a potential match to the query point A. Point C's spherical spin image is dissimilar to the query's point signature. The square box in the figure shows the geometry of the algorithm described in [11] for finding closest points. The geometry of our algorithm is related to the geometry of that algorithm by the geodesic radius θ_ϵ, since the side of the cube is set to $4\sin(\theta_\epsilon/2)$.

a set of potential matches that are scrutinized in a verification process resulting in rigid transformations that register the surfaces. The best transformation is selected as the one that generates the greatest overlap between surfaces.

5. Comparative Study

We developed an experimental protocol to compare the performance of four recognition and localization algorithms in complex 3-D scenes. They are 1) *standard spin images (SI)*, 2) *standard spin images with compression using PCA (SIC)*, 3) *spherical spin images (SSI)*, and 4)*spherical spin images with compression using random projection (SSIC)*. SI and SIC are Johnson's algorithms, while SSI and SSIC are our new algorithms.

A model library of 5 objects was used in the study. Models were constructed by registering and integrating multiple range views of the objects, and enforcing a uniform distribution of the vertices using the algorithm described in [8]. The spherical spin images of the models were created using the same generation parameters: bin size of 1 (mm), image width 20 and support angle 60°. The dimension of the uncompressed (compressed) tuples was set to 400 (40) respectively. Our test database consisted of 138 samples created by placing, without any systematic method, four or five models in the 3-D scene by hand. The resolution of all the surfaces in the study was set to 1 mm, ρ_ϵ was set to 0.96, and κ to 0.98.

For each of the 138 scenes and for each model present in the scene, the algorithms were executed one at a time. The resulting recognition state after running was classified as *true positive* if the model was recognized by the algorithm; as *false positive* if the algorithm placed the model in an incorrect position; and *false negative* if the model was not recognized. The *true negative* states did not apply in our experiments, since the models being sought in the scene were always present. For all the trials with true positive recognition state, the localization state was also determined. A localization state was classified as *true localization* if the position of the model in the scene was determined accurately (mean square error $mse < 0.01$); otherwise it was called *false localization*.

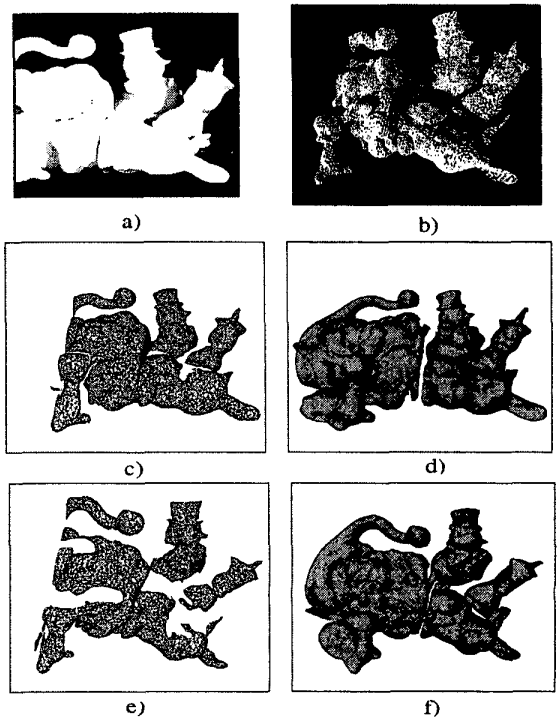

Figure 6. Recognition/localization result for five models in a typical 3-D scene. a) Color image. b) Three-dimensional raw data. c) Surface mesh of the 3-D scene, 21,340 vertices (front view). d) Three-dimensional scene with recognized models (front view). e) Surface mesh of the 3-D scene (side view). f) Three-dimensional scene with recognized models (side view). The five models, unicorn, bunny, deer, santa and snowman in the scene had varying levels of occlusion (75.54%, 65.7%, 73.6%, 58.6% and 72.21%, respectively) and varying levels of clutter (25.4%, 94.53%, 91.9%, 56.6% and 24.4%, respectively).

The effect of clutter and occlusion on recognition and localization, was measured by means of a graphical inter-

face. The occlusion of a model is defined as the *percent of the total area of the model that is not visible in the scene.* The clutter is defined as the *percentage of oriented points of the model, that are visible in the scene, whose spherical spin images have been corrupted by points in the scene that do not belong to the model.* The localization state was determined by calculating the average distance error between the model and the matched segment of the scene and also by visually inspecting the alignment. Figure 6 shows a qualitative result obtained with the SSI algorithm for a typical scene in our database in which the five models are present.

Data were analyzed by estimating the experimental recognition/localization rates as a function of the scene clutter and occlusion by using average shift histograms (ASHs) on the population of recognition (localization) states. This technique prevented problems inherent to the binning of the data [12]. Confidence intervals for the resulting rates were calculated for each algorithm by using standard bootstrap techniques [5]. More specifically, the population of recognition states for each algorithm was randomly sampled with replacement. The resulting sample was 75% the size of the original population. Then, the recognition (localization) rate versus occlusion (or clutter) was obtained from the sample by means of the ASH technique. The same procedure was repeated for 10,000 different samples. The mean and the standard deviation were then calculated from the samples to obtain the final empirical rate.

The performance of the algorithms SSI and SSIC is illustrated in Figure 7, which shows that: 1) The empirical recognition/localization rates versus occlusion are monotonically decreasing functions. 2) The recognition rate is about 90% for values of occlusion smaller than 70%, and the localization rate is about 95% for values of occlusion below 80%; this suggests that if about 30% of the area of the object is visible, it will be recognized and located with high probability. 3) The recognition/localization rates seem to be relatively uniform across the levels of clutter below 70%, showing that spherical spin images are robust to clutter. The variations observed are most likely related to the non-uniform sampling of the scene data. Johnson observed similar trends for the spin images in [9].

We used the paired t-test and the Wilcoxon matched pairs test analysis to detect significant differences in the matching times between the four algorithms tested, and significant correlation and linear regression analysis to study the relationship among the matching times of all four algorithms. Our proposed algorithms are faster than the other two as suggested by the results shown in Table 1 where the average matching times are statistically different with a significance level of $p = .0001$. The times were measured using a real-time clock on a dedicated Silicon Graphics O2 workstation (194MHz).

The recognition/localization performance of all four tested algorithms is illustrated in Figure 8. A summary of the improvements achieved with the proposed algorithms is shown in Table 2. The algorithms SSI and SSIC significantly improve the performance of the SI and SIC algorithms, respectively; our SSI and SSIC algorithms are faster and more accurate.

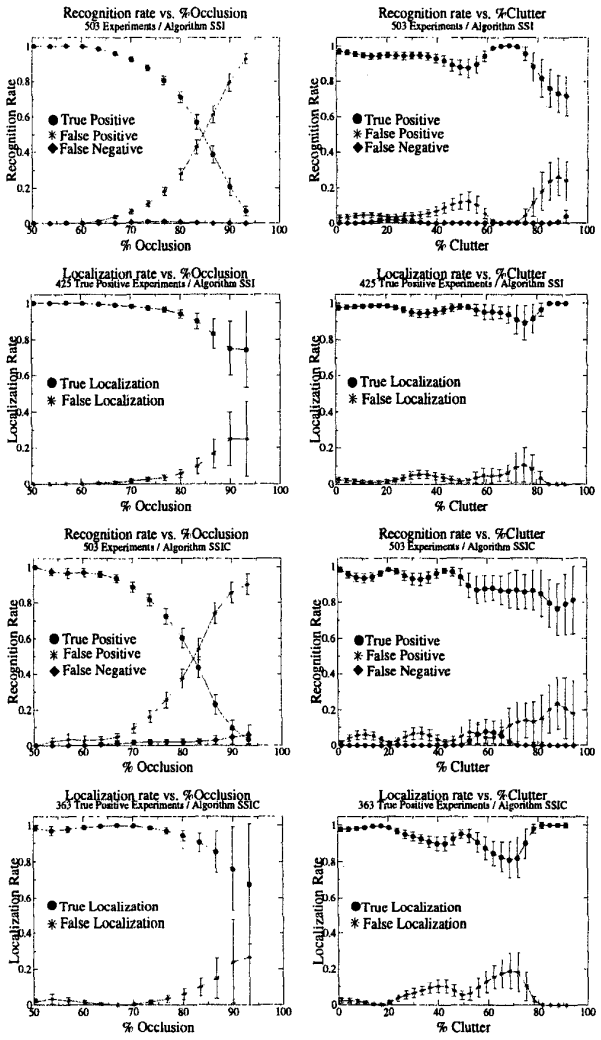

Figure 7. Recognition/localization rates for uncompressed spherical spin images (SSI) (four plots at the top), and compressed spherical spin images (SSIC) (four plots at the bottom). The left column illustrates the recognition rates versus occlusion, and the right column, versus clutter.

6. Summary and Conclusions

We have presented a general representation of shape that allows efficient and effective recognition and pose estima-

tion for 3-D objects in complex 3-D scenes. Spherical spin images are simple yet rich descriptors of shape that are very robust to clutter and occlusion, thus widely applicable in practical situations. We have shown that they can be compressed using a random projection method without significant loss of descriptive power and robustness. We have experimentally validated the effectiveness and efficiency of our representation in a comparative study that suggests that the overall performance of our proposed algorithms is better than the performance of the two original spin-image algorithms. Finally we note that the representation scheme in terms of equivalence classes can be adapted to any kind of signatures that can be discriminated by the linear correlation coefficient.

Algorithm	Matching Time
SSI	1879.1 ± 592.6
SI	7866.8 ± 3159.4
SSIC	1686 ± 700.4
SIC	2009.9 ± 758.5

Table 1. Average time per match in (ms).

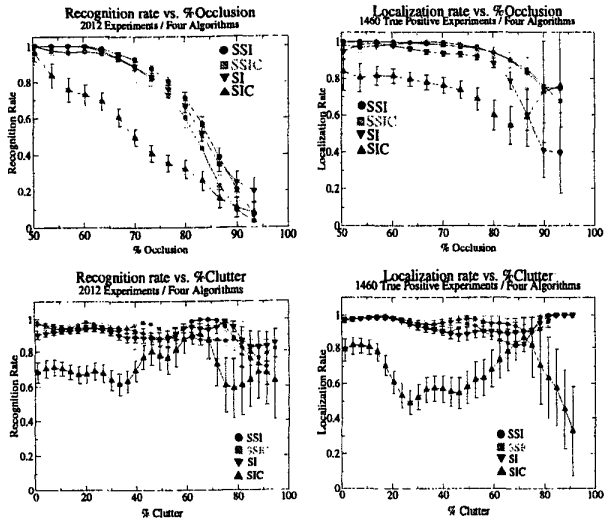

Figure 8. Recognition/localization rates versus occlusion (top) and clutter (bottom) for the algorithms tested.

Algorithm	RO	RC	LO	LC	T
SSI vs SI	3.36	4.24	4.74	6.68	76.11
SSIC vs SIC	32	25.41	22.2	33.13	16.12

Table 2. Average improvement (%) of the proposed algorithms with respect to the recognition rate (R) and localization rate (L) versus occlusion (O) and clutter (C); the matching time (T) improvement is also shown.

References

[1] P. J. Besl and R. Jain, "Range Image Understanding," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 430-449, 1985.

[2] C. S. Chua and R. Jarvis, "Point Signatures: A New Representation for 3-D Object Recognition," *International Journal of Computer Vision*, Vol. 25, No. 1, 1997, pp. 63-85.

[3] S. Dasgupta, "Experiments with Random Projection," *Uncertainity in Artificial Intelligence, UAI'2000*, 2000.

[4] C. Dorai and A. K. Jain, "COSMOS-A Representation Scheme for 3-D Free-Form Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10), October 1997, pp. 1115-1130.

[5] B. Efron, *The Jackknife, the Bootstrap, and other Resampling Plans*, Philadephia S.I.A.M., 1982.

[6] O. D. Faugeras and M. Hebert, "The Representation, Recognition, and Locating of 3-D Objects," *The International Journal of Robotics Research*, Vol. 5, No. 3, Fall 1986, pp. 27-52.

[7] B. W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz Mappings into a Hilbert Space," *Contemp. Math.*, 26, 1984, pp. 189-206.

[8] A. E. Johnson and M. Hebert, "Control of Polygonal Mesh resolution for 3-D Computer Vision," *Graphics, Modeling and Computer Vision*, 1998.

[9] A. E. Johnson and M. Hebert, "Using Spin Images for Efficient Object Recognition in Cluttered 3D scenes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(5), pp. 433-449, 1999.

[10] H. Murase and S. K. Nayar, "Visual Learning and Recognition of 3-D Objects from Appearance," *Int'l J. Computer Vision*, Vol. 14, pp. 5-24, 1995.

[11] S. A. Nene and S. K. Nayar, "A Simple Algorithm for Nearest Neighbor Search in High Dimensions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(9), pp. 999-1003, 1997.

[12] W. D. Scott, *Multivariate Density Estimation. Theory Practice and Visualization*, John Wiley and Sons, New York, 1992.

[13] F. Stein and G. Medioni, "Structural Indexing: Efficient 3-D Object Recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(2), 1992, pp. 125-145.

[14] S. M. Yamany, A. A. Fraag, A. El-Bialy, "Free-form surface registration and object recognition using surface signatures," *IEEE Int'l Conf. Computer Vision 1999*, Kerkyra, Grece, 1999.

[15] D. Zhang and M. Hebert, "Harmonic maps and their applications in surface matching," *IEEE Conference in Computer Vision and Pattern Recognition*, (CVPR'99), 1999.