published in proceedings *Forth International Conference on 3-D Digital Imaging and Modeling (3DIM 2003)*
6-10 October 2003, Banff, Alberta, Canada, IEEE Computer Society Press, pages 474-481

Surflet-Pair-Relation Histograms: A Statistical 3D-Shape Representation for Rapid Classification

Eric Wahl, Ulrich Hillenbrand, Gerd Hirzinger
German Aerospace Center (DLR)
Institute of Robotics and Mechatronics
Oberpfaffenhofen, 82234 Wessling, Germany

E-mail: eric.wahl@dlr.de

Abstract

A statistical representation of three-dimensional shapes is introduced, based on a novel four-dimensional feature. The feature parameterizes the intrinsic geometrical relation of an oriented surface-point pair. The set of all such features represents both local and global characteristics of the surface. We compress this set into a histogram. A database of histograms, one per object, is sampled in a training phase. During recognition, sensed surface data, as may be acquired by stereo vision, a laser range-scanner, etc., are processed and compared to the stored histograms. We evaluate the match quality by six different criteria that are commonly used in statistical settings. Experiments with artificial data containing varying levels of noise and occlusion of the objects show that Kullback-Leibler and likelihood matching yield robust recognition rates. The present study proposes histograms of the geometric relation between two oriented surface points (surflets) as a compact yet distinctive representation of arbitrary three-dimensional shapes.

1 Introduction

Robust scene interpretation by means of machine vision is a key factor in various new applications in robotics. Part of this problem is the efficient recognition and classification of previously known three-dimensional (3D) shapes in arbitrary scenes. So far, heavily constrained conditions have been utilized, or otherwise solutions have not been achieved in real time.

With the availability of ever faster computers and 3D-sensing technology (real-time stereo processing, laser range-scanner, etc.), more general approaches become feasible. They allow for weaker scene restrictions and hence facilitate new scenarios. Fundamental to visual object recognition are descriptions of general free-form shapes. A good overview of the currently prevalent approaches is given in [2].

In computer graphics, surface meshes are a popular description of free forms. They are also useful for recognition purposes and the Internet makes them accessible to everybody for testing and comparing algorithms. A major drawback, however, is their large memory requirement. Furthermore, surface meshes are defined with respect to a global coordinate system. Thus time consuming registration is necessary to align the object of interest to the frame of the referenced object model before matching is possible. The same problems apply to voxel-based descriptions of shape.

Representations based on superquadrics, generalized cylinders, and splines all suffer from a great sensibility to noise and outliers in the sensed data. A significant effort is required to obtain a robust fit procedure and to select the model order so as to avoid over-fitting.

It is, therefore, most desirable to develop a shape representation that (i) is compact, (ii) is robust, (iii) does not depend on a global coordinate frame, and (iv) has the descriptive capacity to distinguish arbitrary shapes.

A promising approach is to analyze the statistical occurrence of features on a surface in 3D space. This has been pursued by extracting local features such as surface curvatures or geometric relations such as distances. Their distributions are represented as discrete histograms or piecewise-linear approximations thereof. The classification step may be realized by matching a measured distribution against distributions in a reference database of prototypes or by the search for characteristic patterns in a distribution.

For instance, Osada et al. [8] sample the statistics of

point-pair distances across the whole surface of 3D objects. They demonstrate similarity search based on the distance distribution. However, a lot of information on shape is discarded by reduction to this one-dimensional feature. Vandeborre et al. [11] use three distributions of one-dimensional geometric features, based on curvature, distance, and volume. In both works, recognition performance is moderate and only suitable for a preliminary selection as performed, e.g., by an Internet search engine.

Hameiri and Shimshoni [3] look for symmetric form primitives, such as cylinders, cones, etc., in depth images. As the basic local feature, they use the two principle surface curvatures, accumulated in a two-dimensional histogram. The surface-curvature histogram is characteristic of each ideal form primitive and known a-priori from geometrical considerations. For real measured data, however, reliance upon curvatures is very sensitive to noise and artifacts. Moreover, for general shapes the distribution of curvatures will not be as crisp as for highly symmetric shapes, may hence be less informative, and many histograms may be required to cover all object views.

Multiple view-based histograms have been used by Hetzel et al. [4, 7] who adapted a probabilistic approach from Schiele and Crowley [9] to depth images. According to Bayes' rule, the best match is calculated as the one with the highest a-posterior probability, given a set of random feature samples. As feature they have employed a collection of local surface measures, namely, pixel depth, surface normal, and curvature. Generally, however, a high number of histograms per object model increases processing time.

An alternative line of research has sought to describe single, possibly characteristic points on an object by their local surface shape. This includes the *spin images* of Johnson and Hebert [6] and the *surface signatures* of Yamany and Farag [12]. For creating their histograms, surface points are picked and a plane is rotated about their local surface normal. The surrounding points are accumulated in that plane. Both approaches require dense surface meshes. Hillenbrand and Hirzinger [5] have characterized singular surface shape by *four-point-relation densities* that are directly constructed from a 3D-point set.

In this paper, we propose statistical analysis of a new four-dimensional geometric feature. The distribution of this feature captures both local and global aspects of shape. The relevant measures may be calculated from a surface mesh or be estimated from multiple 3D-data points. Here we rely on triangular meshes as the input data. We need just one stored histogram per object that is learned from training data. In the presence of significant noise or occlusion, we still obtain reasonable recognition rates above 80%. The processing time with a database containing 20 object models is around five milliseconds. The present study describes preliminary results that justify further research along this line.

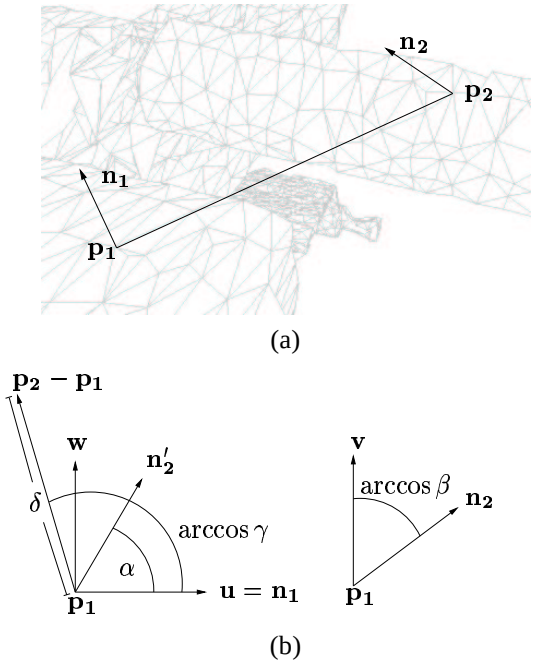


Figure 1. (a) Two surface points $\mathbf{p_1}, \mathbf{p_2}$ and their orientations $\mathbf{n_1}, \mathbf{n_2}$. (b) Illustration of the four parameters of our feature. The vector $\mathbf{n_2'}$ is the projection of $\mathbf{n_2}$ in the $\mathbf{uw}$-plane. α, $\arccos\beta$, and $\arccos\gamma$ are angles; δ is the length of the vector $\mathbf{p_2 - p_1}$.

The paper is organized as follows. Section 2 introduces the four-dimensional geometric feature. In Section 3, the sampling of histograms in the training phase is discussed. Section 4 defines six different criteria for comparison of sensed data with the trained histograms. We evaluate these criteria for classifiers in Section 5. Recognition rates and processing times are demonstrated for artifical data, and performance under conditions of noise and partial object visibility is investigated. Furthermore, we verify generalization of the classifiers across a wide range of mesh resolutions. The paper concludes in Section 6 with a final rating of the different classifiers and a prospect of future work.

2 Four-dimensional geometric feature

We now introduce a four-dimensional feature which is invariant to translation and rotation. The intrinsic geometrical relation between a pair of oriented surface points is parameterized, where an oriented point consists of its position and its local surface normal. In the following, oriented points are referred to as *surflets*.

Surflet-pair relations can be viewed as a generalization of curvatures. While curvatures measure geometric relations

between neighboring surflets, surflet-pair relations encode the same for any two surflets.

Each surflet is described by a pair $(\mathbf{p}, \mathbf{n})$, consisting of the position vector $\mathbf{p}$ and the surface normal $\mathbf{n}$. Positions and surface normals are here extracted from a triangular mesh, but may as well be estimated from multiple 3D-data points.

Let $\circ$ denote the scalar product of two vectors, $\times$ the cross product of two vectors, $||\cdot||$ the Euclidean norm of a vector, and $|\cdot|$ the modulus of a real number. For each pair of surflets $(\mathbf{p_1}, \mathbf{n_1})$ and $(\mathbf{p_2}, \mathbf{n_2})$, we define a coordinate system as follows. The origin is chosen to be $\mathbf{p_1}$, if

$$|\mathbf{n_1} \circ (\mathbf{p_2} - \mathbf{p_1})| \leq |\mathbf{n_2} \circ (\mathbf{p_2} - \mathbf{p_1})| \,, \tag{1}$$

and it is $\mathbf{p_2}$ else. Let us now assume that $\mathbf{p_1}$ is the origin. The base vectors of the coordinate system are then defined as

$$\mathbf{u} = \mathbf{n_1} \,, \tag{2}$$

$$\mathbf{v} = \frac{(\mathbf{p_2} - \mathbf{p_1}) \times \mathbf{u}}{\|(\mathbf{p_2} - \mathbf{p_1}) \times \mathbf{u}\|} \,, \tag{3}$$

$$\mathbf{w} = \mathbf{u} \times \mathbf{v} \,. \tag{4}$$

The relation between the surflets $(\mathbf{p_1}, \mathbf{n_1})$ and $(\mathbf{p_2}, \mathbf{n_2})$ is described by the parameters

$$\alpha = \arctan(\mathbf{w} \circ \mathbf{n_2}, \mathbf{u} \circ \mathbf{n_2}) \,, \tag{5}$$

$$\beta = \mathbf{v} \circ \mathbf{n_2} \,, \tag{6}$$

$$\gamma = \mathbf{u} \circ \frac{\mathbf{p_2} - \mathbf{p_1}}{\|\mathbf{p_2} - \mathbf{p_1}\|} \,, \tag{7}$$

$$\delta = \|\mathbf{p_2} - \mathbf{p_1}\| \,, \tag{8}$$

which define our feature $S = (\alpha, \beta, \gamma, \delta)$. Here we have used the shorthand notation

$$\arctan(x, y) = \begin{cases} \arctan(y/x) & \text{for } x > 0 \wedge y > 0, \\ \arctan(y/x) + \pi & \text{for } x < 0, \\ \arctan(y/x) + 2\pi & \text{for } x > 0 \wedge y < 0. \end{cases}$$

The attributes α and β represent $\mathbf{n_2}$ as an azimuthal angle and the cosine of a polar angle, respectively; γ and δ represent the direction and length of the translation from $\mathbf{p_1}$ to $\mathbf{p_2}$, respectively. This parameterization is illustrated in Figure 1. Of course, if Condition (1) determines $\mathbf{p_2}$ to be the origin, the parameters are obtained by interchanging the indices 1 and 2 in the equations above.

Equations (5)–(8) map every configuration of a surflet pair onto a unique set of parameters, and every possible set of parameters describes exactly one such configuration. Moreover, Condition (1) ensures that the base vectors $\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$ are defined in the most robust manner: by choosing the more orthogonal angle between $\mathbf{p_2} - \mathbf{p_1}$ and the two surface normals $\mathbf{n_1}$, $\mathbf{n_2}$ for defining $\mathbf{v}$ [cf. Equations (2) and (3)], the direction of $\mathbf{v}$ is determined with higher accuracy. From a surface with m surflets we obtain a total of $m(m-1)/2$ features.

3 Training phase

The four-dimensional feature distribution as sampled from a surface in 3D space is described by a histogram. Each feature S is mapped onto exactly one bin i of the histogram $H(i)$,

$$h : S \mapsto i \in \{1, 2, \ldots, d\} \,; \tag{9}$$

d is the number of bins in the histogram. The mapping $h(S)$ is defined by quantizing each of the four feature dimensions in five equal intervals. The resulting number of $d = 5^4 = 625$ bins for the complete histogram is both easy to handle and sufficient for classification. The length dimension δ [cf. Equation (8)] is normalized to the maximal occurring length Δ. An entry $H(i)$ of the histogram is the normalized frequency of features S that are mapped onto bin i,

$$H(i) = \frac{\mathrm{card}\{S \in \mathcal{S} | h(S) = i\}}{\mathrm{card}\mathcal{S}} \,, \tag{10}$$

where $\mathcal{S}$ is the set of all sampled features and card denotes the cardinality of a set.

When working with meshed surfaces, it is a good idea to collect for training all samples from multiple meshes of the same surface. In this way, we incorporate variations introduced by the mesh procedure.

The histogram $H(i)$ together with the maximal length Δ constitute an object model. The additional information of Δ is necessary for scaling at recognition time. We store a collection of such models in a database, one for each object we want to recognize.

4 Recognition phase

The goal of the recognition phase is to obtain feature distributions from sensed objects, to compare them with the model database, and to find the closest match. In order to avoid excessive computation time during recognition, we draw only a tiny subset (0.005%) of all available features $\mathcal{S}$.

Not the whole range of feature parameters $S = (\alpha, \beta, \gamma, \delta)$ is necessarily covered by every subsample, or even full sample of features from an object. Hence, some bins of a histogram may remain zero. This leads to numerical problems when computing divisions or logarithms. In such cases, all zero bins of a histogram are set to a common value, lower than the lowest non-zero value occurring in all histograms. This value has the effect of a penalty term.

In this section, H_O denotes the histogram of an object model O from the database. We define six different criteria that we will evaluate for their classification performance. Five of them are based on comparison of H_O to a histogram $H_{O'}$ that is built at recognition time from a test object O'. One implements the maximum-likelihood classifier.

4.1 Histogram-similarity criteria

We calculate the histogram $H_{O'}$ from the sensed subsample of features analogously to Section 3. The first criterion for comparison with a database histogram H_O is the intersection

$$\bigcap(H_O, H_{O'}) = \sum_{i=1}^{d} \min(H_O(i), H_{O'}(i)) \ , \qquad (11)$$

often used with fuzzy-set techniques and previously applied to color-histogram classification [10]. It is very fast to compute, because, apart from summation, no arithmetic operations are needed. Another straightforward criterion is the squared Euclidian distance

$$\mathcal{E}(H_O, H_{O'}) = \sum_{i=1}^{d} (H_O(i) - H_{O'}(i))^2 \ , \qquad (12)$$

which is known to be sensitive to noise and does not generalize very well. Next, the statistical χ^2-test is examined in its two forms

$$\chi_1^2(H_O, H_{O'}) = \sum_{i=1}^{d} \frac{(H_O(i) - H_{O'}(i))^2}{H_O(i)} \qquad (13)$$

and

$$\chi_2^2(H_O, H_{O'}) = \sum_{i=1}^{d} \frac{(H_O(i) - H_{O'}(i))^2}{H_O(i) + H_{O'}(i)} \ . \qquad (14)$$

Finally, we test the symmetric form of the Kullback-Leibler divergence

$$\mathcal{K}(H_O, H_{O'}) = \sum_{i=1}^{d} (H_{O'}(i) - H_O(i)) \ln \frac{H_{O'}(i)}{H_O(i)} \ . \qquad (15)$$

Because of the logarithmic operation, it is the computationally most expensive of all six criteria.

4.2 Likelihood criterion

Drawing a tiny, random subset of all features, we can safely assume individual samples to be statistically independent of each other. The logarithmic likelihood of object O, described by database histogram H_O, given the sensed subsample $\mathcal{S}_{O'}$ of features, thus is

$$\mathcal{L}(O|\mathcal{S}_{O'}) = \sum_{S \in \mathcal{S}_{O'}} \ln H_O(h(S)) \ . \qquad (16)$$

The mapping $h(S)$ is as defined in Equation (9). In contrast to the Kullback-Leibler divergence (15), all logarithms can here be calculated in the training phase and logarithmic histograms $\ln H_O$ can be stored.

Table 1. In this test, the six classifiers defined in Section 4 are evaluated using randomly drawn feature samples from complete and noise-free surface meshes of the 20 objects shown in Figure 2. Achieved recognition rates are given in percent. The processing times are measured on a standard PC with an Intel Pentium IV 2.66 GHz processor and Linux as operating system.

criterion	recognition in %	time in ms
$\bigcap$	42.7	5.12
$\mathcal{E}$	40.6	5.01
χ_1^2	75.4	6.16
χ_2^2	45.5	6.25
$\mathcal{K}$	99.6	7.42
$\mathcal{L}$	99.7	4.79

5 Experiments

All experiments are based upon the 20 objects shown in Figure 2. The objects are initially given as surface meshes, which are, however, unrelated to the meshes we use as inputs to our algorithm. To ensure that classification cannot be dominated by object size, all objects are scaled to a common maximal diameter.

Models are trained by the following procedure. For each object, five sets of points, from 25,000 to 389,000 points per set, are drawn randomly from the surface and passed to a mesh generator. A training mesh consists of between 3,500 and 5,500 vertices. Features are built from pairs of surflets, which are in turn picked from each vertex. All features obtained from the five training meshes, that is, between 30,616,250 and 75,611,250 features, are collected into a histogram [cf. Section 3].

In the recognition phase, new meshes are generated from each object. Features are randomly subsampled from the vertices of these meshes. The number of features drawn is 0.005% of all available features. This arbitrary, low sampling rate turns out to be high enough for good recognition. Results presented on classification rate and timing are averaged over between 100 and 1000 meshes per test object.

5.1 Ideal conditions

Under ideal conditions, the test objects' surfaces are completely exposed to the sensor and sensed data are free of noise. Table 1 shows the achieved recognition rates and times for the six criteria [cf. Section 4]. The measured times include all steps from drawing feature samples to the output of the best matching object model. Generating the sur-

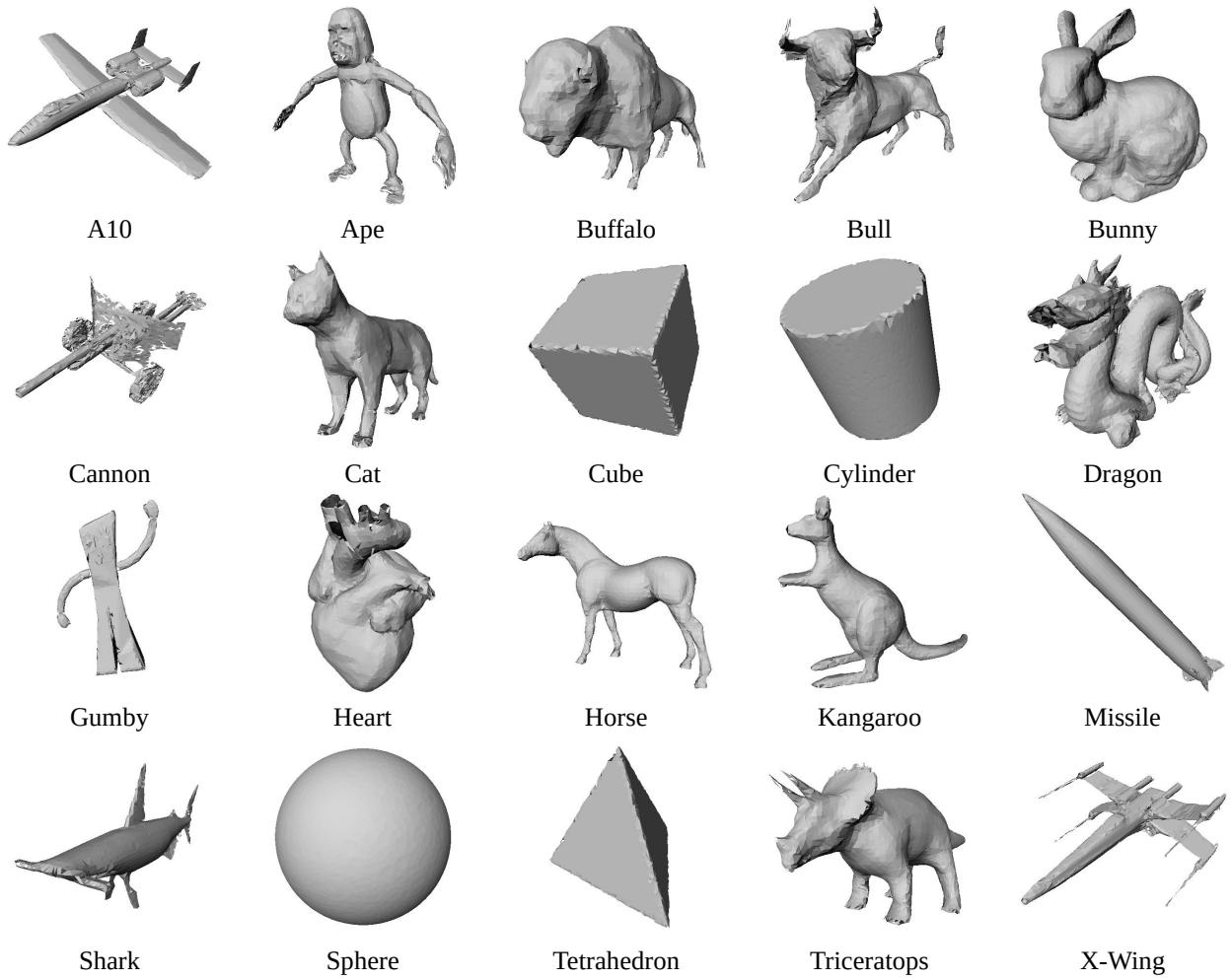


Figure 2. The 20 objects of the database.

face mesh is not included. Almost perfect classification has been achieved by the $\mathcal{K}$ and $\mathcal{L}$ criteria. Interestingly, the χ^2_2 criterion performs dramatically weaker than the χ^2_1 criterion. Apparently, the weighting of histogram differences by the reciprocal of the trained histogram value alone is much more reliable than taking also the estimate from the small test sample into account [cf. Equations (13), (14)].

Correct classification and confusion rates between all pairs of objects are shown in Figure 3. All classifiers work well for simple shapes like cube or sphere. Interestingly, the objects that are difficult to classify differ drastically across the criteria. On the other hand, the $\cap$ and χ^2_2 criteria exhibit a strikingly similar pattern of classification performance. This similarity will also be retained in all the other tests of the classifiers we report below. The same similarity holds for the best, the $\mathcal{K}$ and $\mathcal{L}$ criteria.

5.2 Noisy data

If the point cloud is obtained from real sensors like laser range-scanners, laser profilers, or stereo cameras, the data will be corrupted in various ways. Therefore, in a second set of experiments, sensitivity of the feature histograms to noise is evaluated. Uniformly distributed noise is simulated by randomly translating vertices from a surface mesh inward or outward along the local surface normal. The level of noise is defined as the range of translations, measured in percent of the maximal object diameter[1]. As an example, Figure 4(b) shows a surface mesh corrupted by the maximal level of noise we have tested (20%).

In Figure 5(a), we present plots of recognition rates for the six classifiers as a function of noise level. For the $\mathcal{E}$, χ^2_1, $\mathcal{K}$, and $\mathcal{L}$ criteria, classification performance degrades rapidly with increasing noise. This is explained by the fact

[1]Remember that the diameter was scaled to the same value for all objects.

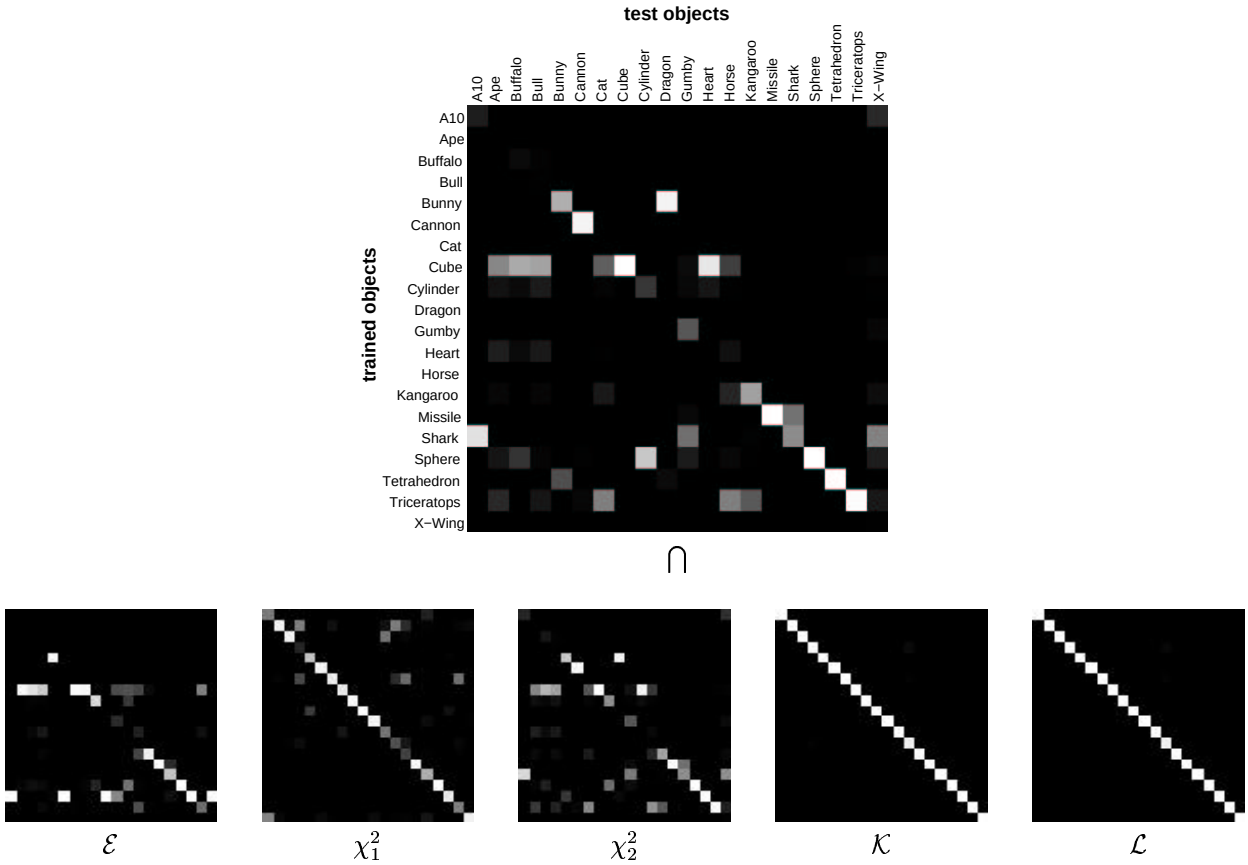
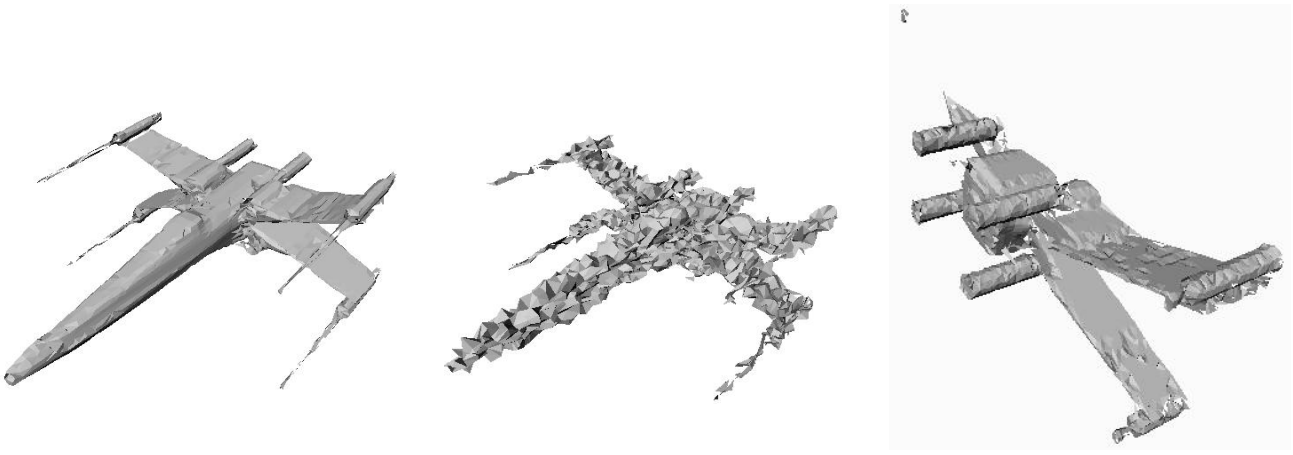


Figure 3. The six arrays represent classification results for the 20 objects shown in Figure 2 using the six different criteria defined in Section 4. Surfaces are completely visible and data are noise free. In each array, columns represent test objects, rows trained objects. Grey values indicate the rate of classification of a test object as a trained object; a brighter shade means a higher rate. The more distinct the diagonal, the higher the allover performance of the classifier. Evidently, the $\mathcal{K}$ and $\mathcal{L}$ criteria achieve almost perfect classification within our database of objects.



(a) (b) (c)

Figure 4. (a) X-wing; (b) X-wing with noise (4%); (c) partially visible X-wing (33%).

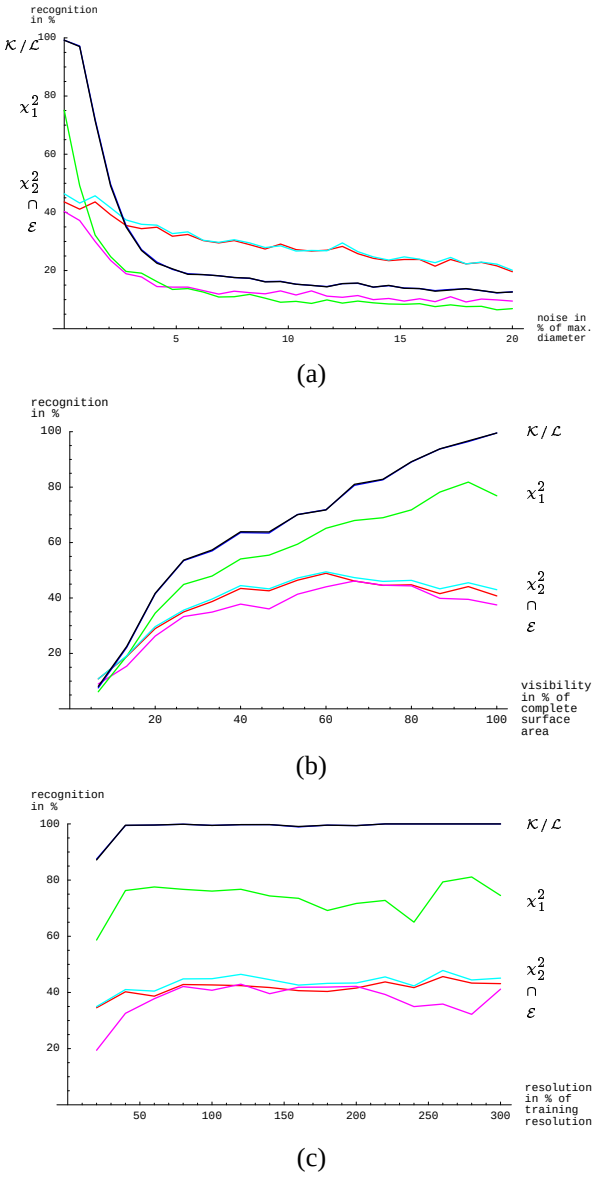


Figure 5. Plots of recognition rates for the 20 objects shown in Figure 2 using the six different criteria defined in Section 4. The conditions for the test data are varied; (a) varying level of noise (in percent of maximal object diameter); (b) varying visibility (in percent of complete surface area); (c) varying mesh resolution (in percent of training resolution). The curves for the $\mathcal{K}$ and $\mathcal{L}$ criteria nearly coincide in all three graphs.

that the angular attributes α, β, γ are very sensitive to noise such that surface information is largely lost. Interestingly, these criteria reach a rather stable rate of between 10% and 15% correct classification. Some residual performance may be expected, as the distance attribute δ remains informative up to much higher noise levels. The χ_2^2 and $\cap$ criteria, on the other hand, are a lot less sensitive to noise, exhibiting significantly lower performance at low noise and higher performance at high noise levels. Under realistic conditions of measurement (noise $< 1\%$), however, the $\mathcal{K}$ and $\mathcal{L}$ criteria yield a reasonable recognition rate above 80%.

5.3 Partial visibility

In real applications, objects to be recognized are often just partially visible. Reasons are self-occlusion in single-view data or occlusions by other objects. Partial objects yield incomplete surface meshes. Therefore, in this set of experiments, each test object is meshed and classified with varying fraction of visible surface. Visible parts are determined by intersecting point clouds by a random plane. Subsequently, data on one side of the plane are processed by the mesh generator. Visibility is defined as the sum of remaining triangle areas in percent of the complete surface area. Figure 4(c) gives an example of a partially visible mesh (33%).

Results on recognition rates for various visibilities are plotted in Figure 5(b). Performance can be seen to drop off more gradually with occlusion than with data corruption by noise [cf. Figure 5(a)]. Correct classification by the $\mathcal{K}$ and $\mathcal{L}$ criteria remains above 80% down to roughly 65% visibility.

We note that recognition with partial visibility depends in fact heavily on the particular section of the object that remains visible.

5.4 Generalization across mesh resolution

Since we have relied upon surface meshes as the input representation, it is interesting to ask how recognition performance is affected by changes to the mesh procedure. The most demanding scenario is *generalization* across mesh procedures, that is, being confronted at recognition time with a mesh of a type essentially different from what training has been based on.

In a final set of experiments, we thus have investigated the effect of varying the mesh resolution for the test objects. Figure 5(c) shows plots of correct-classification rates under such conditions, where mesh resolution is given in percent of the (constant) resolution in the training phase. Apparently, recognition performance does not critically depend on test-mesh resolution. Only below 50% of the training resolution, recognition performance drops off. In part, this can be ascribed to the low absolute number of feature samples drawn. In particular, the $\mathcal{K}$ and $\mathcal{L}$ criteria exhibit a high degree of generalization across meshes.

6 Conclusion

In this paper, we have introduced a novel four-dimensional feature that describes the intrinsic geometrical relation between a pair of surflets, i.e., oriented surface points in 3D space. The statistical distribution of this feature as sampled from an object's surface captures both local and global aspects of shape. Empirically learned histograms of the feature distribution have here been demonstrated as a compact and efficient representation of arbitrary 3D shapes. This representation allows for rapid classification of shapes based on a single histogram per object model, independent of translation and rotation.

We have evaluated six different criteria for the shape classifier. The Kullback-Leibler and likelihood criteria have been found to perform equally well and superior to the others. They have shown nearly perfect classification under ideal sensing conditions and robust performance in the face of noise and occlusion. They are, moreover, largely independent of the resolution used for meshing surfaces of test objects. Considering its lower computational cost, we recommend using the maximum-likelihood classifier.

More specifically, the experiments clearly indicate that, for best performance, high noise during recognition should be reduced by spatial averaging, at the cost of a lower mesh resolution.

The more invariant a classifier, the less can be recovered from an act of classification. The present classifiers are by design invariant to object pose. Especially for robotic applications, however, it would be most desirable to obtain an estimate of the pose of an object, along with its identity. One direction of future research will hence be to augment the algorithm by a method for locating an object's data within a larger set that may comprise multiple objects. This would imply segmentation of the data into the objects' components.

To more firmly establish the potential of the proposed representation of shape, the dependence of classification performance on various design parameters, like feature quantization and subsampling rate, has to be investigated. Moreover, the database of objects will be extended in the future.

When moving to real data, we plan to use the DLR laser range-scanner [1] or some sort of stereo processing to acquire 3D-point clouds from a scene as a first processing step.

Apart from scene analysis, potential applications of the present shape classifiers include similarity search in a database of 3D object-models, e.g., on the Internet. In this context, normalization of the model dimensions will make the classifiers invariant to object scale.

References

[1] http://www.robotic.dlr.de/mechatronics/scanner/.

[2] R. J. Campbell and P. J. Flynn. A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding*, 81:166–210, 2001.

[3] E. Hameiri and I. Shimshoni. Using principal curvatures and darboux frame to recover 3D-geometric primitives form range images. *3D Data Processing Visualization Transmission (3DPVT'02)*, pages 656–663, June 2002.

[4] G. Hetzel, B. Leibe, P. Levi, and B. Schiele. 3D object recognition from range images using local feature histograms. *International Conference on Computer Vision and Pattern Recognition*, December 2001.

[5] U. Hillenbrand and G. Hirzinger. Probabilistic search for object segmentation and recognition. In *Proceedings ECCV 2002*, volume 2352 of *Lecture Notes in Computer Science*, pages 791–806. Springer, 2002.

[6] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433–449, May 1999.

[7] B. Leibe, G. Hetzel, and P. Levi. Local feature histograms for object recogntition from range images. Technical report, University of Stuttgart, 2001.

[8] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Transcations on Graphics*, 21(4):807–832, October 2002.

[9] B. Schiele and J. L. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–52, 2000.

[10] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[11] J.-P. Vandeborre, V. Couillet, and M. Daoudi. A practical aproach for 3D model indexing by combining local and global invariants. *3D Data Processing Visualization Transmission (3DPVT'02)*, pages 644–647, June 2002.

[12] S. M. Yamany and A. A. Farag. Surface signatures: An orientation indepentent free-form surface representation scheme for the purpose of objects registration and matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1105–1120, 2002.