Recognizing Objects in Range Data Using Regional Point Descriptors

Andrea Frome[1], Daniel Huber[2], Ravi Kolluri[1],
Thomas Bülow[1⋆], and Jitendra Malik[1]

[1] University of California Berkeley, Berkeley CA 94530, USA,
{afrome,rkolluri,malik}@cs.berkeley.edu
thomas.buelow@philips.com
[2] Carnegie Mellon University, Pittsburgh PA 15213, USA
dhuber@cs.cmu.edu

Abstract. Recognition of three dimensional (3D) objects in noisy and cluttered scenes is a challenging problem in 3D computer vision. One approach that has been successful in past research is the regional shape descriptor. In this paper, we introduce two new regional shape descriptors: 3D shape contexts and harmonic shape contexts. We evaluate the performance of these descriptors on the task of recognizing vehicles in range scans of scenes using a database of 56 cars. We compare the two novel descriptors to an existing descriptor, the spin image, showing that the shape context based descriptors have a higher recognition rate on noisy scenes and that 3D shape contexts outperform the others on cluttered scenes.

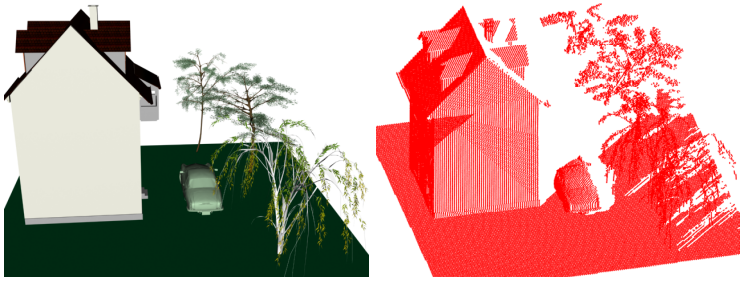
1 Introduction

Recognition of three dimensional (3D) objects in noisy and cluttered scenes is a challenging problem in 3D computer vision. Given a 3D point cloud produced by a range scanner observing a 3D scene (Fig. 1), the goal is to identify objects in the scene (in this case, vehicles) by comparing them to a set of candidate objects. This problem is challenging for several reasons. First, in range scans, much of the target object is obscured due to self-occlusion or is occluded by other objects. Nearby objects can also act as background clutter, which can interfere with the recognition process. Second, many classes of objects, for example the vehicles in our experiments, are very similar in shape and size. Third, range scanners have limited spatial resolution; the surface is only sampled at discrete points, and fine detail in the objects is usually lost or blurred. Finally, high-speed range scanners (e.g., flash ladars) introduce significant noise in the range measurement, making it nearly impossible to manually identify objects.

Object recognition in such a setting is interesting in its own right, but would also be useful in applications such as scan registration [9][6] and robot localization. The ability to recognize objects in 2 1/2-D images such as range scans

⋆ Current affiliation is with Philips Research Laboratories, Roentgenstrasse 24-26, 22335 Hamburg

T. Pajdla and J. Matas (Eds.): ECCV 2004, LNCS 3023, pp. 224–237, 2004.



(a) (b)

Fig. 1. (a) An example of a cluttered scene containing trees, a house, the ground, and a vehicle to be recognized. (b) A point cloud generated from a scan simulation of the scene. Notice that the range shadow of the building occludes the front half of the vehicle.

may also prove valuable in recognizing objects in 2D images when some depth information can be inferred from cues such as shading or motion.

Many approaches to 3D object recognition have been put forth, including generalized cylinders [3], superquadrics [7], geons [23], medial axis representations [1], skeletons [4], shape distributions [19], and spherical harmonic representations of global shape [8]. Many of these methods require that the target be segmented from the background, which makes them difficult to apply to real-world 3D scenes. Furthermore, many global methods have difficulty leveraging subtle shape variations, especially with large parts of the shape missing from the scene. At the other end of the spectrum, purely local descriptors, such as surface curvature, are well-known for being unstable when faced with noisy data. Regional point descriptors lie midway between the global and local approaches, giving them the advantages of both. This is the approach that we follow in this paper.

Methods which use regional point descriptors have proven successful in the context of image-based recognition [17][15][2] as well as 3D recognition and surface matching [22][13][5][21]. A regional point descriptor characterizes some property of the scene in a local support region surrounding a basis point. In our case, the descriptors characterize regional surface shape. Ideally, a descriptor should be invariant to transformations of the target object (e.g., rotation and translation in 3D) and robust to noise and clutter. The descriptor for a basis point located on the target object in the scene will, therefore, be similar to the descriptor for the corresponding point on a model of the target object. These model descriptors can be stored in a pre-computed database and accessed using fast nearest-neighbor search methods such as locality-sensitive hashing [11]. The limited support region of descriptors makes them robust to significant levels of occlusion. Reliable recognition is made possible by combining the results from multiple basis points distributed across the scene.

In this paper we make the following contributions: (1) we develop the 3D generalization of the 2D shape context descriptor, (2) we introduce the harmonic shape context descriptor, (3) we systematically compare the performance of the 3D shape context, harmonic shape context, and spin images in recognizing sim-

ilar objects in scenes with noise or clutter. We also briefly examine the trade-off of applying hashing techniques to speed search over a large set of objects.

The organization of the paper is as follows: in section 2, we introduce the 3D shape context and harmonic shape context descriptors and review the spin image descriptor. Section 3 describes the representative descriptor method for aggregating distances between point descriptors to give an overall matching score between a query scene and model. Our data set is introduced in section 4, and our experiments and results are presented in section 5. We finish in section 6 with a brief analysis of a method for speeding our matching process.

2 Descriptors

In this section, we provide the details of the new 3D shape context and harmonic shape context descriptors and review the existing spin-image descriptor. All three descriptors take as input a point cloud $\mathcal{P}$ and a basis point p, and capture the regional shape of the scene at p using the distribution of points in a support region surrounding p. The support region is discretized into bins, and a histogram is formed by counting the number of points falling within each bin. For the 3D shape contexts and spin-images, this histogram is used directly as the descriptor, while with harmonic shape contexts, an additional transformation is applied.

When designing such a 3D descriptor, the first two decisions to be made are (1) what is the shape of the support region and (2) how to map the bins in 3D space to positions in the histogram vector. All three methods address the second issue by aligning the support region's "up" or north pole direction with an estimate of the surface normal at the basis point, which leaves a degree of freedom along the azimuth. Their differences arise from the shape of their support region and how they remove this degree of freedom.

2.1 3D Shape Contexts

The 3D shape context is the straightforward extension of 2D shape contexts, introduced by Belongie et al. [2], to three dimensions. The support region for a 3D shape context is a sphere centered on the basis point p and its north pole oriented with the surface normal estimate $\mathcal{N}$ for p (Fig. 2). The support region is divided into bins by equally spaced boundaries in the azimuth and elevation dimensions and logarithmically spaced boundaries along the radial dimension. We denote the $J+1$ radial divisions by $R = \{R_0 \dots R_J\}$, the $K+1$ elevation divisions by $\Theta = \{\Theta_0 \dots \Theta_K\}$, and the $L+1$ azimuth divisions by $\Phi = \{\Phi_0 \dots \Phi_L\}$. Each bin corresponds to one element in the $J \times K \times L$ feature vector. The first radius division R_0 is the minimum radius $r_{\min}$, and R_J is the maximum radius $r_{\max}$. The radius boundaries are calculated as

$$R_j = \exp\left\{\ln(r_{\min}) + \frac{j}{J}\ln\left(\frac{r_{\max}}{r_{\min}}\right)\right\}. \quad (1)$$

Sampling logarithmically makes the descriptor more robust to distortions in shape with distance from the basis point. Bins closer to the center are smaller in all three spherical dimensions, so we use a minimum radius ($r_{\min} > 0$) to avoid being overly sensitive to small differences in shape very close to the center. The Θ and Φ divisions are evenly spaced along the 180° and 360° elevation and azimuth ranges.

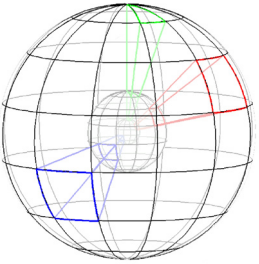


Fig. 2. Visualization of the histogram bins of the 3D shape context.

Bin(j, k, l) accumulates a weighted count $w(p_i)$ for each point p_i whose spherical coordinates relative to p fall within the radius interval $[R_j, R_{j+1})$, azimuth interval $[\Phi_k, \Phi_{k+1})$ and elevation interval $[\Theta_l, \Theta_{l+1})$. The contribution to the bin count for point p_i is given by

$$w(p_i) = \frac{1}{\rho_i \sqrt[3]{V(j,k,l)}} \tag{2}$$

where $V(j, k, l)$ is the volume of the bin and ρ_i is the local point density around the bin. Normalizing by the bin volume compensates for the large variation in bin sizes with radius and elevation. We found empirically that using the cube root of the volume retains significant discriminative power while leaving the descriptor robust to noise which causes points to cross over bin boundaries. The local point density ρ_i is estimated as the count of points in a sphere of radius δ around p_i. This normalization accounts for variations in sampling density due to the angle of the surface or distance to the scanner.

We have a degree of freedom in the azimuth direction that we must remove in order to compare shape contexts calculated in different coordinate systems. To account for this, we choose some direction to be Φ_0 in an initial shape context, and then rotate the shape context about its north pole into L positions, such that each Φ_l division is located at the original 0° position in one of the rotations. For descriptor data sets derived from our reference scans, L rotations for each basis point are included, whereas in the query data sets, we include only one position per basis point.

2.2 Harmonic Shape Contexts

To compute harmonic shape contexts, we begin with the histogram described above for 3D shape contexts, but we use the bin values as samples to calculate a spherical harmonic transformation for the shells and discard the original histogram. The descriptor is a vector of the amplitudes of the transformation, which are rotationally invariant in the azimuth direction, thus removing the degree of freedom.

Any real function $f(\theta, \phi)$ can be expressed as a sum of complex spherical harmonic basis functions Y_l^m.

$$f(\theta,\phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{m=l} A_l^m Y_l^m(\theta,\phi) \tag{3}$$

A key property of this harmonic transformation is that a rotation in the azimuthal direction results in a phase shift in the frequency domain, and hence amplitudes of the harmonic coefficients $\|A_l^m\|$ are invariant to rotations in the azimuth direction. We translate a 3D shape context into a harmonic shape context by defining a function $f_j(\theta,\phi)$ based on the bins of the 3D shape context in a single spherical shell $R_j \leq R < R_{j+1}$ as:

$$f_j(\theta,\phi) = SC(j,k,l), \theta_k < \theta \leq \theta_{k+1},\ \phi_l < \phi \leq \phi_{l+1}. \tag{4}$$

As in [14], we choose a bandwidth b and store only b lowest-frequency components of the harmonic representation in our descriptor, which is given by $HSC(l,m,k) = \|A_{l,k}^m\|,\ l,m = 0 \ldots b, r = 0 \ldots K$. For any real function, $\|A_l^m\| = \|A_l^{-m}\|$, so we drop the coefficients A_l^m for $m < 0$. The dimensionality of the resulting harmonic shape context is $K \cdot b(b+1)/2$. Note that the number of azimuth and elevation divisions do not affect the dimensionality of the descriptor.

Harmonic shape contexts are related to the rotation-invariant shape descriptors $\mathrm{SH}(f)$ described in [14]. One difference between those and the harmonic shape contexts is that one $\mathrm{SH}(f)$ descriptor is used to describe the global shape of a single object. Also, the shape descriptor $\mathrm{SH}(f)$ is a vector of length b whose components are the energies of the function f in the b lowest frequencies: $\mathrm{SH}_l(f) = \|\sum_{m=-l}^{l} A_l^m Y_l^m\|$. In contrast, harmonic shape contexts retain the amplitudes of the individual frequency components, and, as a result, are more descriptive.

2.3 Spin Images

We compared the performance of both of these shape context-based descriptors to spin images [13]. Spin-images are well-known 3D shape descriptors that have proven useful for object recognition [13], classification [20], and modeling [10]. Although spin-images were originally defined for surfaces, the adaptation to point clouds is straightforward. The support region of a spin image at a basis point p is a cylinder of radius $r_{\max}$ and height h centered on p with its axis aligned with the surface normal at p. The support region is divided linearly into J segments radially and K segments vertically, forming a set of $J \times K$ rings. The spin-image for a basis point p is computed by counting the points that fall within each ring, forming a 2D histogram. As with the other descriptors, the contribution of each point q_i is weighted by the inverse of that point's density estimate (ρ_i); however, the bins are not weighted by volume. Summing within each ring eliminates the degree of freedom along the azimuth, making spin-images rotationally invariant. We treat a spin-image as a $J \times K$ dimensional feature vector.

3 Using Point Descriptors for Recognition

To compare two descriptors of the same type to one another, we use some measure of distance between the feature vectors: ℓ_2 distance for 3D shape contexts and spin images, and the inverse of the normalized correlation for harmonic shape contexts. Given a query scene $\mathcal{S}_q$ and a set of reference descriptors calculated from scans of known models, we would like to choose the known model which is most similar to an object in $\mathcal{S}_q$. After we calculate descriptors from $\mathcal{S}_q$ and distances between the query descriptors and reference descriptors, we face the problem of how to aggregate these distances to make a choice as to which model is the best match to $\mathcal{S}_q$.

A straightforward way of doing this would be to have every descriptor from $\mathcal{S}_q$ vote for the model that gave the closest descriptor, and choose the model with the most votes as the best match. The problem is that in placing a hard vote, we discard the relative distances between descriptors which provide information about the quality of the matches. To remedy this, we use the representative shape context method introduced in Mori et al. [18], which we refer to as the *representative descriptor method*, since we also apply it to spin images.

3.1 Representative Descriptor Method

We precompute M descriptors at points $p_1, ...p_M$ for each reference scan $\mathcal{S}_i$, and compute at query time K descriptors at points $q_1, ...q_K$ from the query scene $\mathcal{S}_q$, where $K \ll M$. We call these K points *representative descriptors* (RDs). For each of the query points q_k and each reference scan $\mathcal{S}_i$, we find the descriptor p_m computed from $\mathcal{S}_i$ that has the smallest ℓ_2 distance to q_k. We then sum the distances found for each q_k, and call this the *representative descriptor cost* of matching $\mathcal{S}_q$ to $\mathcal{S}_i$:

$$\text{cost}(\mathcal{S}_q, \mathcal{S}_i) = \sum_{k \in \{1,...,K\}} \min_{m \in \{1,...,M\}} \text{dist}(q_k, p_m) \tag{5}$$

The best match is the reference model $\mathcal{S}$ that minimizes this cost.

Scoring matches solely on the representative descriptor costs can be thought of as a lower bound on an ideal cost measure that takes geometric constraints between points into account. We show empirically that recognition performance using just these costs is remarkably good even without a more sophisticated analysis of the matches.

One could select the center points for the representative descriptors using some criteria, for example by picking out points near which the 3D structure is interesting. For purposes of this paper, we sidestep that question altogether and choose our basis points randomly. To be sure that we are representing the performance of the algorithm, we performed each representative descriptor experiment 100 times with different random subsets of basis points. For each run we get a recognition rate that is the percentage of the 56 query scenes that we correctly identified using the above method. The mean recognition rate is the recognition rate averaged across runs.

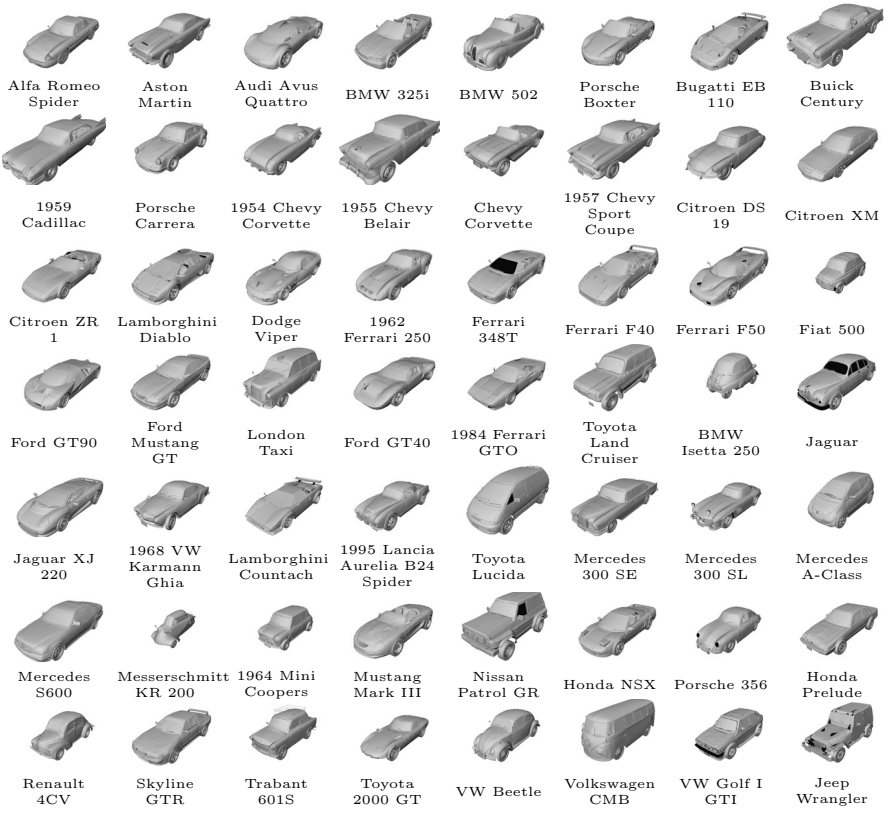


Fig. 3. The 56 car models used in our experiments shown to scale.

4 Data Set

We tested our descriptors on a set of 56 3D models of passenger vehicles taken from the De Espona 3D model library [12] and rescaled to their actual sizes (Fig. 3).[1] The point clouds used in our experiments were generated using a laser sensor simulator that emulates a non-commercial airborne range scanner system. We have shown in separate experiments that these descriptors work well for real data, but for these experiments, our goal was to compare the performance of the descriptors in controlled circumstances.

We generated two types of point clouds: a set of model or "reference" scans, and several sets of scene or "query" scans. For each vehicle, we generated four reference scans with the sensor positioned at 90° azimuth intervals ($\phi = 45°$, 135°, 225°, and 315°), a 45° declination angle, and a range of 500 m from the

[1] The Princeton Shape Benchmark, a data set with 1,814 3D models, was recently released. We didn't learn of the data set in time to use it in this paper, but we will be using it in future experiments. It can be found online at http://shape.cs.princeton.edu/benchmark/.

target. The resulting point clouds contained an average of 1,990 target points spaced approximately 6 cm apart. The query scans were generated in a similar manner, except that the declination was 30° and the azimuth was at least 15° different from the nearest reference scan. Depending on the experiment, either clutter and occlusion or noise was added. Clutter and occlusion were generated by placing the model in a test scene consisting of a building, overhanging trees, and a ground plane (Fig. 1(a)). The point clouds for these scenes contained an average of 60,650 points. Noisy scans were modeled by adding Gaussian noise ($\mathcal{N}(0, \sigma)$) along the line of sight of each point.

Fig. 4. The top row shows scans from the 1962 Ferrari 250 model, and the bottom scans are from the Dodge Viper. The scans in column (a) are the query scans at 30° elevation and 15° azimuth with $\sigma = 5$ cm noise, and those in (b) are from the same angle but with $\sigma = 10$ cm noise. With 10 cm noise, it is difficult to differentiate the vehicles by looking at the 2D images of the point clouds. Column (c) shows the reference scans closest in viewing direction to the query scans (45° azimuth and 45°elevation). In the 5 cm and 10 cm noise experiments, we first chose 300 candidate basis points and sampled RDs from those.

Basis points for the descriptors in the reference point clouds were selected using a method that ensures approximately uniform sampling over the model's visible surface. Each point cloud was divided into 0.2-meter voxels and one point was selected at random from each occupied voxel, giving an average of 373 descriptors per point cloud (1,494 descriptors per model). Basis points in the query point clouds were chosen using the same method, except that the set was further reduced by selecting a random subset of N basis points (N=300 for the clutter-free queries and N=2000 for the clutter queries) from which representative descriptors were chosen. For a given experiment, the same subset of basis points were used in generating the three types of descriptors. After noise and clutter were added, normals for the basis points were computed using a method which preserves discontinuities in the shape and that accounts for noise along the viewing direction [16]. The algorithm uses points within a cube-shaped window around the basis point for the estimation, where the size of the window can be chosen based on the expected noise level.

5 Experiments

The parameters for the descriptors (Table 1) were chosen based on extensive experimentation on other sets of 3D models not used in these experiments (Table 1). However, some parameters (specifically K and $r_{\min}$) were fine-tuned using descriptors in 20 randomly selected models from our 56 vehicle database. The basis points used for training were independent from those used in testing. The relative scale of the support regions was chosen to make the volume encompassed comparable across descriptors.

Table 1. Parameters used in the experiments for shape contexts (SC), harmonic shape contexts (HSC), and spin images (SI). All distances are in meters

	SC	HSC	SI
max radius ($r_{\max}$)	2.5	2.5	2.5
min radius ($r_{\min}$)	0.1	0.1	-
height (h)	-	-	2.5
radial divisions (J)	15	15	15
elev./ht. divisions (K)	11	11	15
azimuth divisions (L)	12	12	-
bandwidth (b)	-	16	-
dimensions	1980	2040	225
density radius (δ)	0.2	0.2	0.2

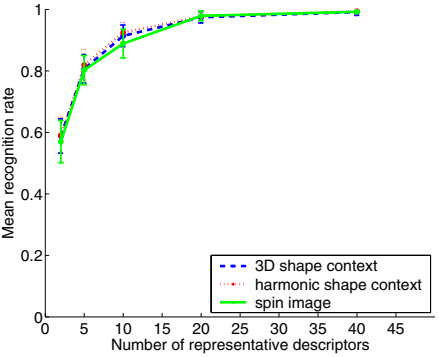


Fig. 5. Results for the 5cm noise experiment. All three methods performed roughly equally. From 300 basis points sampled evenly from the surface, we chose varying numbers of RDs, and recorded the mean recognition rate. The error bars show one standard deviation.

5.1 Scenes with 5 cm Noise

In this set of experiments, our query data was a set of 56 scans, each containing one of the car models. We added Gaussian noise to the query scans along the scan viewing direction with a standard deviation of 5 cm (Fig. 4). The window for computing normals was a cube 55 cm on a side. Fig. 5 shows the mean recognition rate versus number of RDs. All of the descriptors perform roughly equally, achieving close to 100% average recognition with 40 RDs.

5.2 Scenes with 10 cm Noise

We performed two experiments with the standard deviation increased to 10 cm (see Fig. 4). In the first experiment, our window size for computing normals was the same as in the 5 cm experiments. The results in Fig. 6 show a significant decrease in performance by all three descriptors, especially spin images. To test how much the normals contributed to the decrease in recognition, we performed

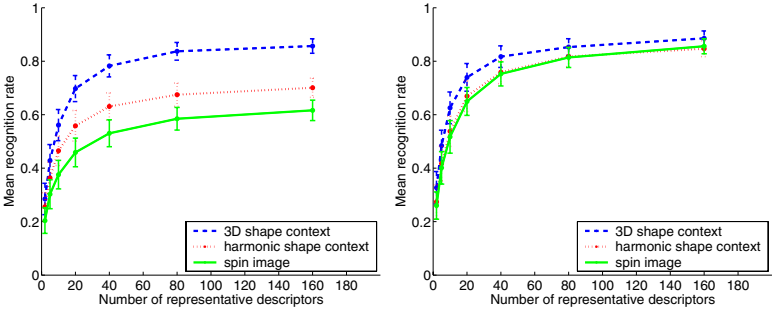


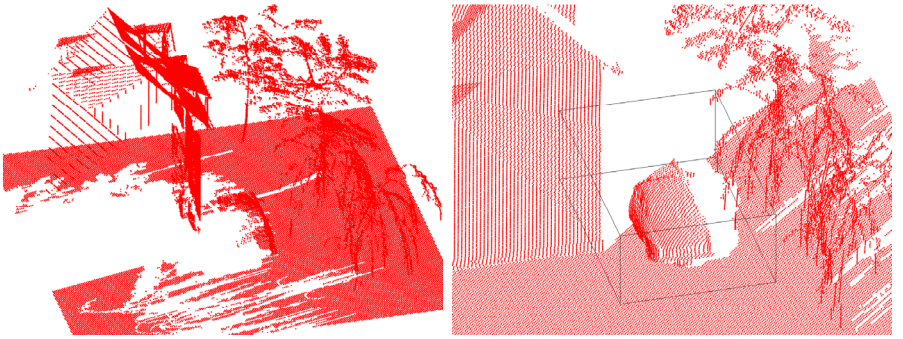
Fig. 6. Results for 10 cm noise experiments. In experiment (a) we used a window for the normals that was a cube 55 cm on a side, whereas in (b) the size was increased to a cube 105 cm on a side. The error bars show one standard deviation from the mean. From this experiment, we see that shape contexts degrade less as we add noise and in particular are less sensitive to the quality of the normals than spin images. All three methods would benefit from tuning their parameters to the higher noise case, but this would entail a recalculation of the reference set. In general, a method that is more robust to changes in query conditions is preferable.

a second experiment with a normal estimation window size of 105 cm, giving us normals more robust to noise. The spin images showed the most improvement, indicating their performance is more sensitive to the quality of the normals.

5.3 Cluttered Scenes

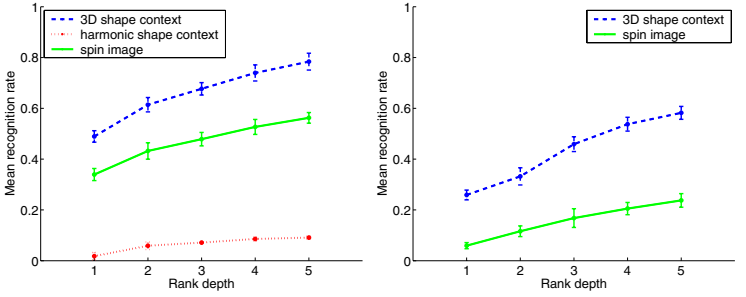
To test the ability of the descriptors to handle a query scene containing substantial clutter, we created scenes by placing each of the vehicle models in the clutter scene shown in Fig. 1(a). We generated scans of each scene from a 30° declination and two different azimuth angles ($\phi = 150°$ and $\phi = 300°$), which we will call views #1 and #2 (Fig. 7). We assume that the approximate location of the target model is given in the form of a box-shaped volume of interest (VOI). The VOI could be determined automatically by a generic object saliency algorithm, but for the controlled experiments in this paper, we manually specified the VOI to be a 2 m × 4 m × 6 m volume that contains the vehicle as well as some clutter, including the ground plane (Fig. 7(b)). Basis points for the descriptors were chosen from within this VOI, but for a given basis point, all the scene points within the descriptor's support region were used, including those outside of the VOI.

We ran separate experiments for views 1 and 2, using 80 RDs for each run. When calculating the representative descriptor cost for a given scene-model pair, we included in the sum in equation (5) only the 40 smallest distances between RDs and the reference descriptors for a given model. This acts as a form of outlier rejection, filtering out many of the basis points not located on the vehicle. We chose 40 because approximately half of the basis points in each VOI fell on a vehicle. The results are shown in Fig. 8.



(a) (b)

Fig. 7. The cluttered scene with the Karmann Ghia. Picture (a) is the scan from view 2, and (b) is a close-up of the VOI in view 1. For the fully-rendered scene and the full scan from view 1, refer to Fig. 1. The scanner in view 1 was located on the other side of the building from the car, causing the hood of the car to be mostly occluded. In view 2, the scanner was on the other side of the trees, so the branches occlude large parts of the vehicle. There were about 100 basis points in the VOI in each query scene, and from those we randomly chose 80 representative descriptors for each run.



(a) (b)

Fig. 8. Cluttered scene results. In both, we included in the cost the 40 smallest distances out of those calculated for 80 RDs. The graphs show recognition rate versus rank depth with error bars one standard deviation from the mean. We calculated the recognition rate based on the k best choices, where k is our *rank depth* (as opposed to considering only the best choice for each query scene). We computed the mean recognition rate as described before, but counted a match to a query scene as "correct" if the correct model was within the top k matches. Graph (a) shows the results for view #1 and (b) for view #2. Using the 3D shape context we identifying on average 78% of the 56 models correctly using the top 5 choices for each scene, but only 49% of the models if we look at only the top choice for each. Spin images did not perform as well; considering the top 5 matches, spin images achieved a mean recognition rate of 56% and only 34% if only the top choice is considered. Harmonic shape contexts do particularly bad, achieving recognition slightly above chance. They chose the largest vehicles as matches to almost all the queries.

The shape context performance is impressive given that this is a result of doing naïve point-to-point matching without taking geometric constraints into account. Points on the ground plane were routinely confused for some of the car models which geometric constraints could rule out. A benefit of the 3D shape context over the other two descriptors is that a point-to-point match gives a candidate orientation of the model in the scene which can be used to verify other point matches.

6 Speeding Search with Locality-Sensitive Hashing

In this section, we briefly explore the cost of using 3D shape contexts and discuss a way to bring the amount of computation required for a 3D shape context query closer to what is used for spin images while maintaining accuracy.

In the spin image and harmonic shape context experiments, we are comparing each of our representative descriptors to 83,640 reference descriptors. We must compare to the 12 rotations when using 3D shape contexts, giving a total of 1,003,680. Our system implementation takes 7.4 seconds on a 2.2 GHz processor to perform the comparison of one 3D shape context to the reference set.

Fast search techniques such as locality-sensitive hashing (LSH) [11] can reduce the search space by orders of magnitude, making it more practical to search over the 3D shape context rotations, though there is a tradeoff between speed and accuracy of the nearest-neighbor result. The method divides the high-dimensional feature space where the descriptors lie into hypercubes, divided by a set of k randomly-chosen axis-parallel hyperplanes. These define a hash function where points that lie in the same hypercube hash to the same value. The greater the number of planes, the more likely that two neighbors will have different hash values. The probability that two nearby points are separated is reduced by independently choosing l different sets of hyperplanes, thus defining l different hash functions. Given a query vector, the result is the set of hashed vectors which share one of their l hash values with the query vector.

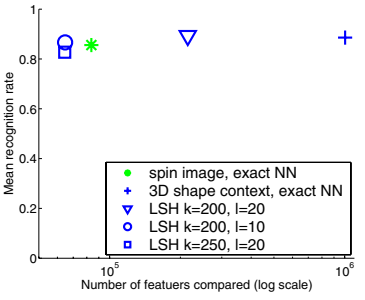


Fig. 9. Results for LSH experiments with 3D shape contexts on the 10cm noise query dataset using the 105 cm window size. Shown are results using 160 RDs where we included the 80 smallest distances in the RD sum. The exact nearest neighbor results for spin images and 3D shape contexts are shown for comparison.

In Figure 9, we show LSH results on the 10cm noise dataset with the 105 cm window size using 160 RDs (exact nearest neighbor results are shown in Figure 6(b)). We chose this data set because it was the most challenging of the noise tests where spin images performed well (using an easier test such as the 5 cm noise experiment provides a greater reduction in the number of comparisons). In calculating the RD costs, the distance from a query point

to a given model for which there were no hash matches was set to a value larger than any of the other distances. In this way, we penalized for a failure to match any hashed descriptors. To remove outliers caused by unlucky hash divisions, we included in the sum in equation (5) only the 80 smallest distances between RDs and the returned reference descriptors. Note that performing LSH using 3D shape contexts with $k = 200$ hash divisions and $l = 10$ hash functions requires fewer descriptor comparisons than an exact nearest neighbor search using spin images, and provides slightly better accuracy.

Acknowledgements. We would like to thank Bogdan Matei at Sarnoff Corporation for use of his normal calculation code and technical support. Thanks also to Anuj Kapuria and Raghu Donamukkala at Carnegie Mellon University, who helped write the spin image code used for our experiments. This work was supported in part by the DARPA E3D program (F33615-02-C-1265) and NSF ITR grant IIS-00-85864.

References

1. E. Bardinet, S. F. Vidal, Arroyo S. D., Malandain G., and N. P. de la Blanca Capilla. Structural object matching. Technical Report DECSAI-000303, University of Granada, Dept. of Computer Science and AI, Granada, Spain, February 2000.
2. S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002.
3. T. O. Binford. Visual perception by computer. Presented at IEEE Conference on Systems and Control, Miami, FL, 1971.
4. Bloomenthal and C. Lim. Skeletal methods of shape manipulation. In *International Conference on Shape Modeling and Applications*, pages 44–47, 1999.
5. Chin Seng Chua and Ray Jarvis. Point signatures: a new representation for 3D object recognition. *International Journal of Computer Vision*, 25(1):63–85, Oct 1997.
6. D.Zhang and M.Herbert. Experimental analysis of harmonic shape images. In *Proceedings of Second International Conference on 3-D Digital Imaging and Modeling*, pages 191–200, October 1999.
7. Solina F. and Bajcsy R. Recovery of parametric models from range images: The case for superquadrics with global deformations. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, February 1990.
8. Thomas Funkhouser, Patrick Min, Michael Kazhdan, Joyce Chen, Alex Halderman, David Dobkin, and David Jacobs. A search engine for 3d models. *ACM Transactions on Graphics*, 22:83–105, January 2003.
9. G.Roth. Registering two overlapping range images. In *Proceedings of Second International Conference on 3-D Digital Imaging and Modeling*, pages 191–200, October 1999.
10. Daniel F. Huber and Martial Hebert. Fully automatic registration of multiple 3D data sets. *Img. and Vis. Comp.*, 21(7):637–650, July 2003.

11. P. Indyk and R. Motwani. Approximate nearest neighbor - towards removing the curse of dimensionality. In *Proceedings of the 30th Symposium on Theory of Computing*, 1998.
12. De Espona Infografica. *De Espona 3D Models Enciclopedia*. http://www.deespona.com/3denciclopedia/menu.html.
13. Andrew E. Johnson and Martial Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(5):433–449, 1999.
14. Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. Rotation invariant spherical harmonic representation of 3d shape descriptors. In *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 156–164. Eurographics Association, 2003.
15. D. Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1000–1015, Sep 1999.
16. Bogdan Matei and Peter Meer. A general method for errors-in-variables problems in computer vision. In *CVPR*, volume 2, June 2000.
17. K. Mikolajczk and C. Schmid. A performance evaluation of local descriptors. In *CVPR*, volume II, pages 257–263, Jun 2003.
18. G. Mori, S. Belongie, and J. Malik. Shape contexts enable efficient retrieval of similar shapes. In *CVPR*, volume 1, pages 723–730, 2001.
19. R. Osada, T. Funkhouser, B. Chayelle, and D. Dobkin. Matching 3d models with shape distributions. In *Shape Modeling International*, May 2001.
20. Salvador Ruiz-Correa, Linda Shapiro, and Marina Miela. A new paradigm for recognizing 3d object shapes from range data. In *ICCV*, Oct 2003.
21. Fridtjof Stein and Gerard Medioni. Structural indexing: efficient 3D object recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(2):125–45, Feb 1992.
22. Y. Sun and M.A. Abidi. Surface matching by 3d point's fingerprint. In *ICCV*, pages 263–9, Jul 2001.
23. K. Wu and Levine M. Recovering parametrics geons from multiview range data. In *CVPR*, June 1994.