Spatially sensitive statistical shape analysis for pedestrian recognition from LIDAR data

Michalis A. Savelonas[a,*], Ioannis Pratikakis[a], Theoharis Theoharis[b], Georgios Thanellas[a], Frédéric Abad[c], Rémy Bendahan[c]

[a] *Dept. of Electrical and Computer Engineering, Democritus University of Thrace, Greece*
[b] *Dept. of Computer and Information Science, Norwegian University of Science and Technology, Norway*
[c] *IMRA Europe S.A.S., France*

Abstract

Range-based pedestrian recognition is instrumental towards the development of autonomous driving and driving assistance systems. This work introduces encoding methods for pedestrian recognition, based on statistical shape analysis of 3D LIDAR data. The proposed approach has two variants, based on the encoding of local shape descriptors either in a spatially agnostic or spatially sensitive fashion. The latter method derives more detailed cues, by enriching the 'gross' information reflected by overall statistics of local shape descriptors, with 'fine-grained' information reflected by statistics associated with spatial clusters. Experiments on artificial LIDAR datasets, which include challenging samples, as well as on a large scale dataset of real LIDAR data, lead to the conclusion that both variants of the proposed approach (i) obtain high recognition accuracy, (ii) are robust against low-resolution sampling, (iii) are robust against increasing distance, and (iv) are robust against non-standard shapes and poses. On the other hand, the spatially-sensitive variant is more robust against partial occlusion and bad clustering.

Keywords: Local shape descriptors, Fisher encoding, LIDAR, pedestrian recognition

*Corresponding author
Email address: msavelonas@gmail.com (Michalis A. Savelonas)

1. Introduction

Range data are essential for pedestrian recognition, bringing the opportunity to identify higher level patterns, beyond image gradients. Despite this fact, the interest on range-based pedestrian recognition has only recently been considerable, since robust depth inference from monocular optical cameras is a difficult problem. Yet, low cost 3D sensors, such as Microsoft Kinect, are now available, whereas Light Detection and Ranging (LIDAR) sensors, such as the Velodyne HDL-64E, emerge as the primary range-based technology for pedestrian recognition. Although LIDAR-generated point clouds are rather sparse, LIDAR sensors are more reliable than 3D sensors on outdoor settings. In addition, they have a maximum range exceeding 50 m, as opposed to the 4 m range of a 3D sensor such as Kinect.

Considering the sparsity of LIDAR-generated point clouds, which limits the descriptive capability of local shape information, related research could be directed towards encoding methods for the statistical analysis of local shapes, in order to identify patterns beyond the local scale. In addition, the employed encoding methods should cope with problems associated with pedestrian recognition, such as partial occlusion, bad clustering, as well as non-standard shapes and poses. The bag-of-visual-words (BoVW) framework appears as a suitable encoding candidate, having been successfully applied for 3D shape analysis in various settings [1],[2].

This work introduces encoding methods for pedestrian recognition, based on statistical shape analysis of 3D LIDAR data. The proposed approach has two variants based on the encoding of local shape descriptors either in a spatially agnostic or spatially sensitive fashion. The latter method derives more detailed cues, by enriching the 'gross' information reflected by overall statistics of local shape descriptors, with 'fine-grained' information reflected by statistics associated with spatial clusters. The recognition accuracy obtained is evaluated on artificial LIDAR datasets, which include challenging samples addressing occlusion, bad clustering, non-standard shapes and poses, as well as on a large

scale dataset of real LIDAR data. As will be shown in the related work section, current research on LIDAR-based pedestrian recognition has not fully considered state-of-the-art in local shape descriptors, as well as in effective encoding schemes, as the ones introduced here.

The remainder of this paper is organized as follows: Section 2 presents related previous work in pedestrian recognition, after introducing and motivating the topic. Section 3 provides technical background on local shape analysis and Fisher encoding. Section 4 describes the proposed encoding methods and Section 5 presents the experimental evaluation of the proposed methods on datasets of artificial and real LIDAR data. Finally, Section 6 presents the main conclusions of this work.

2. Related work

Kidono et al. [3] introduced a LIDAR-based pedestrian recognition method, which combines the slice feature and the distribution of the reflection intensities in a standard SVM-based classification scheme. Promising recognition results were obtained on a road-environment dataset created by the authors. Teichman et al. [4] utilized time series of point clouds, as well as intensities. A related work of Chen et al. [5] obtains bounding boxes by means of inference in a Markov random field encoding object size priors, ground plane and a variety of depth informed features. Their application on the RGB-D version of the KITTI dataset leads to state-of-the-art recognition accuracy. Du et al. [6] used local-global articulated human parts and defined part-specific features. This method relies on heuristic approaches to identify upper human body and legs. The recognition accuracy obtained was higher than the one obtained by the method of Kidono et al., yet on different datasets. A limitation of this method is that it is tailored to human body, which could hardly allow generalization to non-standard shapes, as is the cases with pedestrians carrying an object (e.g. a bag, an umbrella etc).

There are also hybrid pedestrian recognition methods, based on both image

and LIDAR data. Premebida et al. [7] showed that the two modalities are complementary. Similar conclusions where derived in the work of Gonzalez et al. [8]. These works where based on image-based feature vectors combining standard image descriptors such as HOG [9] and LBP [10], with depth maps generated from LIDAR data.

It could be noted that although LIDAR data have been proved as valuable for the recognition task, related research, either for standalone LIDAR-based methods or for hybrid methods, is relatively limited. More importantly, this research is mostly based on global shape descriptors, such as object size, which cannot reflect the complexity and variability of all pedestrian and non-pedestrian point clouds. State-of-the-art local shape descriptors, which provide detailed shape representations, such as the Fast Point Feature Histograms (FPFH) [11], Spin Images (SI) [12] and Signatures of Histograms of OrienTations (SHOT) [13], have not been employed. Along this direction, an encoding scheme for effectively addressing statistical properties of such local shape descriptors, should be investigated.

3. Background

This section provides the background for the various elements of the proposed recognition schemes, in terms of local shape descriptors, encoding and classification.

3.1. Local shape descriptors

We use Fast Point Feature Histogram (FPFH), which has been introduced by Rusu et al. as an effective and more efficient variant of the previously proposed Point Feature Histogram (PFH) [14]. Both PFH and FPFH had been initially used for point cloud registration but several works have demonstrated their applicability in the context of various tests, which include shape retrieval and recognition [15]. FPFH relies upon geometrical relations between k nearest neighbours, derived from 3D point coordinates (x,y,z) and estimated surface

normals (*nx*,*ny*,*nz*). FPFH can be defined as follows: (i) for each point p, all pairs of points formed by p and each point p_i in the r-neighborhood of p are considered, (ii) the normals n and n_i are estimated by PCA, (iii) a Darboux *uvn* frame ($u = n$, $v = (p - p_i) \times u$, $n = u \times v$) is defined. The angular variations of n and n_i are computed as follows: $\alpha = u \cdot n_i$, $\phi = u \cdot (p_i - p)/||p_i - p||$, $\theta = \arctan(w \cdot n_i, u \cdot n_i)$, (iv) for each pair of points, a single point feature histogram (SPFH) is estimated. The histogram has b binning subdivisions for each one of α, ϕ and θ angle, where b is implementation-dependent. This leads to a histogram size equal to $3b$. Finally, for each point p, an FPFH is calculated as a weighted sum of all SPFH's associated to pairs (p, p_i), where the weight for each pair depends on the distance between its points.

Apart from FPFH, we will also investigate the use of Spin Images (SI) [12], which are among the most popular local 3D shape descriptors and have been widely applied on both structured and unstructured data. An SI of an oriented point is a 2D representation of its surrounding surface, constructed on a pose-invariant 2D coordinate system by accumulating the coordinates of neighboring points. The SI is invariant to rigid transformations, since it encodes the coordinates of points on the surface of an object with respect to a local basis. An important drawback of SI is the large number of histogram bins involved, which is equal to 153 and affects the efficiency of SI-based recognition methods.

Finally, we will investigate the use of Signatures of Histograms of OrienTations (SHOT). SHOT employs a local reference frame and a 3D descriptor which represents both the histograms of normal angles and their spatial distributions. The latter is a hybrid structure between signatures and histograms, aiming at a more favorable balance between descriptive power and robustness. In [13], the authors have shown that SHOT outperforms point signatures and SI.

3.2. Encoding

The Bag-of-Visual-Words (BoVW) framework provides a tool for deriving global statistics from local shape descriptors and has already been successfully employed for 3D shape analysis [1],[2]. Fisher encoding employs a codebook

formed by Gaussian Mixture Models (GMMs) instead of k-means [16]. The mean of a Gaussian fit is subtracted from all observations and the resulting differences comprise the Fisher vector, which has been shown to provide a generalized, enhanced version of a variant of k-means-based BoVW: the vector of locally aggregated descriptors. As demonstrated by Jegou et al. [16], Fisher vector tends to reflect information which is distinctive for each sample. Moreover, Fisher encoding requires much more compact codebooks and has been associated with enhanced recognition accuracy.

Given a training set of N local shape descriptors $\mathbf{x}_1, ..., \mathbf{x}_N \in R^D$, a GMM $p(\mathbf{x}|\theta)$ is the probability density on R^D provided by

$$p(\mathbf{x}|\theta) = \sum_{k=1}^{K} p(\mathbf{x}|\mu_k, \Sigma_k)\pi_k \tag{1}$$

$$p(\mathbf{x}|\mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^D det\Sigma_k}} e^{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \Sigma_k^{-1}(\mathbf{x}-\mu_k)} \tag{2}$$

where K is the number of Gaussian components used, θ is the vector of model parameters $(\pi_1, \mu_1, \Sigma_1, ..., \pi_K, \mu_K, \Sigma_K)$, including the prior probability values $\pi_k \in R_+$ (which sum to one), the means $\mu_k \in R^D$, and the positive definite covariance matrices $\Sigma_k \in R^{D\times D}$ of each Gaussian component. The covariance matrices are assumed to be diagonal, so that the GMM is fully specified by $(2D+1)K$ scalar parameters. Soft data-to-cluster assignments extend the binary assignments to k-means in basic BoVW and can be defined as

$$q_{ki} = \frac{p(\mathbf{x}_i|\mu_k, \Sigma_k)\pi_k}{\sum_{j=1}^{K} p(\mathbf{x}_i|\mu_j, \Sigma_j)\pi_j}, k = 1, ..., K \tag{3}$$

Fisher encoding captures the average first and second order differences between the local descriptors and the GMM centroids. For the k-th GMM, where $k = 1, ..., K$, the following vectors are defined

$$\mathbf{u_k} = \frac{1}{N\sqrt{\pi_k}} \sum_{i=1}^{N} q_{ik} \Sigma_k^{-1/2}(\mathbf{x}_i - \mu_k) \tag{4}$$

$$\mathbf{v_k} = \frac{1}{N\sqrt{2\pi_k}} \sum_{i=1}^{N} q_{ik}[(\mathbf{x}_i - \mu_k)\Sigma_k^{-1}(\mathbf{x}_i - \mu_k) - 1] \tag{5}$$

In $\mathbf{u_k}$ and $\mathbf{v_k}$, the approximate location of the descriptors in each region is encoded, relatively to the mean and the variance, respectively. The division by $\sqrt{2\pi_k}$ can be interpreted as a BoVW inverse document frequency term: the weights of frequent descriptors are reduced [16].

The Fisher encoding of the set of local feature vectors is then given by the concatenation of $\mathbf{u_k}$ and $\mathbf{v_k}$ for all K components, giving an encoding of size $2DK$

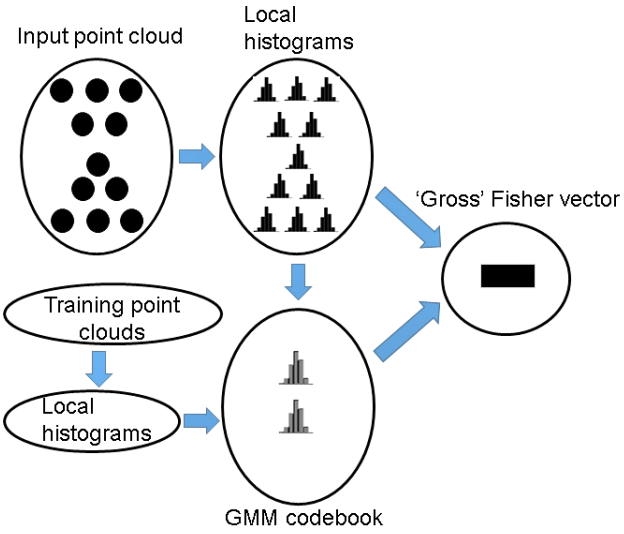
$$\mathbf{f} = [\mathbf{u}_1^T, \mathbf{v}_1^T, ..., \mathbf{u}_k^T, \mathbf{v}_k^T, ...\mathbf{u}_K^T, \mathbf{v}_K^T] \tag{6}$$

4. Proposed recognition approach

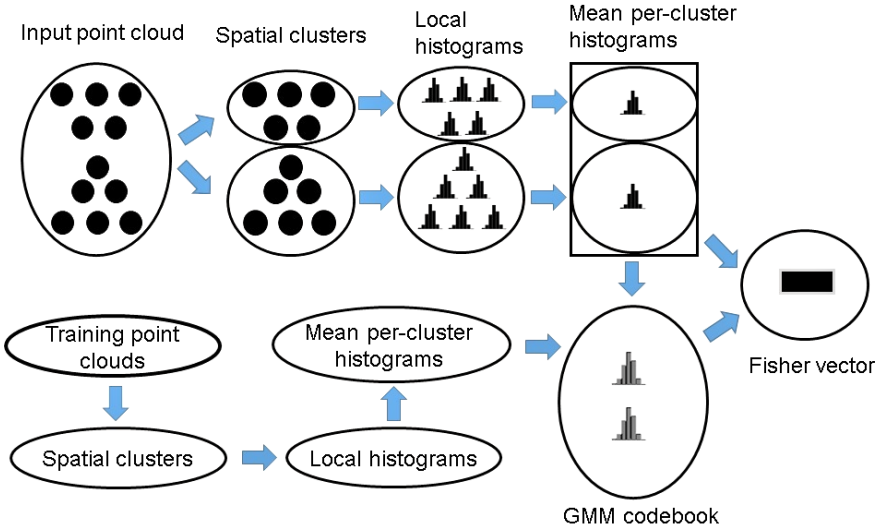
This section presents our approach for pedestrian recognition based on local shape geometry, with two methods employing either spatially agnostic or spatially Fisher sensitive encoding. The spatially agnostic method serves also as an introductory step to formulate the more sophisticated spatially sensitive method.

4.1. Spatially agnostic encoding of local shape geometry

The first encoding method applies Fisher encoding of histogram-based local shape descriptors in a spatially agnostic fashion, with a codebook of GMMs learned from local 'visual words', as is the case with standard BoVW. The resulting Fisher vector reflects 'gross', global statistics of local shape geometry and can be used for classification. Beyond standard pedestrian queries, such an approach can cope with non-standard poses, since in such cases local neighborhoods and the derived statistics of local shape descriptors remain unaffected. In this work we employed standard Support Vector Machines (SVMs) and k Nearest Neighbors (k-NN) classifiers. In the text to follow, we will refer to this method as Spatially Agnostic Fisher Encoding (SAFE). Figure 1 (top) presents a schematic overview of SAFE.



Spatially agnostic encoding of local shape geometry



Spatially sensitive encoding of local shape geometry

Figure 1: Schematic overviews of the two proposed recognition methods.

4.2. Spatially sensitive encoding of local shape geometry

Although SAFE is capable of correctly recognizing standard pedestrian queries, as well as most non-standard shapes and poses, it has its limitations attributed

to the 'gross' nature of the Fisher vector employed. Such limitations can be more prominent in cases of partial occlusion or bad clustering, in which overall shape statistics are significantly affected. Aiming for a more fine-grained shape representation. we introduce Spatially Sensitive Fisher Encoding (SSFE), a spatially sensitive method for encoding local shape geometry. SSFE splits each point cloud into clusters obtained by k-means clustering of points, based on spatial coordinates. Instead of deriving a Fisher vector from local histograms associated with all points or some keypoints, SSFE derives a Fisher vector from the mean histograms of each spatial cluster. As was the case with SAFE, we employed standard SVM and k-NN classifiers, although other classification schemes could also be used. Figure 1 (bottom) presents a schematic overview of this method.

The information reflected in per-cluster statistics of local shape geometry extends the information provided by 'gross', spatially agnostic statistics of local histograms. Indeed, the 'gross' information reflected by SAFE is maintained in SSFE, since the mean histogram of all points, which forms one part of the 'gross' Fisher vector in SAFE, can be derived as the mean of per-cluster histograms in SSFE. In essence, SAFE provides a part of the information provided by SSFE considering that the mean histograms in SAFE are identical to the mean histograms in SSFE with one spatial cluster (i.e. the whole object). In addition, the variance of histograms over all points, which forms the other part of the 'gross' Fisher vector in SAFE, can be derived as the mean of per-cluster histogram variances in SSFE. Instead of reflecting shape statistics in the scale of local neighborhood, the derived Fisher vector reflects shape statistics in the intermediate scale of a spatial cluster.

Following the formalism introduced in subsection 3.2, SSFE employs local shape descriptors augmented by spatial coordinates: $\acute{\mathbf{x}}_i=[\mathbf{x}_i\ x_{i1}\ x_{i2}\ x_{i3}]^T$ ($\acute{\mathbf{x}}_i \in R^{D+3}$, $i = 1, ..., N$ and x_{ik}, x_{il}, x_{im} are the three spatial coordinates associated with point i). In the first layer of clustering, k-means is applied on all augmented local shape descriptors $\acute{\mathbf{x}}_i$ ($i = 1, ..., N$), resulting in no spatial clusters. For each resulting spatial cluster sc, we derive the mean histogram $\bar{\mathbf{x}}_{sc}$. These mean per-cluster histograms are essentially the mean local shape descriptors. Following

Eq. 4 and 5, instead of deriving $\mathbf{u_k}$ and $\mathbf{v_k}$ from $\mathbf{x}_k$, for $k = 1, ..., N$, we derive $\acute{\mathbf{u}}_{\mathbf{sc}}$ and $\acute{\mathbf{v}}_{\mathbf{sc}}$ from $\bar{\mathbf{x}}_{sc}$ for sc=1,...,no. The final Fisher vector of SSFE is derived as

$$\acute{\mathbf{f}} = [\acute{\mathbf{u}}_1^T, \acute{\mathbf{v}}_1^T, ..., \acute{\mathbf{u}}_{sc}^T, \acute{\mathbf{v}}_{sc}^T, ..., \acute{\mathbf{u}}_{no}^T, \acute{\mathbf{v}}_{no}^T] \tag{7}$$

When compared to the Fisher vector $\mathbf{f}$ of SAFE, it can be noticed that $\acute{\mathbf{f}}$ is a product of extra information in the form of: (i) spatial coordinates x_{i1}, x_{i2}, x_{i3}, which augment local shape descriptors $\mathbf{x}_i$, and (ii) an extra layer of clustering, i.e. spatial clustering, before Fisher encoding.

SSFE provides an alternative to introduce spatial context within a BoVW framework. Unlike the spatially sensitive BoVW of Bronstein [1], which employs a 2D histogram counting the co-occurrence of visual words over a local neighborhood, SSFE derives per-cluster statistics of local shape geometry, incorporating the intermediate scale of a spatial cluster.

5. Experimental evaluation

This section presents the experimental evaluation of the proposed recognition approach, with information on the datasets, the experimental setup, as well as the results of various experiments aiming at quantitative and qualitative evaluation.

5.1. Datasets and experimental setup

The experimental evaluation of the proposed recognition approach has been performed on datasets of either artificial or real LIDAR data.

Two artificial datasets have been created by using Blensor software, a Blender sensor simulation [1]. Both datasets will be publicly available to allow future comparisons. Scanning has been performed from a height of 2 m. Pedestrian point clouds were obtained by scanning models created with Makehuman software [2],

[1] http://www.blensor.org/
[2] http://www.makehuman.org/

whereas non-pedestrian point clouds were obtained by scanning models from 3D warehouse [3]. In the case of the first artificial dataset, which will be referred as DS1, non-pedestrian object types were not limited to standard objects such as poles, cars and trees, but they were selected to constitute a more diverse set of samples. High resolution versions have been obtained to simulate Velodyne HDL-64 E2 at frame rate=10 Hz. Low resolution versions have been obtained by decimation of high resolution data, with one line kept every two lines and one point kept every two points within the kept lines. All details associated with DS1 are provided in Table 1, whereas Fig. 2 and 3 provide example samples of pedestrian and non-pedestrian point clouds.

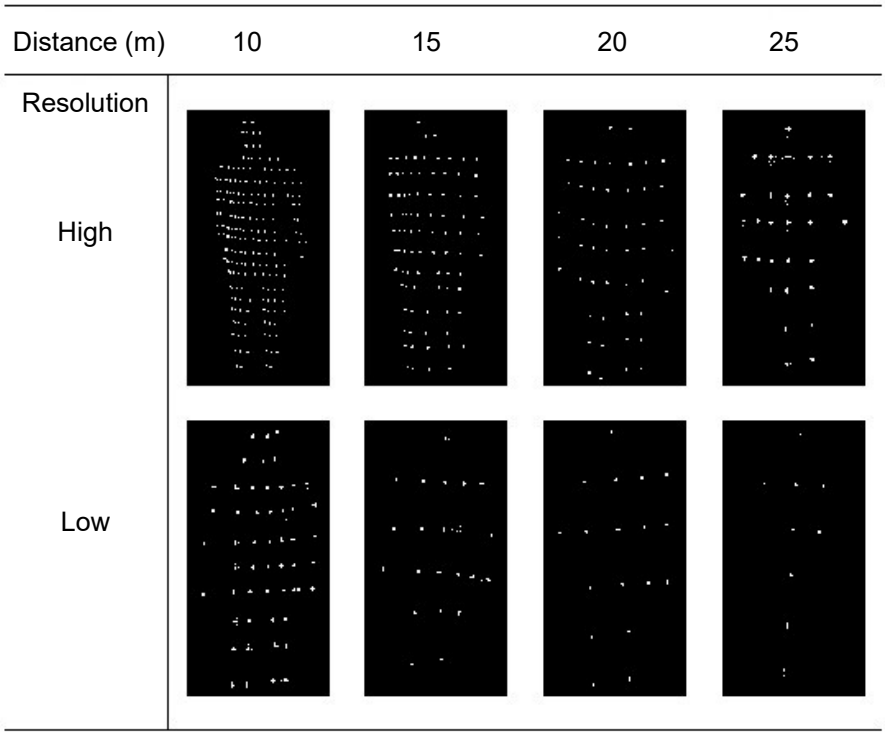


Figure 2: Examples of pedestrian point clouds in DS1.

[3] https://3dwarehouse.sketchup.com/

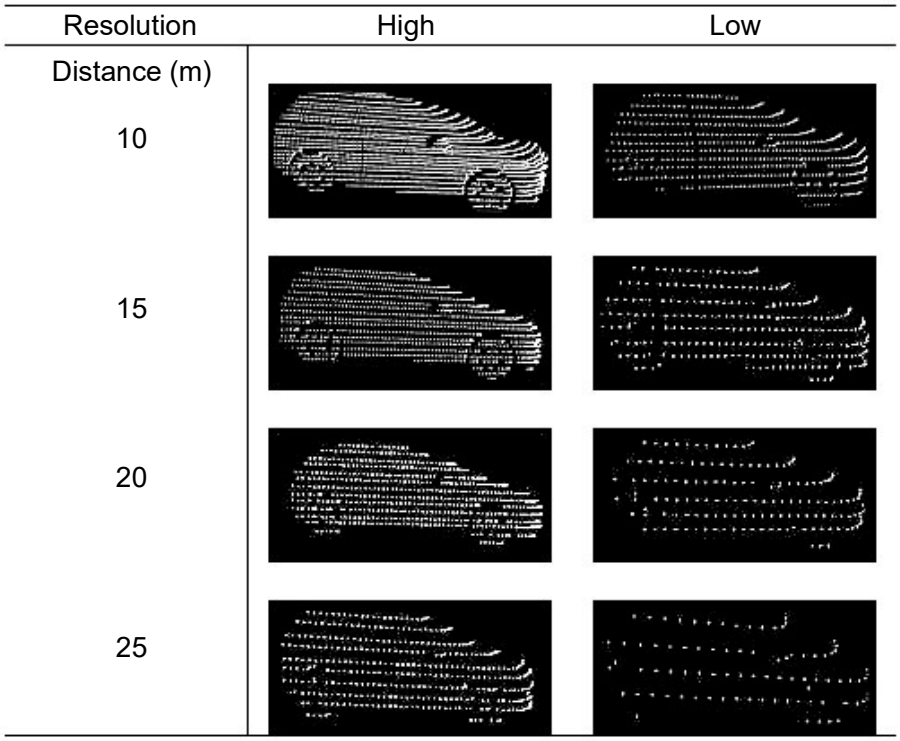


Figure 3: Examples of non-pedestrian point clouds in DS1.

We also used Blensor to create a second artificial dataset, which will be referred as DS2 and comprises a set of challenging pedestrian samples, including cases of occlusion, bad clustering, non-standard poses and non-standard shapes. All details associated with DS2 are provided in Table 1, whereas Fig. 4 provides example samples.

Apart from datasets of artificial LIDAR data, we created a large-scale dataset of approximately 40K real LIDAR samples, derived from the publicly available Stanford Track Collection (STC) [4] [4], which has been recorded in natural street scenes (e.g. university campus, intersections, urban and suburban streets). Similar to DS1, versions for high and low resolution have been created. This dataset,

[4] http://cs.stanford.edu/people/teichman/stc/

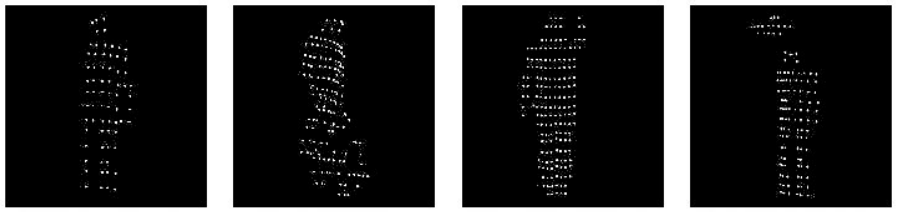


Figure 4: Examples of challenging pedestrian point clouds in DS2, which include partial occlusion from a car, bad clustering with a bicycle, a non-standard pose with hand waving, and a non-standard shape with an umbrella.

named DS3, is balanced in order to enable the training of a classifier capable of detecting both pedestrian and non-pedestrian samples, balancing the accept/reject ability, instead of over-learning either of the two types. It should be noted that STC is the only relevant large scale dataset. However, it contains objects tracked in time and is not directly usable for the evaluation of our approach, which is applied on objects derived from standalone frames of LIDAR data. Moreover, in the original STC and in the range 10-25 m, there were approximately 10K pedestrians in the training set and 10K pedestrians in testing set. We selected all the pedestrians present in that range. Regarding the non-pedestrians, we extracted randomly 10K non-pedestrians in the training set and 10K non-pedestrians in the testing set, all in the same range of interest. In total, this leads to 20K training set and 20K testing set. All details associated with DS3 are provided in Table 1, whereas Fig. 5 provides example samples.

All datasets used in the experiments (DS1, DS2, DS3) are available in: `https://vc.ee.duth.gr/cviu18-db`.

5.2. Results

The results presented include quantitative comparisons for various distances on the artificial dataset DS1, quantitative comparisons on the large scale dataset DS3, qualitative comparisons on challenging queries from DS2, as well as discussion on parameter adjustment and time costs.

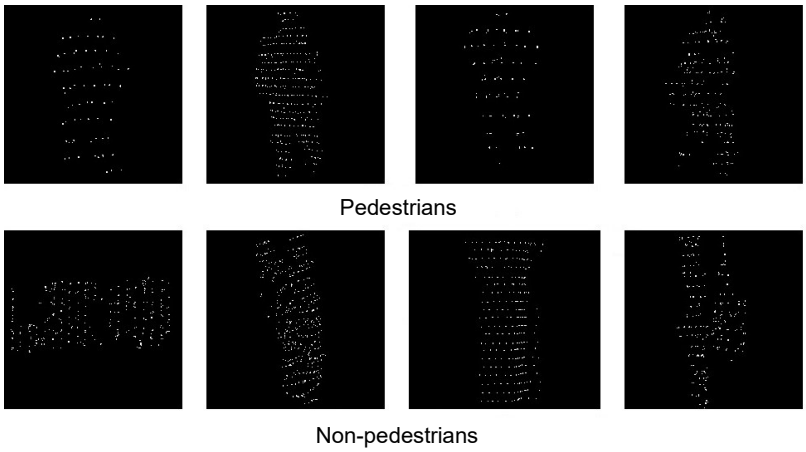


Figure 5: Examples of point clouds in DS3.

5.2.1. Quantitative comparisons on artificial dataset

Table 2 presents the results obtained by SAFE and SSFE, using FPFH, SI and SHOT, when applied on DS1. Experiments are performed for distances equal to 10, 15, 20 and 25 m, both for high and low resolution versions of the dataset. The recognition accuracy for each distance/resolution setting is evaluated by the mean area under curve (AUC) derived from a 10-fold cross-validation experiment, with AUCs measured from receiver operating characteristics (ROCs) corresponding to each fold. Overall, the SVM/SSFE methods obtain higher mean AUCs for most distances and resolutions, with SI/SVM/SSFE

Table 1: Datasets

Dataset name	DS1	DS2	DS3
Difficulty	Standard	Challenging	Standard
Type	Simulated	Simulated	Real
Source	Blensor	Blensor	STC
Distances	10-25 m	15 m	0-35 m
#Pedestrians	84	23	20K
#Non-pedestrians	336	-	20K

method obtaining the highest mean AUCs in most cases. Also, the AUCs obtained using the SVM classifier are mostly higher than the ones obtained using k-NN. Moreover, the SI-based methods obtain slightly higher AUCs than the ones obtained by FPFH and SHOT. Finally, most methods are robust as distance increases.

5.2.2. Quantitative comparisons on large scale dataset

Table 3 presents the results, in terms of mean AUC, obtained by the application of SAFE and SSFE, using SVM and k-NN classifier, with FPFH, SI and SHOT, on high and low resolution versions of DS3. Figure 6 illustrates the corresponding ROC curves. For clarity, we only illustrate ROCs associated with the best performing classifier for each descriptor/encoding pair on the respective resolution. These results validate the recognition capability of SAFE and SSFE on a large scale dataset. Similar to DS1, the SVM/SSFE methods obtain higher mean AUCs per descriptor, for most distances and resolutions, with two marginal exceptions, whereas the AUCs obtained using the SVM classifier are mostly higher than the ones obtained using k-NN. The FPFH/SVM/SSFE method obtains the highest mean AUC, for both high and low resolution. It could be noticed that, unlike DS1, FPFH outperforms SI and SHOT. This is a result of the higher sensitivity of SI and SHOT on radius (see comments to follow on parameterization), which affects their performance when using a uniform radius on a dataset of samples acquired from mixed distances, as is the case with DS3.

5.2.3. Qualitative comparisons on challenging cases

Another set of experiments is performed to qualitatively investigate the robustness of each method against four types of challenging queries, suffering from partial occlusion, bad clustering, non-standard poses and non-standard shapes. Overall, 12 challenging queries from DS2 are performed with SVM or k-NN trained on DS1, with 3 queries for each type. DS1 is used as training dataset for these experiments, since both DS1 and DS2 have been created with the same

Method	AUC (10 m)	AUC (15 m)	AUC (20 m)	AUC (25 m)
FPFH/SVM/SAFE	0.994/0.963	0.993/**0.996**	0.894/0.855	0.793/0.756
FPFH/k-NN/SAFE	0.923/0.934	0.927/0.919	0.906/0.794	**0.956**/0.794
FPFH/SVM/SSFE	**0.999**/**0.978**	**0.998**/0.951	**0.929**/**0.906**	0.918/0.804
FPFH/k-NN/SSFE	0.933/0.947	0.901/0.898	0.924/0.807	0.877/**0.817**
SI/SVM/SAFE	0.953/0.959	0.956/0.964	0.951/0.953	0.948/0.943
SI/k-NN/SAFE	0.948/0.945	0.957/0.974	0.932/0.944	0.954/0.953
SI/SVM/SSFE	**1.000**/**0.998**	**1.000**/**1.000**	**0.998**/**1.000**	**0.986**/**0.999**
SI/k-NN/SSFE	0.962/0.935	0.933/0.948	0.961/0.942	0.940/0.932
SHOT/SVM/SAFE	0.919/0.896	0.891/0.883	**1.000**/**0.983**	0.881/0.954
SHOT/k-NN/SAFE	0.874/0.868	0.841/0.772	0.909/0.896	0.757/0.964
SHOT/SVM/SSFE	**0.982**/**0.960**	**0.979**/**1.000**	0.967/0.979	**0.964**/**0.966**
SHOT/k-NN/SSFE	0.938/0.938	0.923/0.954	0.899/0.942	0.918/0.860

Table 2: Mean AUC obtained by spatially Fisher agnostic encoding (SAFE) and spatial Fisher encoding (SSFE), with FPFH, SI, SHOT local shape descriptors and SVM, k-NN classifiers, when applied on DS1. The results are presented as mean AUC for (high resolution/low resolution). The highest mean AUC per descriptor/resolution is marked with bold, whereas the highest mean AUC per resolution is underlined.

process. In all cases of challenging queries, the SVM-based methods performed equal or better than their k-NN-based countrparts. For this reason and aiming for clarity, we will only present recognition results from the SVM-based methods in the comparisons to follow. Overall, in most types of challenging queries the SSFE-based methods performed better than SAFE-based methods. With respect to the local shape descriptor, the overall performance of FPFH, SI and SHOT is similar.

Figures 7 and 8 illustrate the recognition results obtained in the case of samples associated with partial occlusion and bad clustering, respectively. The SAFE-based methods frequently fail to detect pedestrians in these cases. This

Method	AUC (0-35 m)
FPFH/SVM/SAFE	0.935/0.930
FPFH/k-NN/SAFE	0.908/0.905
FPFH/SVM/SSFE	**0.946**/**0.934**
FPFH/k-NN/SSFE	0.927/0.898
SI/SVM/SAFE	0.895/**0.897**
SI/k-NN/SAFE	0.858/0.869
SI/SVM/SSFE	**0.901**/0.893
SI/k-NN/SSFE	0.815/0.852
SHOT/SVM/SAFE	0.754/0.766
SHOT/k-NN/SAFE	**0.805**/0.737
SHOT/SVM/SSFE	0.803/**0.778**
SHOT/k-NN/SSFE	0.779/0.737

Table 3: Mean AUC obtained by spatially Fisher agnostic encoding (SAFE) and spatial Fisher encoding (SSFE), with FPFH, SI, SHOT local shape descriptors and SVM, k-NN classifiers, when applied on DS3. The results are presented as mean AUC for (high resolution/low resolution). The highest mean AUC per descriptor/resolution is marked with bold, whereas the highest mean AUC per resolution is underlined.

can be attributed to the 'gross' nature of the Fisher vector employed in SAFE, which only reflects overall statistics, naturally affected by partial occlusion or clustering. On the other hand, SSFE employs a more fine grained representation, which reflects statistics per spatial cluster. Spatial clusters which are part of standard pedestrian samples are also present in pedestrian samples suffering from partial occlusion or clustering, allowing SSFE to classify such challenging queries correctly. Still, the SSFE-based methods miss the 'pedestrian occluded by car' sample. In that case, the car covers a large part of the sample, affecting spatial clustering and cluster shape statistics. Interestingly, unlike the other two SSFE-based methods, SHOT/SVM/SSFE misses the 'pedestrian occluded

by pedestrian' sample.

Figure 9 illustrates the recognition results obtained in the case of samples associated with non-standard poses. Both SAFE-based and SSFE-based methods achieve perfect recognition results in these cases. This can be explained by considering that in the case of such samples, local neighborhoods and the derived statistics of local shape descriptors remain unaffected, both for SAFE and SSFE.

Figure 10 illustrates the recognition results obtained in the case of samples associated with non-standard shapes. In these cases, most methods detect one or two out of three samples. FPFH/SVM/SSFE and FPFH/SVM/SHOT miss the 'pedestrian with the dress' sample. In that case, as was the case with the 'pedestrian occluded by car' sample, the dress covers a large part of the sample, affecting spatial clustering and cluster shape statistics. Interestingly, SI/SVM/SSFE detects this sample. Also, all SAFE-based methods miss the seemingly more straightforward 'pedestrian with closed umbrella' sample.

5.2.4. Sensitivity on parameter adjustment

Another set of experiments has been performed on the large scale DS3 dataset to assess the sensitivity of the proposed methods on parameter adjustment. The three parameters affecting recognition accuracy are the radius of the local shape descriptor, the number of GMMs used by both SAFE and SSFE for Fisher encoding, as well as the number of spatial clusters used by SSFE. The FPFH radius which results in the highest mean AUC is 0.15 for both high and low resolution, with a difference of less than 2% when radius is altered by up to 10%. SI and SHOT are more sensitive to radius, with a difference of 4% and 6% in AUC, for similar radius variations. The optimal number for GMMs has been found to be 9 for high resolution and 4 for low-resolution, both for SAFE and SSFE, with a difference of less than 2% when this parameter is altered by 10% for high resolution. The difference in AUC for low resolution is approximately 5% when the same parameter is altered by 25%. The optimal number of spatial clusters in the case of SSFE, has been found to be 5 for high resolution and 3

for low resolution. The difference in AUC is less than 2%, when this parameter is altered by 10% for high resolution. The difference in AUC for low resolution is approximately 5% when the same parameter is altered by 33%. It should also be noted that in the case of the experiments in DS3, the distance of each input, although known, is not used for optimally setting the radius, which is uniformly set to one value [5]. With respect to k-NN, k has been set to 5, whereas k=3 and k=7 resulted in less than 0.5% difference in AUC. Finally, in order to verify the robustness of our evaluation approach against the pedestrian/non-pedestrian sample ratio, we also performed experiments on a testing set derived from DS3, with a ratio equal to that of the original STC. The resulting AUC differs less than 0.5%, which is negligible compared to the differences in AUC measured throughout this work.

5.2.5. Time costs

Table 4 presents the computational cost of each step of the pipeline for SAFE and SSFE, using FPFH descriptor and SVM classifier. The offline part is calculated on the large scale dataset DS3 and comprises descriptor calculation for all samples, mean histogram calculations in the case of SSFE, GMM codebook creation, encoding and classifier training. The online part comprises descriptor calculation for the query sample, encoding and classifier testing. It could be noticed that SSFE is slightly slower than SAFE. In the online part, this can be explained by the extra step of SSFE for mean histogram calculation over the spatial clusters. This step requires 0.03 sec, which is exactly the time difference between the two encoding methods. In the offline part, the difference in time emerges from: i) the mean histogram calculations in SSFE, ii) the codebook creation, which in the case of SSFE involves two layers of clustering (GMM and spatial clustering) instead of only one layer in the case of SAFE. The mean histogram calculations in SSFE require 10 min and the two layers of clustering

[5] In the case of the experiments on DS1, this information is used with a small positive impact on the obtained recognition accuracy

Pipeline step	SAFE	SSFE
Descriptor calculation (min)	60	60
Mean histogram calculation (min)	-	10
GMM codebook creation (min)	9	12
Encoding (min)	5	3
Training (min)	10	10
Total offline (min)	84	95
Descriptor calculation (sec)	0.200	0.200
Encoding (sec)	0.030	0.030
Mean histogram calculation (sec)	-	0.030
Testing (sec)	0.005	0.005
Total online (sec)	0.235	0.265

Table 4: Offline and online time costs for SAFE and SSFE, using FPFH descriptor and SVM classifier. The offline part is calculated on the large scale dataset DS3.

in SSFE require 12 min as opposed to 9 minutes required by the single layer of clustering in SAFE. On the other hand SAFE encoding is slightly slower (5 min instead of 3 min for SSFE), resulting in a total difference of 11 min between SAFE and SSFE offline parts. Using k-NN instead of SVM is associated with a smaller time cost in the offline part but induces 2 to 3 times larger time costs for training in the online part. Also SI-based and SHOT-based methods are approximately 55% and 100% slower than their FPFH-based counterparts, both in offline and online parts. These time costs have been measured for our single core C++/ Matlab implementation running on an Intel Core i7 workstation, operating at 3.5 GHz with 16 GB of RAM.

6. Conclusions

This work introduces two encoding methods for pedestrian recognition based on the statistical shape analysis of 3D LIDAR data: SAFE and SSFE. SAFE is a spatially agnostic recognition method, which employs Fisher encoding to derive global shape statistics. SSFE employs spatially sensitive encoding of local shape geometry, providing a more fine-grained shape representation.The proposed recognition approach is evaluated on artificial LIDAR datasets, comprising standard and challenging samples, as well as on a large scale dataset of real LIDAR data. Three local shape descriptors have been used for testing: FPFH, SI and SHOT, as well as two classifiers: SVM and k-NN. The experimental results lead to the following conclusions:

- both SAFE and SSFE obtain high recognition accuracy on the artificial dataset (DS1), for most descriptor/classifier/distance/resolution configurations. SSFE is more accurate than SAFE in most such configurations. The highest AUC is obtained by SSFE, using SI and SVM (Table 2),

- both SAFE and SSFE obtain high recognition accuracy on high and low resolution versions of a dataset of real LIDAR data, consisting of approximately 40K samples (DS3). SSFE is more accurate than SAFE in most configurations. The highest AUC is obtained by SSFE, using FPFH and SVM (Table 3),

- with respect to the local shape descriptor, SI is more accurate on DS1 (Table 2), whereas FPFH is more accurate in DS3 (Table 3). This is a result of the higher sensitivity of SI in radius, which affects its performance when using a uniform radius on a dataset of samples acquired from mixed distances, as is the case with DS3,

- with respect to the classifier, SVM outperforms k-NN for most descriptor/distance/resolution configurations, in all datasets. k-NN-based methods induce smaller offline time costs but are 2 to 3 times slower in the online part,

- both methods are robust against increasing distance,

- both methods are robust against non-standard shapes and poses,

- SSFE is more robust than SAFE against partial occlusion and bad cluster-

ing. Still, it fails to cope with some challenging queries,

- both methods behave quite smoothly for varying parameter adjustment,

- SSFE is slightly slower than SAFE, with its online part running in about 0.27 sec per sample for our single core C++/Matlab implementation (using FPFH and SVM). Chen et al. [5] report 1.2 sec on a single core for feature calulation. It could be noted that the proposed method requires conventional CPU whereas CNN-based methods require GPU-based parallelizations.

- overall, SSFE appears as a more accurate encoding method than SAFE without inducing significant extra time cost.

The proposed LIDAR-based pedestrian recognition approach could potentially be hybridized with image-based features, as well as with global shape features. With respect to the latter, our preliminary experiments with simple feature concatenation between FPFH and global shape dimensions, led to a boost of approximately 2%, in terms of AUC (DS3). Another advance of SSFE could be based on point cloud segmentation beyond k-means spatial clustering. In the future, spatially sensitive encoding could be applied on features derived with deep learning methods. Overall, SSFE provides a promising direction for pedestrian recognition.

References

[1] A. Bronstein, M. Bronstein, L. Guibas, M. Ovsjanikov, Shape google: geometric words and expressions for invariant shape retrieval, ACM Transactions on Graphics 30 (2011) 1–20.

[2] G. Lavoué, Combination of bag-of-words descriptors for robust partial shape retrieval, The Visual Computer 28 (2012) 931–942.

[3] K. Kidono, T. Miyasaka, A. Watanabe, T. Naito, J. Miura, Pedestrian recognition using high-definition lidar, in: Proc IEEE IV, 2011, pp. 405–410.

[4] A. Teichman, J. Levinson, S. Thrun, Towards 3D object recognition via classification of arbitrary object tracks, in: Proc. ICRA, 2011.

[5] X. Chen, K. Kundu, Y. Zhu, A. Berneshawi, H. Ma, S. Fidler, R. Urtasum, 3D object proposals for accurate object class detection, in: Proc NIPS, 2015, pp. 424–432.

[6] S. Du, B. Liu, Y. Liu, J. Liu, Global local articulation pattern-based pedestrian detection using 3D lidar data, Remote Sensing Letters 7 (2016) 681–690.

[7] C. Premebida, J. Carreira, J. Batista, U. Nunes, Pedestrian detection combining rgb and dense lidar data, in: Proc IEEE IROS, 2014, pp. 424–432.

[8] A. González, J. Villalonga, J. Xu, D. Vásquez, J. Amores, A. López, Pedestrian detection combining rgb and dense lidar data, in: Proc IEEE IROS, 2014, pp. 424–432.

[9] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proc IEEE CVPR, 2005, pp. 886–893.

[10] T. Ahonen, A. Hadid, M. Pietikainen, Face recognition with local binary patterns, in: Proc ECCV, 2004, pp. 469–481.

[11] R. B. Rusu, N. Blodow, M. Beetz, Fast point feature histograms (FPFH) for 3D registration, in: Proc. ICRA, 2009, pp. 3212–3217.

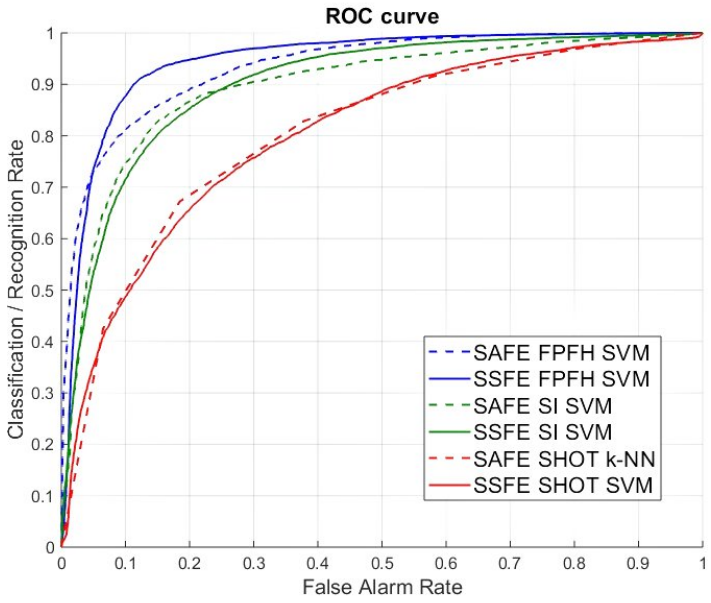
[12] A. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (1999) 433–449.

[13] F. Tombari, S. Salti, L. Di Stefano, Unique signatures of histograms for local surface description, in: Proc. ECCV, 2010, pp. 356–369.

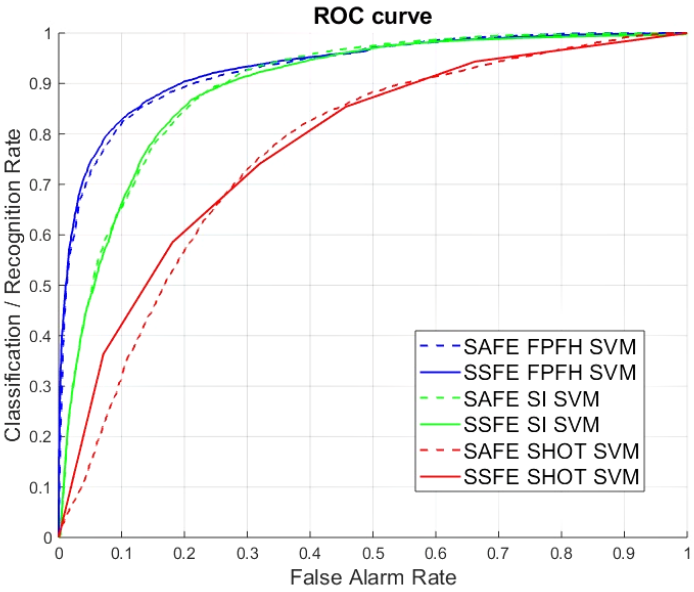
[14] R. B. Rusu, Z. C. Marton, N. Blodow, M. Beetz, Persistent point feature histograms for 3D point clouds, in: Proc. ICIAS, 2008.

[15] I. Pratikakis, M. Savelonas, F. Arnaoutoglou, G. Ioannakis, A. Koutsoudis, T. Theoharis, M.-T. Tran, V.-T. Nguyen, V.-K. Pham, H.-D. Nguyen, H.-A. Le, B.-H. Tran, Q. To, M.-B. Truong, T. Phan, M.-D. Nguyen, T.-A. Than, K.-N. Mac, M. Do, A.-D. Duong, T. Furuya, R. Ohbuchi, M. Aono, S. Tashiro, D. Pickup, X. Sun, P. Rosin, R. Martin, SHREC16 track: Partial shape queries for 3D object retrieval, in: Proc. 3DOR, 2016.

[16] H. Jegou, F. Perronnin, M. Douze, J. Sánchez, P. Perez, C. Schmid, Aggregating local image descriptors into compact codes, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2012) 1704–1716.



High resolution



Low resolution

Figure 6: ROC curves obtained by the application of SAFE and SSFE, with FPFH, SI and SHOT, on high and low resolution versions of DS3. For clarity, we only illustrate ROCs associated with the best performing classifier for each descriptor/encoding pair on the respective resolution.



FPFH/SVM/SAFE	Missed	Missed	Detected
FPFH/SVM/SSFE	Detected	Detected	Missed
SI/SVM/SAFE	Missed	Detected	Detected
SI/SVM/SSFE	Detected	Detected	Missed
SHOT/SVM/SAFE	Missed	Detected	Detected
SHOT/SVM/SSFE	Detected	Missed	Missed

Figure 7: Recognition results on samples associated with partial occlusion.



FPFH/SVM/SAFE	Missed	Missed	Missed
FPFH/SVM/SSFE	Detected	Detected	Detected
SI/SVM/SAFE	Missed	Detected	Missed
SI/SVM/SSFE	Detected	Detected	Detected
SHOT/SVM/SAFE	Detected	Missed	Missed
SHOT/SVM/SSFE	Detected	Detected	Detected

Figure 8: Recognition results on samples associated with bad clustering.

FPFH/SVM/SAFE	Detected	Detected	Detected
FPFH/SVM/SSFE	Detected	Detected	Detected
SI/SVM/SAFE	Detected	Detected	Detected
SI/SVM/SSFE	Detected	Detected	Detected
SHOT/SVM/SAFE	Detected	Detected	Detected
SHOT/SVM/SSFE	Detected	Detected	Detected

Figure 9: Recognition results on samples associated with non-standard poses.

FPFH/SVM/SAFE	Detected	Detected	Missed
FPFH/SVM/SSFE	Missed	Detected	Detected
SI/SVM/SAFE	Missed	Detected	Missed
SI/SVM/SSFE	Detected	Detected	Detected
SHOT/SVM/SAFE	Detected	Detected	Missed
SHOT/SVM/SSFE	Missed	Detected	Detected

Figure 10: Recognition results on samples associated with non-standard shapes.