‘Misspelled’ Visual Words in Unsupervised Range Data Classification: The Effect of Noise on Classification Performance

Michael Firman and Simon Julier

Abstract— **Recent work in the domain of classification of point clouds has shown that topic models can be suitable tools for inferring class groupings in an unsupervised manner. However, point clouds are frequently subject to non-negligible amounts of sensor noise. In this paper, we analyze the effect on classification accuracy of noise added to both an artificial data set and data collected from a Light Detection and Ranging (LiDAR) scanner, and show that topic models are less robust to ‘misspelled’ words than the more näive k-means classifier. Furthermore, standard spin images prove to be a more robust feature under noise than their derivative, ‘angular’ spin images.**

We additionally show that only a small subset of local features are required in order to give comparable classification accuracy to a full feature set.

I. INTRODUCTION

Labeled three-dimensional models of indoor environments are useful in applications that range from robotic interaction to augmented reality. Virtual reality telepresence applications often require the transmission of a large virtual environment in which a ‘visitor’ can immerse themselves [1]. However, the transmission of a fully detailed virtual representation of a real environment is costly and time-consuming. Ways to compress a scene are therefore of high value to such applications. One such method is to recognize repeated components, and only transmit one instance of each. The scene can then be fully reconstructed at the visitor’s end with only a small fraction of the original geometry being transmitted and minimum loss of veracity.

However, most current systems for procuring such labeled models rely on user input, either to manually label individual objects or to label a training set that is then used in a supervised classifier to identify the remaining objects. Such methods are laborious, prone to human bias and most importantly rely on each object that is to be identified in the final scan as having been seen at the user input stage [2].

Topic models have become an extremely popular method for unsupervised object classification. These models were initially developed to identify, in a wholly unsupervised manner, common themes in discrete inputs such as text corpora. Authors have recently shown that the same principles can be applied to other forms of sensor data including images [3] and depth maps [4]. Topic models have been used to create recognition algorithms that allow objects to share parts [5], and to automatically classify both high-level and low-level object primitives [4].

M. Firman and S. Julier are with the Department of Computer Science, University College London. {m.firman, s.julier}@cs.ucl.ac.uk

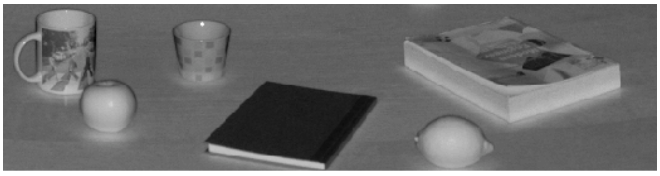


Fig. 1. Examples of the types of object being classified.

The performance of any classification system is strongly determined by the saliency of the feature descriptors used. In the original text-based application of topic models, features consist of individual words. However, when using data from sensors, numerous feature descriptors can be used with different parameterizations. Furthermore, almost all sensing systems are corrupted by non-negligible amounts of measurement noise.

Authors have previously examined the effect of noise supervised classification of text data; for example [6]. In this paper, we perform a systematic study of the effects of classification routines, when the input point cloud data is subject to noise. We use both artificial and real datasets to evaluate classification performance, using both dissimilar and similar sets of objects.

In section II we outline the types of features commonly used in 3D object classification, before introducing features and their conversion to words in sections III. In section IV we then describe the methods used in this paper, and describe the workings of generative topic models. Our results are presented in section V, and section VI states our conclusions and discussion for further work. We find that topic models are more sensitive to noisy input data than traditional classifiers such as k-means, and additionally show that only a small number of features are required for a good classification performance.

II. THE UNSUPERVISED CLASSIFICATION PROBLEM

The goal of unsupervised classification is to assign a label l_i to each object $\mathbf{w}_i$ in a set (corpus) of M distinct objects $D = \{\mathbf{w}_1, \ldots, \mathbf{w}_M\}$, such that objects with common labels are in some way ‘similar’. In this paper we focus on three-dimensional geometry, and aim to assign common labels to objects that are similarly shaped. Shape in itself allows for the type and use of many different objects to be identified, and in the context of scene transmission, shape and texture are the only required properties to give an accurate visual representation of objects.

A. Classification Algorithms

Many clustering algorithms exist for grouping items, for instance Gaussian mixture models, k-means and k-medioids. These have previously been used for unsupervised classification of 3D objects [7] and images [8]. In recent years, various forms of *topic model* have been used for unsupervised discovery of themes in text documents [9], images [2] and 3D objects [4]. Rather than relying on distance metrics in feature space, topic models are generative models that provide a probablistic approach to classification.

B. Latent Dirichlet Allocation

Topic models are a family of generative statistical models used to find common themes, or *topics* in a corpus D Each document $\mathbf{w}_j$ is represented as a sequence of N discrete words $\mathbf{w}_j = (w_1, \ldots, w_N)$. Latent Dirichlet Allocation (LDA) is a popular topic model proposed by Blei et al. [9], where each word w_i is assumed to be drawn from a multinomial distribution associated with a hidden topic variable $z_i \in \{1, \ldots, K\}$, and each z_i is drawn from a multinomial distribution over topics associated with the document $\mathbf{w}_i$.

In our work, we make use of David Blei's C implementation of LDA[1], which infers the mixtures of topics and mixtures of words through Gibbs sampling. We omit full details of the generative model and the steps for inference for reasons of space; see [10] or [4] for full details.

III. Using local feature descriptors as words

As with previous work such as [4], [11], we make use of local features for classification, which provide a description of an object in a neighborhood $N_{\mathbf{p}}$ of a specific scan point $\mathbf{p} = (x, y, z)$. In contrast to many object-level features, local features are not vulnerable to occlusions, tend to be orientation invariant and allow for point-wise classification of unsegmented objects. One widely-used example of a local feature is the 'spin image'.

A. Spin images

Spin images are a descriptor proposed by [12] for locating objects in cluttered 3D scenes with multiple occlusions; they have become a staple feature in classification work (see, for example, [13] and [14]). The spin image at point $\mathbf{p}$ is a 2D array formed by orientating a grid with its normal, and rotating it through a full circle. Each cell in the grid array 'accumulates' each point that falls into it, eventually storing the total number of points. As the array is orientated with the surface normal at $\mathbf{p}$, and the rotational position of points around this normal are ignored, spin images are invariant to both the position and the orientation of the object.

Parameters affecting the spin image are the *support distance* (representing the volume of the scan used to construct the image) and the *raster resolution* (the number of cells in the 2D histogram). By adjusting the support distance, spin images can be converted from a global to a local descriptor, and from a less to a more discriminating feature [15].

[1]Available from `http://www.cs.princeton.edu/~blei/`.

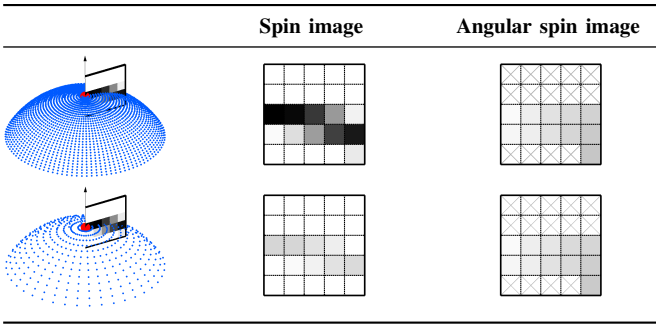


Fig. 2. The sensitivity of standard and angular spin images to sampling density. Standard spin images are highly vulnerable to changes in the sampling density, while angular spin images are more stable. Crosses in the angular spin images indicate cells into which no points fell, which are differentiated from cells in which the average angle is zero.

To help account for noise in the input data, Johnson et al. [12] proposed bilinear interpolation to smooth the image, spreading the contribution of each data point to its four neighboring cells.

B. Angular spin images

A variation of standard spin images was proposed by Endres et al. [4], where each cell records the average angular discrepancy between the normal at $\mathbf{p}$ and the normal of each point falling into the cell. As well as capturing more information about the object's shape, this measure helps to deal with inconsistency in scanning density; the bin values are inherently normalized in the range $[0, \pi]$, whereas the bin values in standard spin images vary with the scanning density (Fig. 2). Angular spin images were shown by [4] to have improved classification accuracy over standard spin images, when used with topic model classifiers.

C. Computation of Normals

Features such as spin images rely on accurate estimates of surface normals, even in the presence of noise. Because the values in angular spin images rely on the estimated normals in the neighborhood $N_{\mathbf{p}}$ as well as at $\mathbf{p}$, we hypothesize that they are more sensitive to errors in normal estimation than standard spin images.

A common approach to normal estimation in unstructured point cloud data is to fit a plane to the points in the neighborhood $N_{\mathbf{p}}$. A larger size of $N_{\mathbf{p}}$ filters out high frequency noise, at the expense of missing surface details.

Cole et al. [16] define $N_{\mathbf{p}}$ as the k-nearest neighbors of each point, and account for noise by adaptively increasing the value of k when the condition number of the plane fit is poor. Mitra et al. [17] use a neighborhood radius approach, where $N_{\mathbf{p}} = \{\mathbf{p}_i : \|\mathbf{p}_i - \mathbf{p}\| < r\}$. They analyze a systematic way of estimating normals in noisy data, deriving a formula for finding the lowest bound on error at each point based on sampling density, local curvature and the level of noise. However, their method still requires two parameters to be empirically found for each dataset.

D. Conversion from local features to words

To convert local features to an object level, histograms are often generated over the values of local features ([13], [18]). Each histogram bin can then be directly used in the object's feature vector. In applying this to spin images, Endres et al. [4] discretize the values in each individual spin image to one of a set number of values. Each unique spin image in the corpus is a word in the vocabulary (or equivalently a dimension in the feature vector). This conversion from local to object features means the classifier operates under the 'bag-of-words' assumption, which allows for interchangeability of word order in a document.

E. Classification in noisy scan data

Noise in LiDAR scans is caused by inaccuracies in the laser beam transmitter and receiver, and from multiple and imperfect reflections off different surface types.

Previous work has added artificial noise (i.e. misspellings) to text corpora to analyze the effect this has on classification performance; see, for example, [6]. Misspelled words in natural languages have different properties to visual words as words are more likely to be misspelled in similar ways (for example typographical errors or common spelling mistakes). Much work in dealing with noise in text therefore aims to correct misspellings. Noise in our case is a corruption over all words, in a far less predictable way (the noise present in LiDAR varies with a multitude of factors such as temperature, ambient lighting and distance to scanner, to name a few [19]) and there is no dictionary of 'correct' spellings.

While classifiers tend to be robust to noise to some extent, we are interested in seeing what combination of features and descriptors perform optimally as noise is increased in the input data.

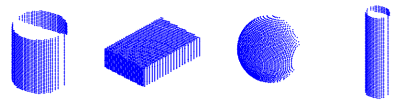
IV. Methodology

To establish the effect of noise on classification, we used two datasets; the first was an artificial dataset, to which we could add arbitrary amounts of noise. The second is a dataset of real objects captured with a LiDAR scanner.

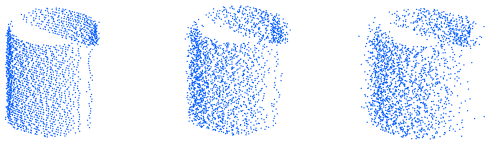
A. Artificial data generation

To examine the effect of noise, we generated an artificial dataset consisting of four differently shaped objects; tall cylinders, short cylinders, spheres and cuboids (Fig. (a)). We generated five instances of each class, with only the viewpoint changing between each instance. The objects were 'scanned' at an equivalent distance of 2 m, at an angular and azimuth resolution of $0.054°$—this represents a typical 'mid-resolution' scanning density on the Faro scanner.

Artificial Gaussian noise, with mean σ_n, was added to each point, along the direction of the ray cast. This simulates the type of noise present in LiDAR scans (Figs. (b)–(d)).



(a) $\sigma_n = 0.0$



(b) $\sigma_n = 0.002$ (c) $\sigma_n = 0.005$ (d) $\sigma_n = 0.010$

Fig. 3. Different levels of distance noise applied to artificially generated objects.

B. LiDAR Dataset

We collected our LiDAR dataset using a FARO Photon 120 phase-shift laser scanner. We scanned at a resolution one-fifth of the maximum resolution at a scanning distance of 2 m, giving a density of approximately 38 points/cm^2; this represents a typical range that may be used in a virtual reality application.

The data set consists of four different classes—mugs, books, confectionery and fruit. Four instances of each class exist—so, for example, there are four types of mug—and we took four scans of each object instance, at various poses. In our experiments we aim to separate the 64 objects into their four ground-truth class assignments.

As with the artificial dataset, we added artificial Gaussian noise to the LiDAR data, to simulate data collected in an environment more susceptible to noise.

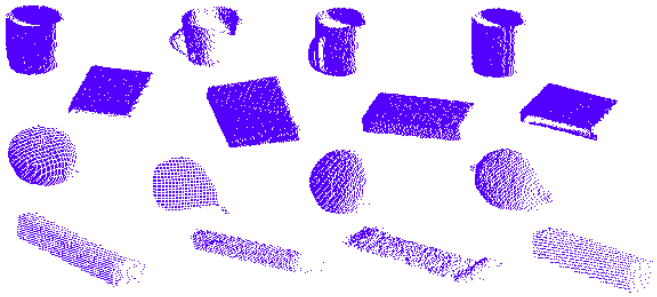
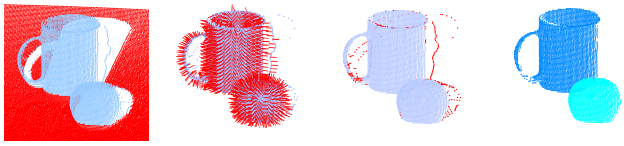


Fig. 4. Point cloud data of the type being classified. From top to bottom: mugs, books, fruit and chocolate bars. See also Fig. 1.

C. Pre-processing

As in previous work ([4], [20], [18]), we focus on classifying objects sufficiently widely spaced to be separated from both the background and each other using straightforward segmentation techniques. Segmenting objects in point clouds without classification cues is less challenging than in images, as the 3D data gives more clues as to suitable partition boundaries. In man-made environments in particular, the planar nature of the surroundings allows for isolated foreground objects to easily be separated from the background and other objects through simple methods such as min-cut graph-based algorithms [20] or $2\frac{1}{2}$D bounding boxes [18]. We leave the



(a) (b) (c) (d)

Fig. 5. The pre-processing steps used to create distinct objects from point cloud scans. (a) the table plane is located and removed using a RANSAC algorithm . (b) The computation of normals. (c) Floating pixels are identified as those points having normals close to perpendicular to the scan beam. (d) Finally, a distance-based algorithm is used to separate the data.

challenge of simultaneous segmentation and classification using topic models as further work.

To segment our LiDAR data set, we used a RANSAC algorithm [21] to detect and remove the desktop surface that the objects sit on. We then used distance-based clustering to separate the point cloud into distinct scan segments.

The surface normal at each point $\mathbf{p}$ in both the artificial corpus D_{art} and the LiDAR corpus D_{lidar} was estimated by fitting a plane to its k-neighborhood using the method of least squares. As in [16], we adaptively increase the k-neighborhood size when the conditioning number of the plane fit (ratio of largest to smallest eigenvectors of the covarience matrix) was below a certain threshold. For our scan data, we empirically found that a good fit was achieved by iteratively increasing k from 75 up to 200, stopping when the condition number of the plane fit either rose above 25 or started to fall, indicating a maximum had been found. This typically equated to a radius of inclusion of between 1 and 2 cm.

Estimating the normals through plane-fitting leads to an ambiguity, in that each normal could point in one of two directions from the surface. We use the fact that our data is collected from a single viewpoint to resolve this ambiguity.

Around the edges of foreground objects in D_{lidar}, the laser beam is often reflected from both the object and the background. This causes *floating* or *mixed* pixels, which appear as data points between the foreground and background (Fig. 5(c)). We identify these points as those having an estimated surface normal approximately (i.e. within 0.025π) perpendicular to the direction of the scanner beam.

Our final, segmented LiDAR dataset thus consists of 64 separated objects, made up of a total of 207 695 points.

D. Feature Computation

For each point in both D_{art} and D_{lidar}, we computed both standard and angular spin images over a range of parameter settings. Preliminary experiments confirmed the findings of Endres et al. [4] that low resolution spin images gave optimum classification performance. We experimented over five values of resolution and support distance, and found that 4×4 angular spin images with a horizontal and vertical support distance of 20 mm gave the best classification. We accept the limitation in our approach that these parameters were hand-refined rather than selected adaptively.

Unlike standard spin images, a zero-valued cell in an angular spin image does not indicate that no points fell into it, but instead indicates that the average angle difference was zero. To differentiate between zero angle difference and zero points falling into a bin, we indicated these empty bins with a `NaN` value (Fig. 2), thus augmenting angular spin images to contain an extra degree of saliency.

E. Conversion to Words

To convert the continuous spin images to discrete words, we quantized the values in each cell. When quantizing the spin images, the values in each cell in the angular spin images were quantized from $[0, 2\pi]$ to T discrete values. We experimented with $T = \{2, 6, 10, 14, 18\}$, and found $T = 6$ gave the highest classification rate. As the values in standard spin images are unbounded, we found that similarly quantizing to a fixed number of values led to the largest values biasing the conversion of the smaller values. Instead, we found we achieved a better classification rate by rounding the cell values to the nearest multiple of T'; experimentation led to a value of $T' = 35$ being selected, which gave good classification results for our data. Each unique spin image in the corpus was then used as a dimension in the feature space of the corpus.

F. Classification

Once the feature vectors were computed, a number of different classification algorithms were used. k-means was used on normalized feature vectors; 50 multiple replicates were run to reduce the algorithm's sensitivity to initial choice of cluster centers. Our experiments found that using the cosine distance between feature vectors gave a better performance than Euclidean or Manhattan distance.

Occasionally the LDA classifier would assign one topic to two classes, and leave one topic with a low representation in all objects. In these cases, where the number of unique assigned topics was fewer than the topics being looked for, we would discard the result and repeat the classification.

V. Results and Evaluation

A. Evaluating the accuracy of unsupervised classification

Unlike supervised classification, in unsupervised learning the ground truth relates to the partitioning rather than absolute labeling; there is therefore no natural concept of a 'correct' label for any one object. However, we know that for an unsupervised classification method to be accurate, it should ensure similar objects are assigned to the same cluster, and dissimilar objects to different clusters.

To capture these requirements, we use the Adjusted Rand Index (ARI) [22] to compare the ground truth with the inferred labels. Given a ground truth labeling $\{g_1, \ldots, g_M\}$ and our inferred labels $\{l_1, \ldots, l_M\}$, each pair of objects $\{(\mathbf{w}_i, \mathbf{w}_j) | 1 \leq i < j \leq M\}$ is assigned to one of four sets. The cardinalities of these sets, a, b, c and d, are described in Table I.

The ARI is then computed as

TABLE I

THE FOUR SETS USED FOR CALCULATION OF THE ARI

	Ground truth	
Inferred label	Pair in same group	Pair in different groups
Pair in same group	True positive a pairs	False positive b pairs
Pair in different groups	False negative c pairs	True negative d pairs

$$ARI = \frac{\binom{n}{2}(a+d) - [(a+b)(a+c)+(c+d)(b+d)]}{\binom{n}{2}^2 - [(a+b)(a+c)+(c+d)(b+d)]} \quad (1)$$

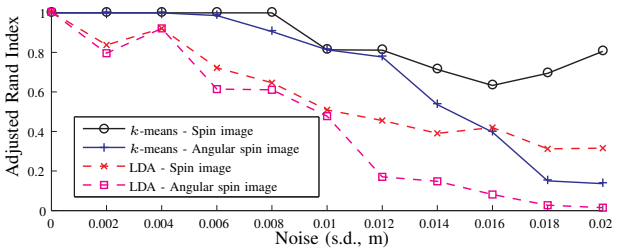
An ARI of 1 indicates a perfect partitioning, while an ARI of 0 represents a partitioning no better than chance.

As this metric requires absolute label values to be given and cannot use mixtures of labels, for the LDA results we use the most strongly weighted topic assignment for each object as its inferred label.

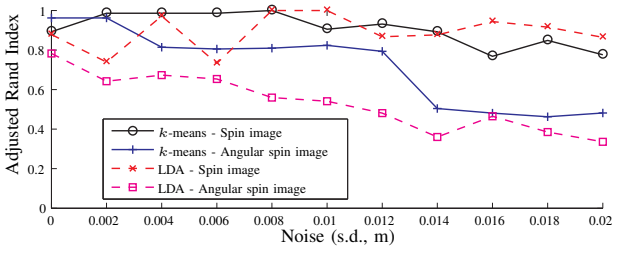
It was noted that the classification routines gave fairly inconsistent results under the same inputs. LDA in particular would tend to give a range of different classification outcomes under multiple reruns. For this reason, we repeated the classifiers several times and averaged the results, in order to give an indication of an expected performance.

B. Effect of noise on classification performance

The Gaussian noise applied to the artificial data had a noticeable effect on classification performance, while the noise added to the LiDAR dataset was less detrimental (Fig. 6). We reason that the more realistic LiDAR dataset, while more noisy to start with, had more descriptive features (such as mug handles) than the simplified artificial dataset, which help to differentiate the classes. For our datasets, the traditional spin images outperform the angular variety as the noise is increased, as hypothesised. We attribute this to the higher order effects of noise present in angular spin images. When computing standard spin images, noise in the data will affect the direction of the normal at $\mathbf{p}$ and will cause some data points to fall into incorrect cells. Angular spin images will suffer from these effects together with further effects from variation in the values of the points being accumulated.

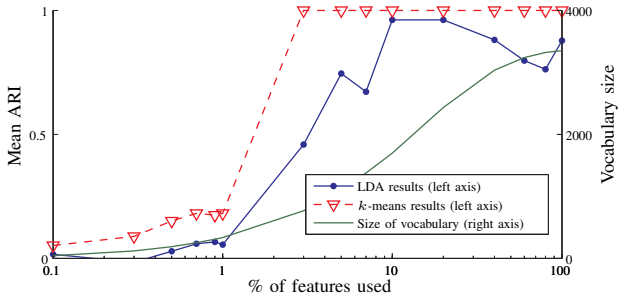


(a) Effect of noise on artificial dataset

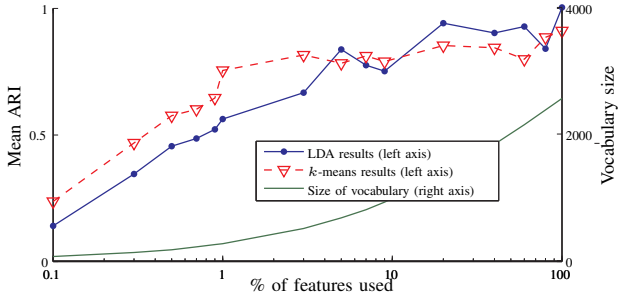


(b) Effect of noise on LiDAR dataset

Fig. 6. Effect of noise on classification performance, for both standard spin images and angular spin images. At each noise level, the classification was repeated 10 times and an average taken to generate the points plotted. Classification performance is measured with the Adjusted Rand Index; 1 is perfect classification, 0 is no better than chance.



(a) Artificial dataset (44,680 features at 100%)



(b) LiDAR dataset (207,695 features at 100%)

Fig. 7. Effect of the number of features used on classification performance of the objects in the LiDAR (b) and artificial (a) datasets, with no noise added. A random subset of the total features (bilinearly interpolated angular spin images) present in the corpus were taken.

C. The effect of number of features used on classification performance

To analyse the effect of feature density on classification performance, we took a random subset of $p\%$ of the standard spin images in the corpus, and ran the classifiers (Fig. 7). As can be seen, while more features tend to improve the performance, the classification rate does not significantly improve above 10%. This has strong implications for creating computationally efficient classification algorithms, as the generation of the spin images is among the more time consuming part of the process (Section V-D).

TABLE II

TIMINGS FOR ONE RUN THROUGH THE CLASSIFICATION PROCESS

	Process	Time (s)
Pre-processing	Plane removal (16 scans)	9.9
	Computation of normals	143.7
	Removal of floats and outliers	54.5
Feature computation	Angular spin image	26.2
	Spin image	21.7
Classification	Conversion to words	0.07
	k-means	0.4
	LDA	2.2

We reason that this is an effect of the repeated nature of features within objects, particularly when sampled at such a high density as the laser scanner; only a few of the features are required in order to give a salient representation.

D. Timings

In Table II we present timings for the computation of each stage in the processing algorithm, for the complete 64-object dataset consisting of 207,695 data points. All code is unoptimised MATLAB code, with the exception of Blei's LDA classification package which was written in C. Timings are for running on a 2.4 GHz Intel Core i5 processor, and code was parallelized where possible.

For the classification we used 10% of the features. Objects were captured in (and planes removed from) 16 separate original scans. k-means and LDA timings are for 1 algorithm replicate.

As can be seen, the computation of normals is the most time-consuming part of the process. Use of constructs such as kd-trees could significantly improve the quoted speed.

VI. CONCLUSIONS AND FURTHER WORK

In this paper, we perform further investigation on previous work using topic models on range data classification. We have shown that noise can have a significant effect on the classification rate when using local feature descriptors for classification. Latent Dirichlet Allocation is shown to be more sensitive to noise than k-means, and standard spin images are shown to perform better than the angular variety for our data. We additionally have shown that only a small subset of features are required to provide a classification performance equivalent to a full feature set.

As with previous work, we have focused on classifying pre-segmented scan objects; we would like to introduce the simultaneous classification-segmentation paradigm to range data processing to allow for closely spaced objects to be segmented. We would additionally like to use a method such as hierarchical Dirichlet processes to automatically infer the number of classes present. While we demonstrate that only a small number of features are required for a good classification performance, this could potentially be further reduced by using visual algorithms for saliency to automatically select the most relevant features.

VII. ACKNOWLEDGEMENTS

We would like to thank the UCL Geomatics department for the use of their scanning equipment, and our anonymous reviewers for their helpful comments.

REFERENCES

[1] M. Buss, A. Peer, T. Schauss, N. Stefanov, U. Unterhinninghofen, S. Behrendt, J. Leupold, M. Durkovic, and M. Sarkis, "Development of a Multi-modal Multi-user Telepresence and Teleaction System," *The International Journal of Robotics Research*, pp. 1298–1316, Oct. 2009.

[2] J. Sivic, B. C. Russell, A. Zisserman, I. Ecole, and N. Suprieure, "Unsupervised Discovery of Visual Object Class Hierarchies," in *Proc. CVPR*, 2008.

[3] E. Bart, I. Porteous, P. Perona, and M. Welling, "Unsupervised Learning of Visual Taxonomies," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, Jun. 2008, pp. 1–8.

[4] F. Endres, C. Plagemann, C. Stachniss, and W. Burgard, "Unsupervised Discovery of Object Classes from Range Data using Latent Dirichlet Allocation," in *Robotics Science and Systems*, 2009.

[5] D. A. Ross and R. S. Zemel, "Learning Parts-Based Representations of Data," *Journal of Machine Learning Research*, vol. 7, pp. 2369–2397, 2006.

[6] S. Agarwal, S. Godbole, D. Punjani, and S. Roy, "How Much Noise Is Too Much: A Study in Automatic Text Classification," in *In ICDM 07: Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*. Ieee, Oct. 2007, pp. 3–12.

[7] S. Goodall, P. H. Lewis, and K. Martinez, "Towards Automatic Classification of 3-D Museum Artifacts Using Ontological Concepts," in *CIVR*, 2005, pp. 435–444.

[8] R. Fergus, P. Perona, and A. Zisserman, "Object Class Recognition by Unsupervised Scale-Invariant Learning," in *CVPR*, 2003.

[9] D. M. Blei, A. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, vol. 3, no. 4-5, pp. 993–1022, 2003.

[10] T. L. Griffiths and M. Steyvers, "Finding Scientific Topics," *Proceedings of the National Academy of Sciences*, vol. 101, no. Suppl. 1, pp. 5228–5235, April 2004.

[11] G. Hetzel, B. Leibe, P. Levi, and B. Schiele, "3D Object Recognition from Range Images using Local Feature Histograms," in *Proceedings of CVPR 2001*, 2001, pp. 394–399.

[12] A. Johnson and M. Hebert, "Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 433–449, 1999.

[13] J. Assfalg, M. Bertini, A. Del Bimbo, and P. Pala, "Content-Based Retrieval of 3-D Objects Using Spin Image Signatures," *IEEE Transactions on Multimedia*, vol. 9, no. 3, pp. 589–599, Apr. 2007.

[14] D. Huber, A. Kapuria, R. R. Donamukkala, and M. Hebert, "Parts-based 3D Object Classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[15] H. Q. Dinh and S. Kropac, "Multi-Resolution Spin-Images," in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 863–870.

[16] D. M. Cole, A. R. Harrison, and P. M. Newman, "Using Naturally Salient Regions for Slam With 3d Laser Data," in *ICRA workshop on SLAM*. IEEE, 2005.

[17] N. J. Mitra, A. Nguyen, and L. Guibas, "Estimating Surface Normals in Noisy Point Cloud Data," in *special issue of International Journal of Computational Geometry and Applications*, vol. 14, no. 4–5, 2004, pp. 261–276.

[18] M. Himmelsbach, A. Mueller, T. Luettel, and H.-J. Wuensche, "LIDAR-based 3D Object Perception," in *Proceedings of 1st International Workshop on Cognition for Technical Systems*, 2008.

[19] M. Hebert and E. Krotkov, "3D Measurements From Imaging Laser Radars : How Good Are They?" *Robotics Institute*, 1992.

[20] A. Golovinskiy, V. G. Kim, and T. Funkhouser, "Shape-based Recognition of 3D Point Clouds in Urban Environments," in *IEEE International Conference on Computer Vision*, no. Iccv, 2009, pp. 2154–2161.

[21] M. Fischler and R. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of the ACM*, vol. 24, no. 6, 1981.

[22] L. Hubert, "Comparing Partitions," *Journal of Classification*, vol. 218, pp. 193–218, 1985.