Learning 3D Point Cloud Histograms

CS229 Machine Learning Project

Brian JONES Michel AOUN

December 11, 2009

Abstract

In this paper we show how using histograms based on the angular relationships between a subset of point normals in a 3D point Cloud can be used in a machine learning algorithm in order to recognize different classes of objects given by their 3D point clouds. This approach extends the work done by Gary Bradski at Willow Garage on point clouds recognition by applying a machine learning approach to learn the histograms. This approach has been tested on a database of 44 types of IKEA models with 40 samples for each type of object.

1 Introduction

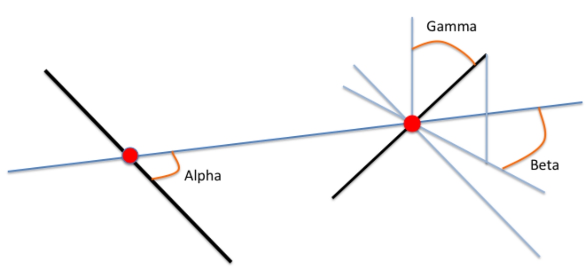
With the advent of robotics and machine learning, the presence of human-like robots in the home is no longer a figment of our imagination, but is becoming a reality. In order for robots to conduct human-like tasks within the home, they need to be able to recognize objects they want to interact with. Our project is aimed at improving this ability. In order for a robot to work independently, it must be able to identify and classify various objects into different categories. Research has already been done in this area through both the use of Neural Networks and Support Vector Machines. The research done with Support Vector Machines was aimed at classifying objects while reducing the number of view points used during training. This research was very successful and showed that Support Vector Machines are a more suitable approach to the object classification problem (see [4]). These SVM experiments used the object's shape and color for classification. Other research papers have proposed 3D point clouds recognition based on histograms. Different types of histograms have been proposed. In [2] Sapiro and al. propose histograms based on local curvature and diffusion distances. Another type of geometric histograms has been proposed by Rusu and al. in [5] We conduct similar experiments with SVM, using such histograms based on normal vectors projecting out from the surface of the object of interest.

2 Description of the learning procedure

Our learning approach procedes in two steps. We start building histograms for each of the objects in our database. The idea of these histograms have been suggested to us by professor Gary Bradski and is described in [1]. It requires computing the normals for each point in the point cloud. To learn these features, we use a support vector machine algorithm that is adapted to our multiclass learning task.

2.1 Building the histograms

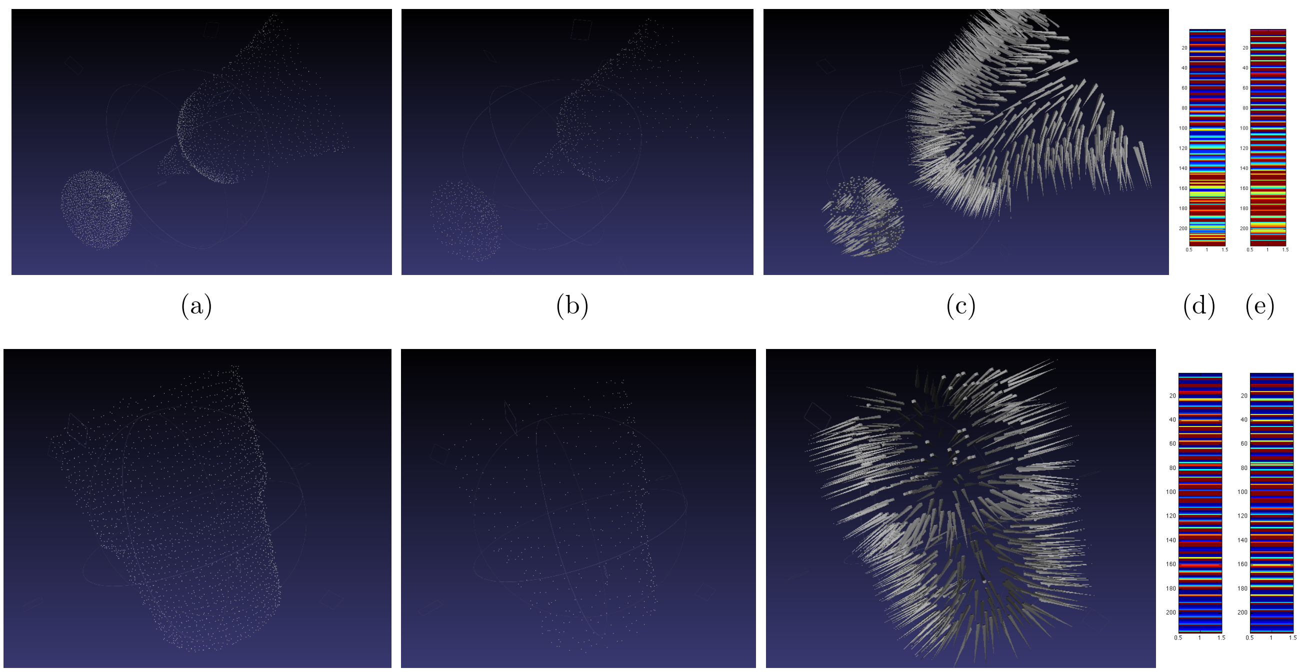
Normals embed essential information about the vicinity of each point in the cloud. They are considered to be reliable information and are commonly used for surface reconstruction. Computing a normal to a point M, has been done by first defining a Region of interest around point M as being a sphere of radius r centered on M. r is proportional to the mean distance between all pairs of points. The points lying in this region of interest are close to the tangent plane at point M. We therefore try to fit to these points the best plane in terms of mean square error. That is done using Principal Component Analysis as described in [3]. The normal to point M is taken as the normal to that plane. This normal is a non oriented normal. Orienting these normals could be done by propagating the normal direction information over neighboring points. However, we chose to build our histograms with non-oriented normals.
Our feature vector takes into account the relation between all couples of 3D points and their corresponding normals. For each couple of points we can compute three angles characterizing the position of the normal to one point relatively to the normal of the other one as described below :



We compute the histogram of all these 3 angles α, β and γ. This histogram is used as the feature describing the point clouds. This feature doesn't depend on how we rotate and translate our object.

In pratice we didn't compute the histograms directly on the entire set of points. We instead downsampled our point clouds. That was done by taking a grid over our whole image and choosing at most one point of the point cloud in each cube of the grid. The step of the grid was fixed empirically in such a way the overall number of points after downsampling is around 1,000 points.

This procedure is described below over two different types of objects. We draw for each type, two histograms corresponding to two different point clouds for the same type of object. We can see a similarity between histograms in the same class and a difference between histograms of different classes of objects. This will allow our learning procedure to behave properly.



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

(f) (g) (h) (i) (j)

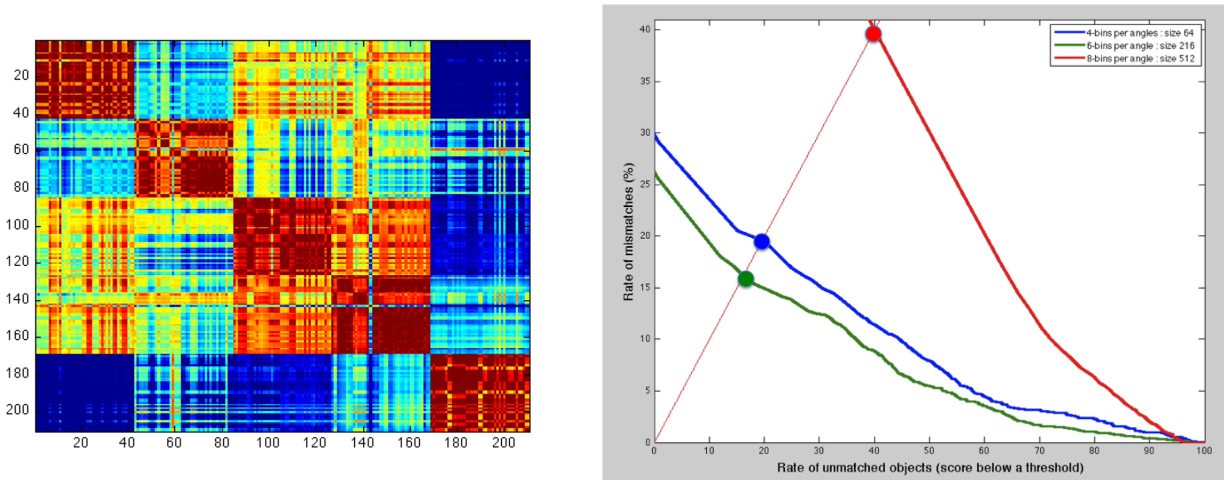
Figure 1: Point Clouds Normals Computation and Histograms Generation : First is our original point cloud, Second is the downsampled point cloud followed by the display of the computed normals. Two Histograms are shown with colder colors corresponding to lower values. These histograms are computed for a discretiztion of each angle into 6 bins. Their size is 6x6x6 = 216.

2.2 Learning the Histograms

In order to learn to classify new objects based on their histograms we used support vector machines. In [1], Chang and al. describe a type of SVM algorithm that is adapted to regression tasks. This type of SVM called ϵ-SVR (R for regression) is useful in the sense it gives a score between 0 and 1 that quantifies the probability an object belongs to a certain class of objects. Our training procedure consists in training 44 ϵ-SVR, one for each class of objects. When it comes to testing, we test a feature histogram with each of our 44 ϵ-SVR. That gives us 44 scores between 0 and 1. We choose the highest score and the corresponding class C. If this score is higher than a certain threshold T we consider that our testing cloud belongs to class C. If the score is less than threshold T, we consider that our object doesn't belong to any of our trained classes.

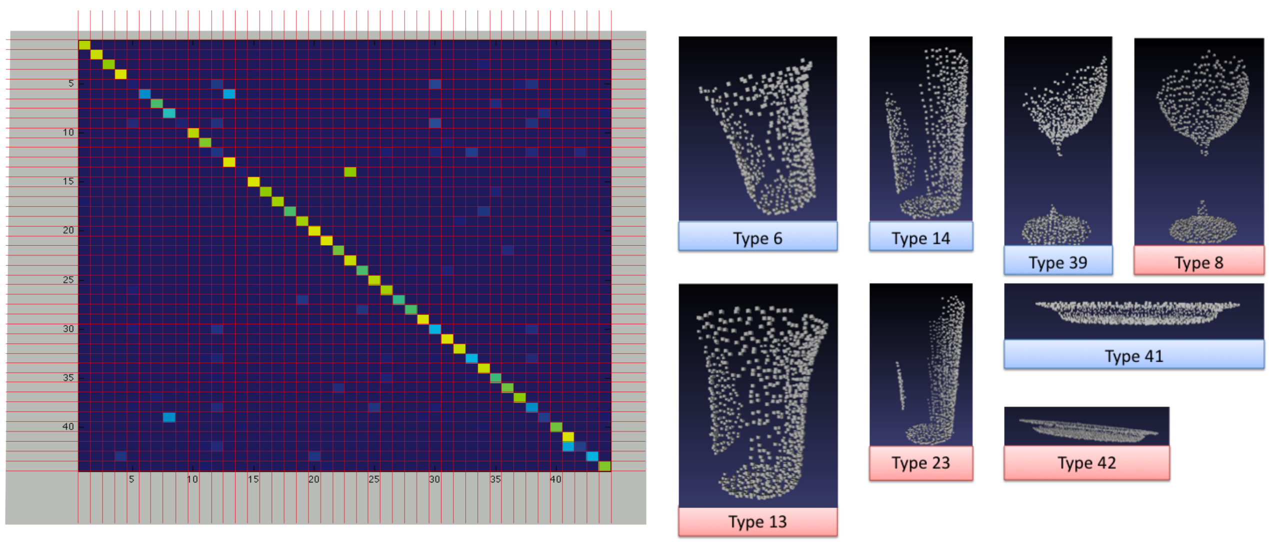
3 Results

The data consists of 3D point map clouds of 44 IKEA models with 40 samples for each model. This database has been provided by Gary Bradski, at Willow Garage. We have first generated histograms for all objects in our database (the histograms were taken of size 216 following a 6 bins discretization for each of our angles). Two histograms were compared using Chi Square. This comparison let us verify our histograms were behaving properly. This is shown in the figure below. After this step of verification, we used k-fold cross validation to train each of our 44 SVR. We did therefore 8 cross validations with 35 elements for training and 5 for testing in each class of object. The results of this k-fold cross validation is reported in the figure below. Here we have chosen different thresholds T and plotted for each threshold the percentage of objects that were misclassified on the y-axis and the number of objects that couldn't be classifier on the x-axis. We did this plot for different sizes of histograms in order to select the most optimal size.
We define the most optimal size of histogram as the one for which the rate of misclassifications, when equal to the rate of unclassified objects, is the lowest. This rate is equal to 16.2% for a 6-bins per angle discretization. It is equal to 19.6% for a 4-bins per angle discretization.
We have drawn for the 216-histogram k-fold cross validation the confusion matrix among the different classes of objects in order to understand where the algorithm misclassified the objects the most. As it is shown below misclassification has occured on pairs of objects that seem highly similar for a human eye. Moreover, it should be noted that our histograms do not take scaling into account (this explains the misclassification between types 41 and 42).



(a) (b)

Figure 2: (a) Comparing Histograms of the first five classes of objects using Chi Square. Hot colors correspond to pairs of histograms that are highly similar - (b) Results of the multiple ϵ-SVR k-fold cross validation tests on IKEA database. The 6x6x6 feature (green curve) is performing the best.



(a) Confusion Matrix (b) Objects of highest confusion degree

Figure 3: (a) Confusion Matrix among the 44 types of objects obtained with k-fold cross validation - (b) Identification of the couples of similar types of objects that are the most confused by our trained ϵ-SVRs : 6 v/s 13, 14 v/s 23, 39 v/s 8 and 41 v/s 42.

4 Acknowledgments

We would like to thank very much professor Gary Bradski for his help and support, for his suggestions and for sharing with us his IKEA models database.

5 Conclusion

Through this project we have proven that combining histograms with machine learning algorithms is a good approach for solving the multi-classification problem on a point clouds data set. The misclassified objects in the data set we have been using correspond to objects of which point clouds are almost the same for a human eye. Our multiple SVM k-fold cross validation approach defined the best histogram size to be a 6x6x6 bins histogram. The success of this learning approach is essentially based on the power of the histograms as it was stated in previous articles such as [5]. We have proven again that these histograms can be considered as a strong discriminative feature to be used.

References

[1] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[2] M. Mahmoudi and G. Sapiro. Three-dimensional point cloud recognition via distributions of geometric distances. *Graph. Models*, 71(1):22–31, 2009.

[3] 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*, volume 14, pages 261–276, 2004.

[4] D. Roobaert and M. M. V. Hulle. View-based 3d object recognition with support vector machines. In *In IEEE International Workshop on Neural Networks for Signal Processing*, pages 77–84, 1999.

[5] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz. Persistent Point Feature Histograms for 3D Point Clouds. In *Proceedings of the 10th International Conference on Intelligent Autonomous Systems (IAS-10), Baden-Baden, Germany*, 2008.