Object Recognition with 3D Models

Bernd Heisele
bheisele@honda-ri.com
Honda Research Institute
Cambridge, USA

Gunhee Kim
gunhee@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, USA

Andrew J. Meyer
ajmeyer.mit.edu
Massachusetts Institute of Technology
Cambridge, USA

Intro: We propose several new ideas of how to use 3D models for view-based object recognition. In an initial experiment we show that even the simple task of distinguishing between two objects requires large training sets if high accuracy and pose invariance are to be achieved. Using synthetic image data, we propose a method for quantifying the degree of difficulty of detecting objects across views and a novel alignment algorithm for pose-based clustering on the view sphere. Finally, we introduce an active learning algorithm that searches for local minima of a classifier's output in a low-dimensional space of rendering parameters.

Experimental setup: Synthetic training and test images were rendered from five textureless 3D models (see fig. 1) by moving a virtual camera on a sphere around each model. The models were illuminated by ambient light and a point light source. The six free rendering parameters were the camera's location in azimuth and elevation, its rotation around its optical axis, the location of the point light source in azimuth and elevation, and the intensity ratio between ambient light and the point light. The rendered images were converted to 23×23 grayvalue images. From those we extracted 640 dimensional vectors of histograms of gradients. Our main classifier was an SVM with a Gaussian kernel.

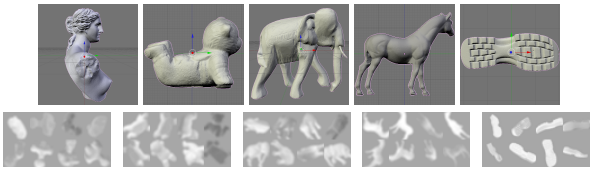

Figure 1: 3D computer graphics models (top) and examples of synthetic images used for training and testing (bottom).

Size of training data: Our first set of experiments dealt with the pose-invariant discrimination between two objects. SVMs were trained and tested on all pairs of objects and compared to nearest neighbor classifiers. Fig. 2 shows the ROC curves for one object pair computed on training sets with sizes between 2,000 and 40,000 samples per class. In all cases, even the ones where the objects looked very different from each other, the best results were achieved with the largest training sets. The large number of support vectors and the poor recognition rates of the nearest neighbor classifiers are further indicators that the learning tasks, simple as they might have seemed initially, are non-trivial and require large training sets.

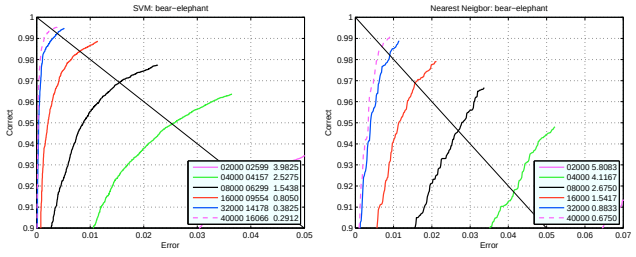

Figure 2: ROC curves for pairwise object recognition, SVM left, nearest neighbor right. The legends contain the number of training samples per class, the total number of support vectors, and the equal error rate (EER) in %.

Pose-invariant object detection: The second set of experiments dealt with pose-invariant object detection. We tried to quantify the degree of difficulty of detecting views of a given object across the view sphere. To do so we computed the mean Euclidean distance in the feature space of each view on the sphere to its nearest neighbors in a large class of background samples, i.e., we based our measure only on the most difficult examples in the background class. The results differed substantially between objects and between views of the same object (see fig. 3). In the future, these results might be used to build classification systems, e.g., by choosing higher pixel resolutions, more training samples, richer features, or more complex classifiers for 'difficult' objects, or 'difficult' regions on the view sphere.

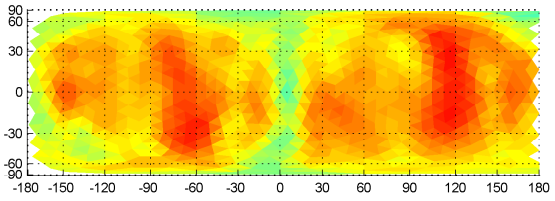

Figure 3: Average distance to nearest background patterns for the views of the elephant model on the view sphere.

Active learning: Our first experiment on pose-invariant discrimination between two objects showed that the training of view-based classifiers from 3D models faces two problems: (1) Large training sets which can break the learning algorithm. (2) The solutions are not sparse, the classifiers will be slow. Our 'active learning' method addresses these two problems. The basic idea is to find valuable/informative object views in the low-dimensional rendering space, six-dimensional in our experiments, and add them to the training set. The method consists of four steps: (1) Train a classifier on a set of randomly selected samples. (2) Find local minima of the classifier's output in the low-dimensional rendering space. (3) Render images at the local minima and add them to the training set. (4) Retrain the classifier on the new training set and repeat the procedure starting from step two. The critical part of the algorithm is step two: the search for local minima. We computed the classifier's output on the same sets of 40,000 views per class that were used in our initial recognition experiment. We picked the most difficult views from each class as starting points of the Nelder-Mead simplex algorithm to find local minima of the classifier's output in the rendering space; some examples of the initially selected views and the views at the nearby local minima are shown in fig. 4. The newly rendered views at the local minima were then added to the existing training set. The results for the bear-vs.-elephant pair are given in fig. 4. A comparison with the results in fig. 2 clearly shows that active learning achieves the same EER with significantly smaller training sets and significantly fewer support vectors than training on a random selection of views.

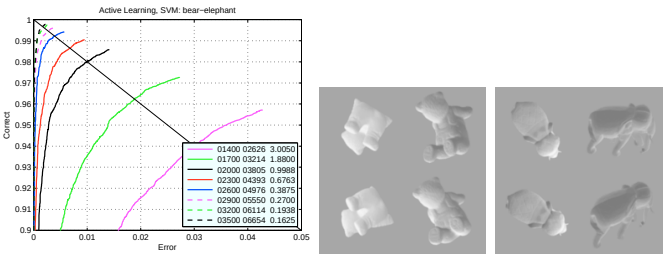

Figure 4: Left: ROC curve for pairwise object recognition with active learning. The legend contains the number of training samples per class, the total number of support vectors, and the EER in %. Right: Example pairs of initial views (top) and views at the nearby local minima (bottom).