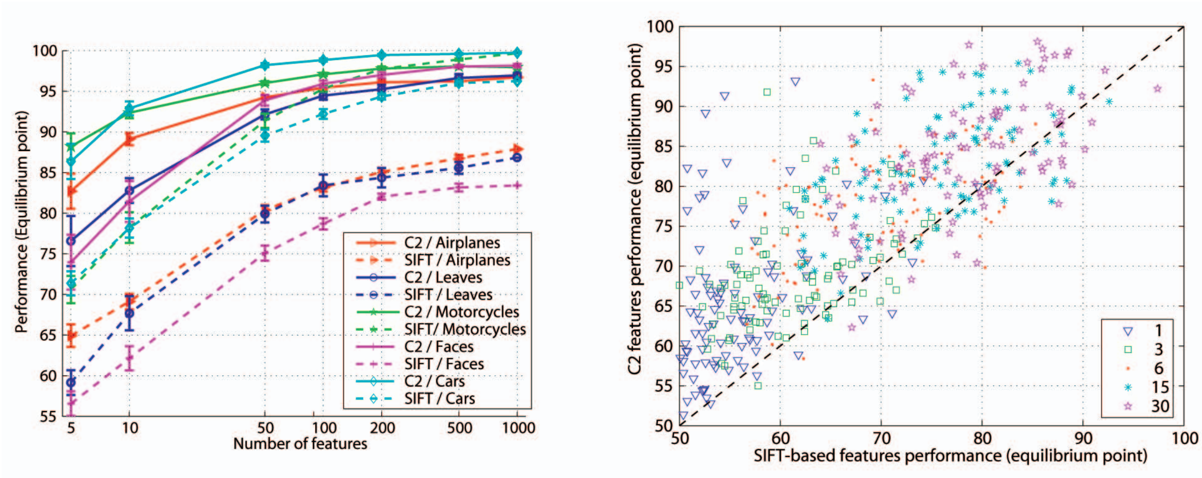
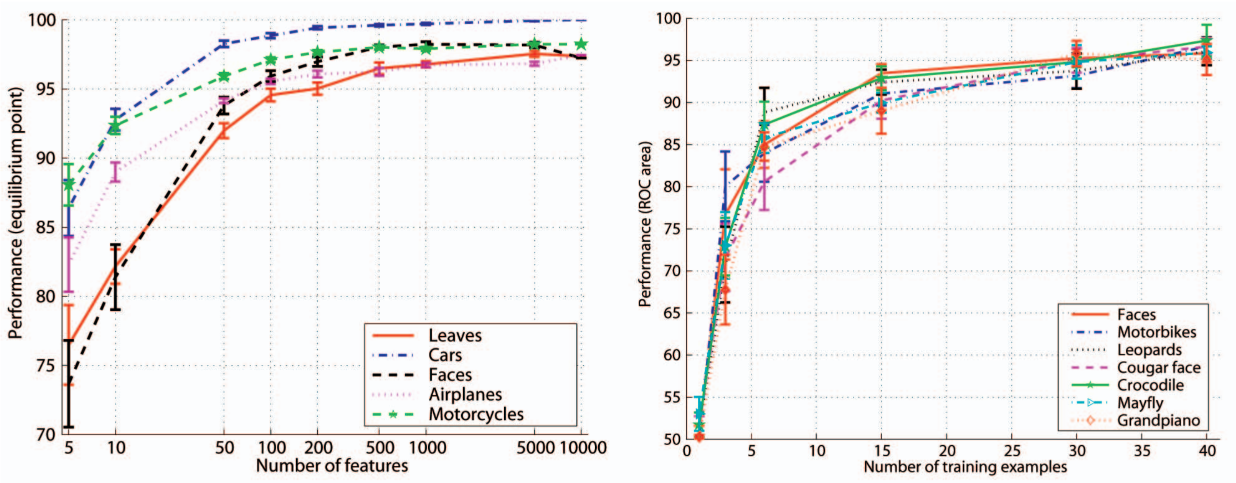


(a) (b)

Fig. 3. Comparison between the SIFT and the C_2 features on the *CalTech5* for (a) different numbers of features and on the (b) *CalTech101* for a different number of training examples.



(a) (b)

Fig. 4. Performance obtained with gentleBoost and different numbers of C_2 features on the (a) *CalTech5* and on sample categories from the (b) *CalTech101* for a different number of training examples.

Fig. 3 shows a comparison between the performance of the SIFT and the C_2 *SMFs* (both with gentleBoost; similar results were obtained with a linear SVM). Fig. 3a shows a comparison on the *CalTech5* database for different numbers of features (obtained by selecting a random number of them from the 1,000 available) and Fig. 3b on the *CalTech101* database for different number of training examples. In both cases, the C_2 features outperform the SIFT features significantly. SIFT features excel in the redetection of a transformed version of a previously seen example, but may lack selectivity for a more general categorization task at the basic level.

Number of features and training examples: To investigate the contribution of the number of features on performance, we first created a set of 10,000 C_2 *SMFs* and then randomly selected subsets of various sizes. The results reported are averaged over 10 independent runs. As Fig. 4a shows, while the performance of the system can be improved with more features (e.g., the whole set of 10,000 features), reasonable performance can already be obtained with 50-100 features. Features needed to reach the plateau (about 1,000-5,000 features) is much larger than the number used by current systems (on the order of 10-100 for [17], [26], [45] and 4-8 for constellation approaches [19], [20], [21]). This may come from the fact that we only sample the space of features and do not perform any clustering step like other approaches (including an earlier version of this system [34]). We found clustering to be sensitive to the choice of parameters and initializations, leading to poorer results.

We also studied the influence of the number of training examples on the performance of the system on the *CalTech101* database. For each object category, we generated different positive training sets of size 1, 3, 6, 15, and 30 as in [21] (see Section 3.1.1). As shown in Fig. 4b, the system achieves error rates comparable to [21] on a few training examples (less than 15), but its performance still improves with more examples (where the system by Fei-Fei et al. seems to be reaching a plateau, see [21]). Results with an SVM (not shown) are similar, although the performance tended to be higher on very few training examples (as SVM seems to avoid overfitting even for one example).