Team Name: Godfellas
Members:
1. Ameya Joshi[1]
2. Ravi Garg[2]
[1] f2010005@goa.bits-pilani.ac.in
[2] ravigarg27@gmail.com

Abstract:

Image categorization has been one of the most difficult problems in the field of computer vision. The major issues that any algorithm has to deal with is affine variance. Therefore, the primary aim of any image categorization algorithm is to encode the variance in each class of images. In our algorithm, we use a first order extension of the bag -of-words approach- Vector of Locally Aggregated Descriptors[1] with SURF[2] features generated for each image. The bag of words approach involves clustering the features from all images and then creating a histogram vector for each image by assigning each feature to a cluster. The resulting vector is then normalized by the L2 norm. In case of VLAD, we add the first order statistics of the residual of each vector with respect to the cluster to which the feature vector belongs. We further train the resulting VLAD vectors for each class with a support vector machine with cross-validation to optimise the parameters. We have implemented the entire pipeline in C++ using OpenCV ver 2.4.6. Our accuracy on the validation set is currently 88.8%.

Regarding VLAD: VLAD is an extension to the bag of words framework. We first cluster our feature vectors- SURF features in our case, with k-means clustering. Then for each cluster, we match the features belonging to that cluster. we then add the residual between the cluster centroid and the matched feature vector to a vector corresponding to the centroid. This gives us a feature vector of KxD size where K is the number of clusters and D is the number of dimensions of the feature vector. This feature vector is then L2 normalized.

Regarding SVM; We use an RBF multiclass classifier SVM. We train the SVM with crossfold validation with k=10. Th tuned SVM is then tested with the validation set. We use the **CvSVM::C_SVC** C-Support Vector Classification in OpenCV ml module for n-class classification ($n \geq 2$), allows imperfect separation of classes with penalty multiplier C for outliers with an RBF kernel (**CvSVM::RBF)** , a good choice in most cases.

$$K(x_i, x_j) = e^{-\gamma ||x_i - x_j||^2}, \gamma > 0.$$

Feature Extraction: We use the SURF feature descriptors[2] as our primary feature vectors as they gave us the fastest and the best performances. Other feature descriptors we tried were SIFT, PCA-SIFT and ORB.

SURF (Speeded Up Robust Features) is a robust local feature detector, first presented by Herbert Bay et al[2]. in 2006, that can be used in computer vision tasks like object recognition or 3D reconstruction. It is partly inspired by the SIFT descriptor. The standard version of SURF is

several times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT. SURF is based on sums of 2D Haar wavelet responses and makes an efficient use of integral images.
It uses an integer approximation to the determinant of Hessian blob detector, which can be computed extremely quickly with an integral image (3 integer operations). For features, it uses the sum of the Haar wavelet response around the point of interest. Again, these can be computed with the aid of the integral image.

Algorithm:

Training:

1. Sort the images according to class using sorter.cpp.
2. For each class:
    a. For each image in the class
        i. Aggregate the SURF descriptors of dimensions D.
3. Run K-Means algorithm on the aggregation of all feature vectors to construct a vocabulary of K clusters.
4 For each image
    a. Initialize a VLAD Matrix(K x D)
    b. For each feature in the image:
        01. Calculate the centroid in the vocabulary closest to the feature vector with L2 norm based matching .
        02. Calculate the difference between the centroid and the feature vector.
        03. Add the difference to the corresponding row of the VLAD matrix
    c. Normalize the VLAD matrix with the L2 norm.
    d. Reshape the KxD matrix to a row matrix of 1xKD
5. Pass the training data of VLAD vectors and corresponding labels to an SVM training algorithm with 10-fold crossvalidation.
6. Save the classifier when the SVM objective function converges.

Testing:
1. For each image:
    a. Generate the SURF vectors.
    b. Calculate the VLAD vectors with the algorithm defined above.
    c. Predict the label using the trained classifier.

Interpretation:

We got a result of 88.8% accuracy on the Validation dataset of 1000 images. We have also tried using multiple feature descriptors namely SIFT, ORB, SIFT with PCA, SURF with PCA. The following are the results of our algorithm with those features:
SIFT - 88.2%

PCA - SIFT - 72% for 64 dimensions
76.4% for 100 dimensions
PCA-SURF - 86.3% for 32 dimensions.

Our results are better than the conventional Bag of Features model as we include first order statistics in our classification algorithm.
We believe that there is huge potential of improvement by adding second order statistics from the feature vectors by using Fisher vectors[3] with tuning of feature vector dimensions and the corresponding GMM though we ran out of time before we could tune the system for better results.

Fisher Vectors with SURF - 15 Gaussians and 64 dimensional SURF - 68.4%
Fisher Vectors with SURF- 10 Gaussians and 64 dimensional SURF -

References:

[1] Jégou, H., Perronnin, F., Douze, M., Sánchez, J., Pérez, P., & Schmid, C. (2012). Aggregating local image descriptors into compact codes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *34*(9), 1704-1716.

[2]Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-up robust features (SURF). *Computer vision and image understanding*, *110*(3), 346-359.

[3]Perronnin, F., Sánchez, J., & Mensink, T. (2010). Improving the fisher kernel for large-scale image classification. In *Computer Vision–ECCV 2010* (pp. 143-156). Springer Berlin Heidelberg.