Binary scene classification

Anjali Poornima K, (anjalipoornima.k16@iiits.in)
B. Tech, ECE, Indian Institute Of Information Technology Chittoor, Sri City, A.P., India

Abstract— ***This paper describes the work on classication of outdoor scenes.Given a set of images of scenes containing multiple object categories (e.g. grass, roads, buildings) our objective is to discover these objects in each image in an supervised manner, and to use this object distribution to perform scene classication. In this paper we have proposed a method for classification of image objects produced by a standard image segmentation al-gorithm using multiclass support vector machine(SVM) classifier integrated with histogram intersection kernel. SIFT is a relatively new feature descriptor which describes a given object in terms of a number of interest points. They are invariant to scaling, translation and partially invariant to illumination changes. This paper primarily focuses on the design of a fast and efficient image object classifier by combining the robust SIFT feature descriptor with intersection kernel SVM which is comparatively better than the existing kernel functions in terms of resource utilization. The experimental results show that the proposed method has good generalization accuracy.***

Keywords:Support Vector Machine (SVM),Histogram Intersection kernel ,SIFT, Feature descriptor.

I. Introduction

Classifying scenes (into MANMADE and NATURAL) is not an easy task owing to their variability, ambiguity, and the wide range of illumination and scale conditions that may apply.We essentially need to figure out features such that Image = f(features). The most important characteristic of feature that we require is a very strong co-relation between the feature value and the class of the image.This makes learning an easier, faster and a less error prone job. Thus, feature identification is among the most important task. Once, the features are identified a classifier can be constructed using state of art algorithms like SVMs, k-means or any other suitable method. With help of this classifier, identification of an input image needs to be done.

The three basic steps involved in scene classification are:

(i) Feature Extraction

(ii) Quantization

(iii) Classification.

The basic steps involved in scene classification are shown in Figure 1.

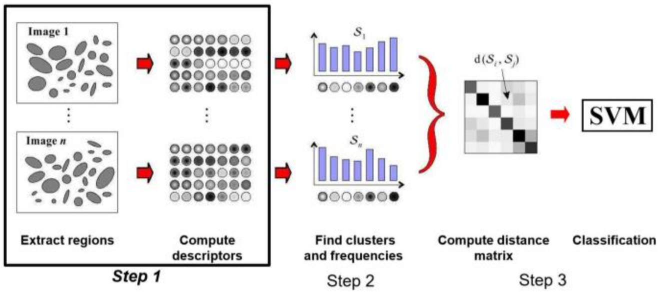


Figure 1. Steps involved in scene classification

Firstly, features are extracted from the scale invariant image regions.Affine invariant regions give too much invariance.Rotation invariance for many realistic collections also give too much invariance.Dense descriptors, Color-based descriptors and shape based descriptors improve results in the context of categories.Multi-scale dense grid: extraction of small overlapping patches at multiple scales and Computes the SIFT descriptor for each grid cells.The SIFT descriptor is an encoding of the edge directions in a local neighbourhood producing a keypoint, given its location in an image at a scale Next, for quantization cluster descriptors are grouped using k-means and each cluster is assigned with each visual word and then frequency histograms are built.In AKM (Approxi-mate k-means), the k-d tree is built on the cluster centers at the beginning of each iteration to increase speed.The k-means clustering is done for that number of clusters (200). The SIFT descriptors are vectors of 128 elements, i. e. points in 128-
dimensional space.

Then a new test image is classified into its respective category using SVM. The SVM is trained for each pair of classes and apply each SVM to test example and assign to it the class of the SVM that returns the highest decision value.Taking each SIFT descriptor in the image, and decide which of the 200 clusters it belongs to, by finding the center of the cluster closest to it. Then simply count how many features from each cluster you have. Thus, for any image with any number of SIFT features you have a histogram of 200 bins. That is your feature vector which you give to the SVM.

II. Related Work

A. Szummer & Picard proposed ”Subblock description ” in 1998

There have been several encouraging research results in scene classification during 1998.Szummer and Picard in 1998, presented a system that classifies indoor and outdoor images on the basis of color histograms and discrete cosine transform coefficients. For a set of 1,300 consumer photographs, the system achieved 90% classification accuracy. While the system achieved relatively high classification accuracy for the Ko-dak consumer photographs reported in [Szummer and Picard

1998],they gradually found that the accuracy is significantly lower (74.7%) on the set of news images we are working with.

B. Anna BoschAndrew, ZissermanXavier, Muoz proposed ”Scene Classification Via pLSA”[1]

Using probabilistic Latent Semantic Analysis (pLSA), a generative model from the statistical text literature, here ap-plied to a bag of visual words representation for each image. The scene classification on the object distribution is carried out by a k-nearest neighbour classifier. The classification per-formance under changes in the visual vocabulary and number of latent topics learnt, and develop a novel vocabulary using colour SIFT descriptors has been investigated.The combination of (unsupervised) pLSA followed by (supervised) nearest neighbour classification achieves superior results.

C. A.Bolovinou,I.Pratikakis,S.Perantonis proposed ”Bag of spatio-visual words for context inference in scene classification”

The proposed method introduces a bag of spatio-visual words representation (BoSVW) obtained by clustering of vi-sual words’ correlogram ensembles. Specifically, the spherical K-means clustering algorithm is employed accounting for the large dimensionality and the sparsity of the proposed spatio-visual descriptors. Experimental results on four standard datasets show that the proposed method significantly improves a state-of-the-art BoVW model and compares favorably to existing context-based scene classification approaches.

D. Yangzihao Wang and Yuduo Wu proposed ”Scene Classification with Deep Convolutional Neural Networks” [2]

This report has presented a novel approach for scene classification based on deep convolutional neural networks. They tried to fill in the semantic gap between the large deep convolutional neural network features from the massive dataset like ImageNet and the high-level context in the scene categories. Their method, which worked by extracting spatial pyramid features from region proposals of images, has shown that deep convolutional neural network is capable of achieving promising results on highly challenging, large-scale dataset which contains both scenes that can be well characterized by global spatial properties and the scenes that can be well characterized by detailed objects they contains. They achieved these results by using a combination of classical computer vision approaches and deep convolutional neural networks

III. DATASET

For this project the famous 15 scene category datasets is used.Images in the dataset are about 250*300 resolution, with 210 to 410 images per class. This dataset contains a wide range of outdoor and indoor scene environments. From this dataset a new dataset is created according to our sub category (i.e., natural or manmade).The scenes from 15 categories that come under natural are coast, open country, mountain, forest and that for manmade office, kitchen, living room, bedroom, store, industrial, tall building, inside cite, street, highway,and suburb.Training is done with 200 images for each sub category and the output is whether the test image is natural or manmade scene.The accuracy is calculated using a distance or conclusion matrix.

NATURAL SCENE IMAGES

MANMADE SCENE IMAGES

IV. SCENE CLASSIFICATION TECHNIQUES

A. Scale Invariant Feature Transform (SIFT) : Image Feature Descriptor

SIFT is used for feature matching in object recognition. The SIFT features are local and based on the appearance of the object at particular interest points, and are invariant to image scale and rotation. First step is to create a set of blurred Gaussian images with different values of sigma.Then the difference of each pair of Gaussian images (DoG) is taken. Then the local maximum which removes the scale uncertainty are found,finding the key points at right scale will be the next step . A feature vector is computed by finding the histogram of gradient directions around the feature point and this is used to match with another image.By doing this we are finding a region which has some fixed properties or features.

B. GIST: Image Feature Descriptor

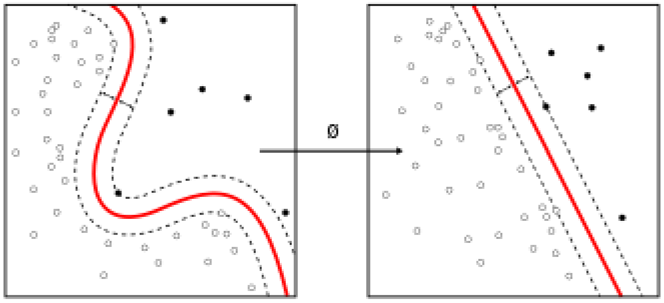
The GIST descriptor focuses on the shape of scene itself, on the relationship between the outlines of the surfaces and their properties, and ignores the local objects in the scene and their relationships. The representation of the structure of the scene, termed spatial envelope is defined, as well as its five perceptual properties: naturalness, openness, roughness, expansion and ruggedness, which are meaningful to human observers. The degrees of those properties can be examined using various techniques, such as Fourier transform and PCA. The contribution of spectral components at different spatial locations to spatial envelope properties is described with a function called windowed discriminant spectral template (WDST), and its parameters are obtained during learning phase. The implementation we used first preprocesses the input image by converting it to grayscale, normalizing the intensities and locally scaling the contrast. The resulting image is then split into a grid on several scales, and the response of each cell is computed using a series of Gabor filters. All of the cell responses are concatenated to form the feature vector

C. K-NEAREST NEIGHBOUR ALGORITHM: Classifier

In pattern recognition, the k-nearest neighbors algorithm (k-NN) is a non-parametric method used for classification and regression.In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression: In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor. In k-NN regression, the output is the propertyvalue for the object. This value is the average of the values of its k nearest neighbors.

D. SUPPORT VECTOR MACHINE(SVM): Classifier

A Support Vector Machine (SVM) is a discriminative clas-sifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples. In two dimentional space this hyperplane is a line dividing a plane in two parts where in each class lay in either side. Multiclass SVM aims to assign labels to instances by using support vector machines, where the labels are drawn from a finite set of several elements.Common methods for such reduction include: Building binary classifiers which distinguish (i) between one of the labels and the rest (one-versus-all) or (ii) between every pair of classes (one-versus-one). Directed acyclic graph SVM (DAGSVM) and Error-correcting output codes



V. SYSTEM FRAMEWORK :Scene Classification

A. Feature extraction

Finding the Pyramid of Sifts Descriptors: The first piece of creating the pyramid sift descriptors was finding the sift descriptors of the image. This was done using the dense sift implementation in VLFeat: vl_dsift. Creating the pyramid of sift descriptors involved splitting the image into multiple levels of sub images.

[FRAMES,DESCRS] = VL DSIFT(I) extracts a dense set of SIFT features from image I. I must be of class SINGLE and grayscale. FRAMES is a 2 x NUMKEYPOINTS, each column storing the center (X,Y) of a keypoint frame (all frames have the same scale and orientation). DESCRS is a 128 x NUMKEYPOINTS matrix with one descriptor per column, in the same format of VL SIFT(). _

We save the descriptors in a .mat file so that once the code is implemented, the features for the training set of images is stored so that each time descriptors for training set are not calculated.

B. Quantization

Creating the Vocabulary and Creating the Histogram: For the new images to be predicted, we need to generate a vocabulary of using the sift descriptors above to create a histogram that defines the image's pyramid of sifts features. The descriptors were grouped together into 200 clusters using K-Means clustering with VLFeat's 'vl kmeans'. Once the vocabulary was defined, we can define an image's pyramid of sifts with a histogram using VLFeat's 'vl kdtreequery' function. This histogram is what we will use when we compare our training and test images.[C, A] = VL KMEANS(X, NUMCENTERS) clusters the columns of the matrix X in NUMCENTERS centers C using k-means. X may be either SINGLE or DOUBLE. C has the same number of rows of X and NUMCENTER columns, with one column per center. A is a UINT32 row vector specifying the assignments of the data X to the NUMCENTER centers.

[INDEX, DIST] = VL KDTREEQUERY(KDTREE, X, Y) computes the nearest column of X to each column of Y (in Euclidean distance). KDTREE is a forest of kd-trees build by VL KDTREEBUILD() When we run vl kdtreequeryfunction to compare these features to our vocabularies features to give us the distance between the features we found and its closest vocabulary feature. Finally, we make a histogram of these closest vocabulary features. These histograms are used to compare the training and the test images. So as to reduce the running time of the code, the histograms that are created are also save as a .mat file.

C. Classification

Once we have features defined for our images, we need a way to classify our test images compared to our training images. The way we use here is with support vector ma-chines(SVM). For each category we are trying to define, we perform a one vs many strategy against our test images. We call 'vl_svmtrain' against training images' features to generate weight and offset vectors.

[W B] = VL_SVMTRAIN(X, Y, LAMBDA) trains a linear Support Vector Machine (SVM) from the data vectors X and the labels Y. X is a D by N matrix, with one column per example and D feature dimensions (SINGLE or DOUBLE). Y is a DOUBLE vector with N elements with a binary (-1 or +1) label for each training point. To a first order approximation, the function computes a weight vector W and offset B such that the score W'*X(:,i)+B has the same sign of LABELS(i) for all i.

D. Implementation

The GIST features used in our project are works done by Antonio Torralba and Oliva Torralba (i) Using SIFT feature descriptor and SVM classifier, we trained the classifier with 500 images for each category and the testing is done with set of training images(50 from each category).

(ii)Using SIFT feature descriptor and SVM classifier, we trained the classifier with 500 images for each category and the testing is done with set of test images(50 from each category).

(iii)Using GIST feature descriptor and SVM classifier, we trained the classifier with 500 images for each category and the testing is done with set of training images(50 from each category).

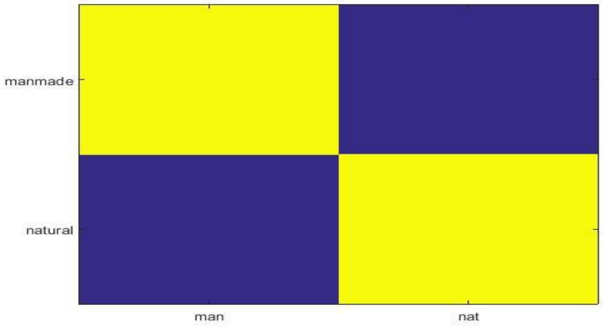
(iv)Using GIST feature descriptor and SVM classifier, we trained the classifier with 500 images for each category and the testing is done with set of test images(50 from each category).

(v)Using SIFT + GIST feature descriptors and SVM classifier, we trained the classifier with 500 images for each category and the testing is done with set of test images(50 from each category).

(vi)Finally testing is done with a single test image and is done at 3 different levels of tree and output is observed.

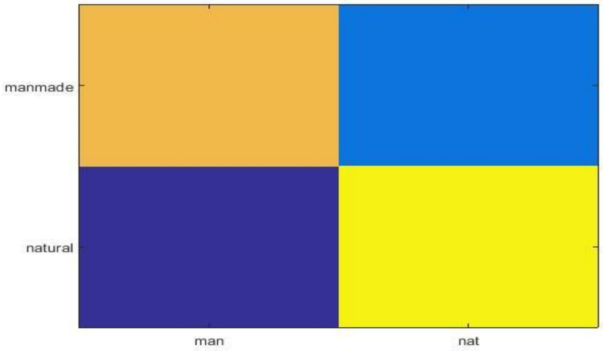
VI. RESULTS

(i) When testing is done with same set of training images, using SIFT and GIST, the accuracy noted was 99% and 100% respectively.

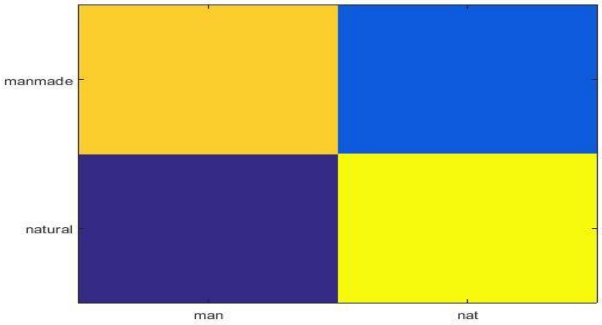


CONFUSION MATRIX USING SIFT AND GIST WITH SET OF TRAINING IMAGES

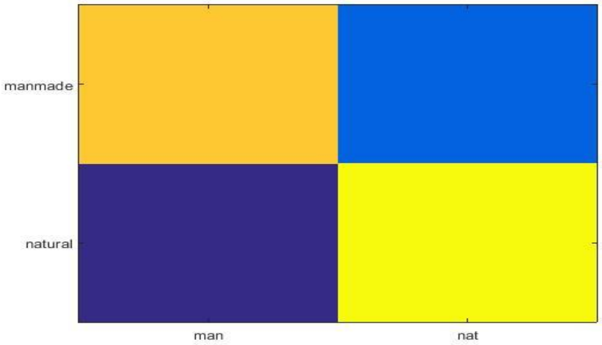
(ii)When the testing is done with seperate test image set, using SIFT we got 90% accuracy, using GIST the accuracy noted was 93% and using SITF + GIST the accuracy was nearly 95.6%



CONFUSION MATRIX USING SIFT WITH SET OF TEST IMAGES



CONFUSION MATRIX USING GIST WITH SET OF TEST IMAGES

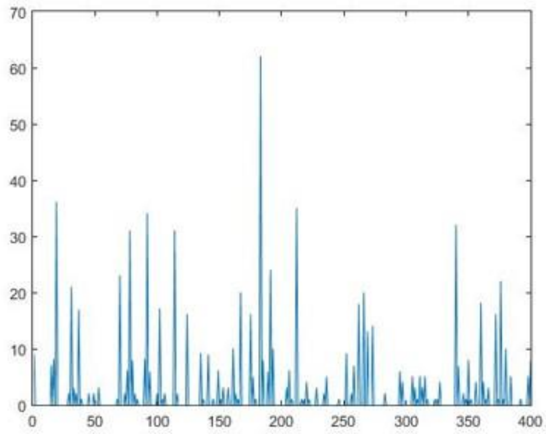


CONFUSION MATRIX USING SIFT + GIST WITH SET OF TEST IMAGES

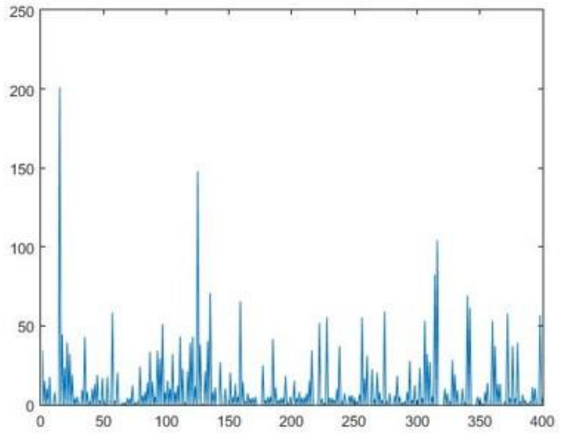
(iii) As this is a binary classification, using SIFT and GIST gives almost equal accuracy , but as the categories increase accuracy given by SIFT gradually differ from GIST as the feature matching becomes complex(there will arise more similar features for few categories, such as Coast and open country ; Office, living room and bedroom etc.) thereby, using GIST or GIST + SIFT feature extraction gives better results.

(iv)While testing for an individual image i.e., an image is given and the output will be weather it is a natural scene or manmade scene, it is tested at 3 levels. Accuracy at L=1 was 80%,image is taken as whole. At L=2, accuracy was 96%. At L=3, accuracy was 94%

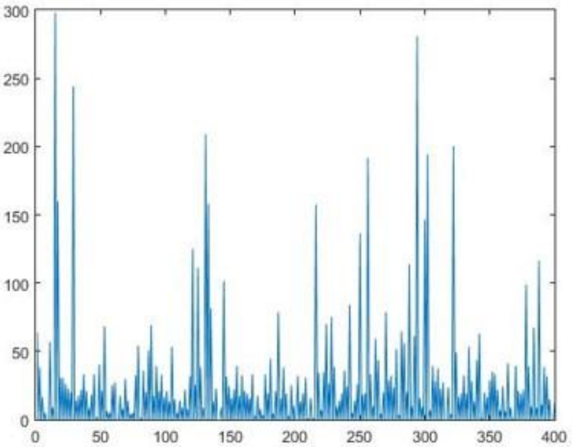
(v)Results improve dramatically as we go from L = 1 to multilevel.For strong features, performance drops from L = 2 to L = 3 because the highest level of L = 3 is too finely subdivided. As it is finely divided individual frames yield too few matches.So at level L = 2 we get strong feature so the accuracy is more.



(i)HISTOGRAM AT L = 1



(ii)HISTOGRAM FOR L= 2



(iii)HISTOGRAM FOR L = 3

VII. SUMMARY AND FUTURE SCOPE

To overcome the disadvantage of global GIST features that lose local information needed for scene classification tasks, a new scene feature description method that combines global GIST with local SIFT features is proposed in this paper. Firstly, local context information and global RGB color quanti-zation information are introduced into the traditional SIFT and GIST features respectively, and then the similarity between the characteristics of the scene is measured based on BOW (Bag Of Words). Finally, the scene classification task is performed with SVM. The influence on classification accuracy of the combined features with different SVM match kernels and BOW is investigated in experiment, and based on three scene datasets, the classification results of the combined feature are compared with that of the methods in literature based on single feature of global GIST or local SIFT, the experimental results show the efficiency of the proposed feature construction method.

REFERENCES

[1] Bosch, Anna, Andrew Zisserman, and Xavier Muoz. ”Scene classifica-tion via pLSA.” Computer VisionECCV 2006 (2006): 517-530.

[2] Zhou, Bolei, et al. ”Learning deep features for scene recognition using places database.” Advances in neural information processing systems. 2014

[3] Dutt, BVV Sri Raj, Pulkit Agrawal, and Sushoban Nayak. ”Scene Classification in images.”

[4] Gokalp, Demir, and Selim Aksoy. ”Scene classification using bag-of-regions representations.” Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on. IEEE, 2007.

[5] Hare, Jonathon S., Sina Samangooei, and Paul H. Lewis. ”Efficient clustering and quantisation of SIFT features: exploiting characteristics of the SIFT descriptor and interest region detectors under image inversion.” Proceedings of the 1st ACM International Conference on Multimedia Retrieval. ACM, 2011.

[6] Hassaballah, M., Aly Amin Abdelmgeid, and Hammam A. Alshazly. ”Image Features Detection, Description and Matching.” Image Feature Detectors and Descriptors. Springer International Publishing, 2016. 11-45.

[7] Hu, Junlin, and Ping Guo. ”Combined Descriptors in Spatial Pyra-mid Domain for Image Classification.” arXiv preprint arXiv:1210.0386 (2012).

[8] Xiao, Jianxiong, et al. ”Sun database: Large-scale scene recognition from abbey to zoo.” Computer vision and pattern recognition (CVPR), 2010 IEEE conference on. IEEE, 2010.

[9] Tomaev, Nenad, and Dunja Mladeni. ”MODIFIED K-MEANS ALGORITHM FOR FINDING SIFT CLUSTERS IN AN IMAGE.”

[10] Li, Li-Jia, et al. ”Objects as Attributes for Scene Classification.” ECCV Workshops (1). 2010.

[11] http://places.csail.mit.edu/index.html

[12] VLFeat Open Source Computer Vision Library. [Online]. Available: http://www.vlfeat.org/.