UB Landscape Recognition System Using SVM and SIFT

Charushi Nanwani, Swati Nair and Jayant Solanki

***Abstract*— This paper talks about the detailed implementation of different categories based scene recognition using geometric correspondence of different interest points detected on the images using SIFT (Scale Invariant Feature Transform). These interest points are based upon the changing gradient of the pixel intensities and summarized to form the local key descriptors for a particular image. In this project we have used another form of SIFT, known as Dense-SIFT as they are supposed to perform better in the object categorization. Our proposed system builds a representation based on bag of visual words and uses SVM classifier along with spatial pyramid matching for classifying the landscape categories. For classification of images, we tried classifiers like SVM with different kernels and KNN and achieved highest accuracy on linear SVM.**

I. INTRODUCTION

This project aimed at reaching higher prediction level on scene classification using **bag-of-words** approach. Starting from the naive approach of bag-of-words with feature extraction using Gaussian filters of different scales on the images and nearest neighbor techniques, we move forward to more advanced SIFT feature detection with linear classifiers learned using Support Vector Machines.

General Model of Scene Recognition (courtesy: Fei Fei):

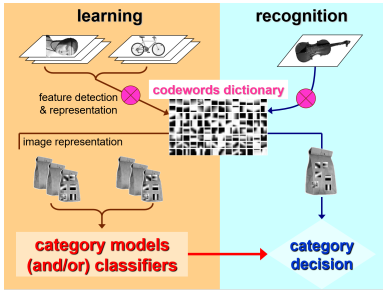

Fig 1.1 General Representation

Bags-of-words approach is one of the most popular and intuitive techniques in solving the scene recognition challenge. This approach uses texture/edges information of the image to create visual words of localized spatial features. Different levels of histograms are then generated to depict the frequencies of these visual words. A dictionary of those visual words is then created using K-Means clustering algorithm. In the end Nearest Neighbor technique is used to find the prediction of the test images. Here different spatial pyramid level has been used starting with the naive zero level pyramid and ending with more sophisticated 3-level pyramid.

Above Bags-of-Words approach has following issues such as:

- Scale and Rotation
- Occlusion
- Translation
- Failure to identify useful information in higher dimensionality representation of image by the nearest neighbor classifier.

To overcome above limitation we went with Bags-of-words model which used **SIFT** descriptors combined with the 1 -vs- ALL Linear SVM classifier for training the image dataset. This classifiers is one of the most simplest learning model. The SIFT feature space is partitioned by a hyperplane learned using the visual words generated from the training images and test cases are categorized based on which side of that hyperplane they fall on. This model outperforms the nearest neighbor model as it disregards those extra features from the visual words which are uninformative. The prediction from a nearest neighbor classifier will still be heavily influenced by those frequent visual words which simply describe the textures and common edges, whereas a linear classifier can learn that those dimensions of the feature vector are less relevant and thus penalize them when making a decision.

II. LITERATURE SURVEY

Bags-of-Words approach was first proposed for the text retrieval domain problem in the text document analysis. Later research made is adaptable for the the Computer Vision related problem. For image analysis, visual words where used based on the vector quantization further processed by clustering local visual features depicting regions of images such as color, edges and textures.

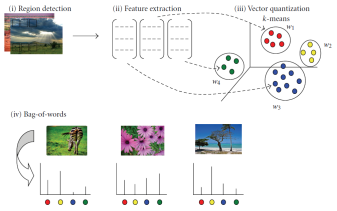

Fig 2.1 Four Steps of Constructing the Visual Words, courtesy ISRN AI

In Grauman[1] we learned about the Kernel-based approach in matching the histograms of different visual words based upon the Euclidean Space. For example, an SVM finds the optimal separating hyperplane between two classes in an embedded space (also referred to as the feature space). A kernel function K : X X R serves to map pairs of data points in an input space X to their inner product in the embedding space F, thereby evaluating the similarities between all points

and determining their relative positions. Linear relations are sought in the embedded space, but a decision boundary may still be nonlinear in the input space, depending on the choice of a feature mapping function : X F. We further learned about the Pyramidal Kernel Matching which performs way better than the Zero-level Pyramid.

We consider an input space X of sets of d-dimensional feature vectors that are bounded by a sphere of diameter D and whose minimum inter-vector distance is d/2:1.

$$X = \left\{ \mathbf{x} | \mathbf{x} = \{[f_1^1, \ldots, f_d^1], \ldots, [f_1^{m_\mathbf{x}}, \ldots, f_d^{m_\mathbf{x}}]\} \right\}, \quad (1)$$

where mx varies across instances in X.

In Svetlana[2] we learned about using the SIFT features along with the Spatial Pyramidal Matching for Scene Recognition. In David[2] we learned about the significance of using the scale invariant SIFT features in identifying the keypoint features in the image which were very useful in the Scene Recognition.

III. PREVIOUS WORK

In Homework 1, we implemented a scene classification system using the bag-of-words approach and Spatial Pyramid Matching. We obtained an overall accuracy of 58.75%. In this project, we aim to build a more accurate recognition system using different algorithms for the feature extractions from the image and SVM/RBF classifier to reduce the misclassification of the images observed in the former approach. Further, we trained this new system on our custom generated image dataset having different landscapes of University of Buffalo, where the system will predict image-classes using the Model generated by the classifier. In the end, test images are then passed to the system where the system will try to correctly classify each test image. The main objective of this project is to achieve higher accuracy rate (greater than 70%).

IV. SPATIAL MATCHING SCHEME

Using the references from the HW 1, we came to know Spatial Matching at different level helps in the precise matching of the two images in high dimensional space. From Svetlana[2] paper we found that the "orthogonal approach" that is performing pyramidal matching in two dimensional image space. Specifically, we quantize all feature vectors into M discrete types, and make the simplifying assumption that only features of the same type can be matched to one another. Each channel *m* gives us two sets of two-dimensional vectors, **Xm** and **Ym**, representing the coordinates of features of type m found in the respective images. The final kernel is then the sum of the separate channel kernels:

$$K^L(X, Y) = \sum_{m=1}^{M} \kappa^L(X_m, Y_m).$$

This approach has the advantage of maintaining continuity with the popular visual vocabulary paradigmin fact, it reduces to a standard bag of features when L = 0.

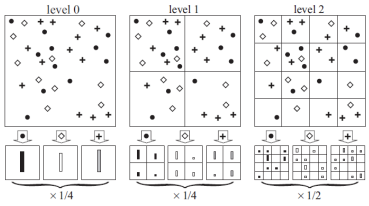

Fig 3.1 Toy example of constructing a three-level pyramid. The image has three feature types, indicated by circles, diamonds, and crosses. At the top, we subdivide the image at three different levels of resolution.

The final implementation issue is that of normalization. For maximum computational efficiency, we normalize all histograms by the total weight of all features in the image, in effect forcing the total number of features in all images to be the same. Because we use a dense SIFT feature representation and we don't want unusual large variance in the pixel values hence we normalise the data.

V. FEATURE EXTRACTION

Scale Invariant Feature Transform (SIFT) is an image descriptor for image-based matching and recognition developed by David Lowe[3]. The features detected using SIFT are localized in spatial as well as frequency domains and are invariant to scale, rotation and viewpoint.

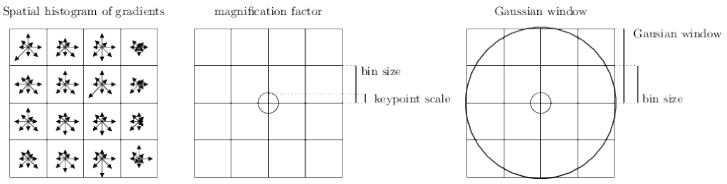

Fig 5.1 SIFT Descriptors

The key stages for generating the set of features are:

- Scale-space extrema detection - This is the initial stage of computation which involves searching across all scales and image locations. This is to ensure scale invariance and is efficiently implemented using Difference-of-Gaussian approach.
- Keypoint localization - Key locations are identified as maxima and minima of the result obtained by applying Difference of Gaussian function across scale space. Low contrast points and edge response points are discarded.
- Orientation assignment -Dominant orientations are assigned to the keypoints. All operations are then performed on the transformed data and this endures that keypoints provide better stability across scale and rotation changes.
- Keypoint descriptor - SIFT descriptors are then obtained by fetching pixels in the selected scale space within a radius of the keypoint. These are transformed into a representation that allows for significant levels of local shape distortion and change in illumination.

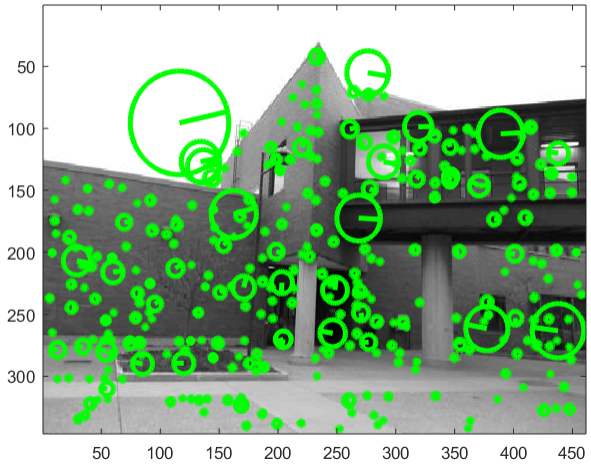

Fig 5.2 Keypoint detection using SIFT

SIFT generates a large number of features which covers the image across different scales and locations. This is useful when identifying objects in a highly cluttered environment. SIFT features are highly distinctive, easy to extract and give a good accuracy with low probabilistic mismatch. For small datasets, recognition can be performed as close to real time.

Generating Features - we used vlfeat[4] which is open source library for computer vision algorithms. We used vl-phow() function which is used for calculating dense SIFT features across color channels. PHOW features are a variant of SIFT descriptors obtained at multiple scales. The color version allows to pass HSV images, extracts descriptors across each channel and stacks them up.

Classification - For training, we used fitecoc() function which is provided by MATLAB Computer Vision Toolbox. We used Linear SVM approach for learning as it provided the best accuracy.

VI. EXPERIMENTAL RESULTS

In this section, we will talk about the result of our scene recognition algorithm on two different data-set:

- SUN- Data Set
- UB Campus Data

Images were read in HSV format for the UB Campus Data Set and HSV/LAB format for the Sun Data Set. We identified **weak features** and **strong features** based upon the number of the clusters we chose during the K-Means Clustering process. Training was done using Multiclass SVM method from the MALTAB Computer Vision toolbox.

Fig 6.1 SUN Data Set.

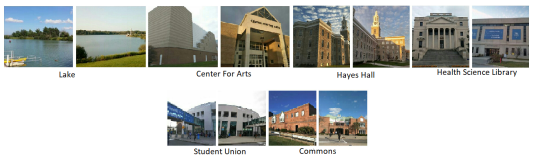

Fig 6.2 UB Campus Set

Id.	Alpha	Clusters	Accuracy % (HW1 method)	Accuracy % (proposed method)	Change in accuracy(%)	Dataset
1	150	200	83	86.6	-3.6	UB Dataset
2	150	200	58.13	83	-24.87	Sun Database
3	150	16	30	65.17	-35.17	UB Dataset
4	150	16	25	66.87	-41.87	Sun Database

fig 6.3 Classification results for the scene category database

	Project table-UB dataset			
Id.	Alpha	Clusters	Accuracy(SPM 3)(%)	Remark
1	60	16	65.1786	*Minimum*
2	60	64	85.7143	
3	60	100	81.25	
4	60	150	87.5	
5	60	200	85.7143	
6	60	400	88.3929	
7	120	16	69.6429	
8	120	64	87.5	
9	120	100	86.6	
10	120	150	82.14	
11	120	200	86.6	
12	120	400	87.5	
13	150	16	65.1786	*Minimum*
14	150	64	82.1429	
15	150	100	92.8571	
16	150	150	88.3929	
17	150	200	86.6071	
18	150	400	91.07	
19	200	16	67.8571	
20	200	64	86.6071	
21	200	100	88.3929	
22	200	150	86.6071	
23	200	200	86.6071	
24	200	400	91.96	
25	500	150	93.75	*Maximum*
26	500	200	93.75	*Maximum*
27	500	400	86.6071	
28	1000	200	88.39	

fig 6.4 Scene Accuracy on UB Campus Data

Best Accuracy for UB Campus at K= 200, alpha = 500		
Class	**Correct Recognition**	**Accuracy**
Center for Arts	16	1
Commons	14	0.875
Crossby Hall	16	1
Hayes Hall	15	0.9375
Health Science Library	16	1
La Salle Lake	12	0.75
Student Union	16	1
Total	**93.75**	**0.8370535714**

fig 6.5 Best Accuracy achieved for UB Campus Data

Comparison of SPM with 3 layers and 4 layers using proposed method on UB database					
Id.	Alpha	Clusters	Accuracy (SPM with 3 layers)(%)	Accuracy (SPM with 4 layers)(%)	Change in accuracy(%)
1	60	16	65.17	86.6	-21.43
2	150	200	86.6	85.7143	0.8857
3	500	200	93.75	86.6071	7.1429
4	1000	200	88.39	90.1786	-1.7886

fig 6.6 Overall

VII. Scene Category Recognition

A. HW 1 dataset

In HW1, we worked with SUN Dataset which consisted of total 1509 images. Out of these, 160 images were used as testing images and the remaining 1349 were training images. For this dataset, we obtained an accuracy of 83.75 % when classified using linear SVM for SIFT features for 180 clusters.

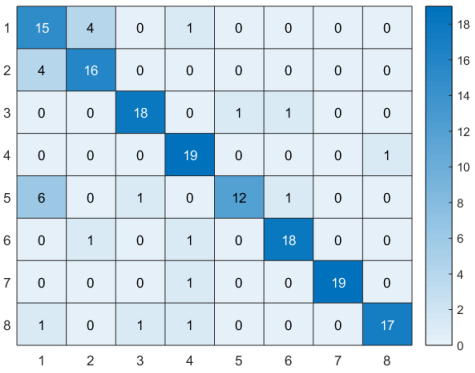

fig 7.1 PLot Confusion for Sun DataSet Campus at K = 180 and Alpha = 120, Accuracy: 83.75%

B. Campus Dataset

For UB Dataset, we have a total of 532 images. Out of these, 112 images were used as testing dataset (16 images for each category) and the remaining were training images. For this dataset, we obtained maximum accuracy of approx. 93 % when classified using linear SVM for SIFT features for 500 clusters.

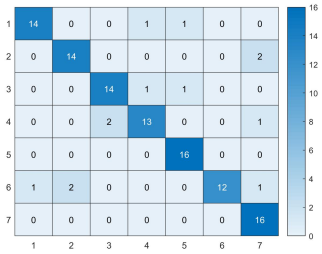

fig 6.7 Plot Confusion for UB Campus at K = 200 and Alpha = 500, Accuracy: 88.39%

VIII. Conclusions

In this project, we attempted to improve the performance of scene classification using a different method for feature extraction and classification. For feature extraction, we used SIFT and SURF for keypoint detection on images. We found out that SURF is efficient than SIFT, but SIFT is more robust. Using SURF, we got an accuracy of 74% and using SIFT, we achieved an accuracy of 86%. Also, we tried different classifiers such as linear SVM and KNN. Linear SVM provided far better accuracy than KNN. We also tried running the proposed method on HW1 dataset i.e. Sun database too and got better performance. As UB dataset is small, we found out that the accuracy is uch higher because of its smaller size. If we train the model on larger dataset, then the accuracy will slightly be decreased, but still will be better than the HW1 method. Thus, we can say that the proposed model gives consistent performance for different datasets.

Acknowledgment

We thank Prof. Kevin Keane and TA, Radhakrishna Dasari for assisting us in project ideas and helping us in understanding different computer vision concepts.

References

[1] The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features, Kristen Grauman and Trevor Darrell

[2] Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, Svetlana Lazebnik, Cordelia Schmid, Jean Ponce

[3] Distinctive Image Features from Scale-Invariant Keypoints, David G. Lowe

[4] VLfeat: An open and Portable Library of Computer Vision Algorithms, Andrea Vedaldi, Brian Fulkerson