Effects of Image Retrieval from Image Database using Linear Kernel and Hellinger Kernel Mapping of SVM

Swathi Rao G

Abstract-In this paper basically we have compared the efficiency of image retrieval using the most efficient type of classifiers i.e. Support Vector Machine (SVM) with linear kernel mapping and the Hellinger kernel mapping applied to various classes of images and also varied representation of the corresponding image classes using Matlab R2009a.The results obtained from simulation show that Hellinger kernel mapping yields improved performance as compared to the linear kernel mapping. The database consists of a collection of images of different classes whose feature vectors are calculated using Dense Scale Invariant Feature transform (SIFT) and are quantized to visual words whose frequency is recorded in a histogram for each spatial tile of the image. Then the resultant feature vectors are used to train both the linear kernel and Hellinger kernel for different class of images and varied representation of them. The resulting precision and recall graphs and Average Precision (AP) gives us the performance efficiency of various classes of images with varied representation and the classifier mapping used. It is observed from the graphs and AP values that efficiency of the system is increased with Hellinger kernel given more positive images are contained in the database.

Keywords- Hellinger Kernel, Image Classification, Image Retrieval, Kernel Functions, Linear kernel, SIFT, Support Vector Machine

1 Introduction

Digital image processing is a rapidly evolving field with growing applications in science and engineering. Image processing holds the possibility of developing the ultimate machine that could perform the visual functions of all living things. Many theoretical as well as technological breakthroughs are required before we could build such a machine. At the same time there is an abundance of image processing applications that can serve mankind with the available and anticipated technology in the near future.

It has a broad spectrum of applications such as remote sensing via satellites and other spacecrafts, image transmission and storage for business applications, medical processing, radar, sonar and acoustic image processing, robotics and automated inspection of industrial parts. But all of this is possible only if we can retrieve the information easily and efficiently. It is not only sufficient to retrieve the information required quickly but also accurately. The process of locating a desired image from a collection of image database is not only quite complicated but also varies from one technique to the other. The problems are becoming widely popular and have become an active area of research and development.

The goal of an image retrieval system is to retrieve a set of images from a collection of images such that this set meets the user's requirements. The user's requirements can be specified in terms of similarity to some other image or a sketch, or in terms of keywords. An image retrieval system provides the user with a way to access, browse and retrieve efficiently and possibly in real time, from these databases. Well-developed and popular international standards, on image coding have also long been available and widely used in many applications. The challenge to image indexing is studied in the context of image database, which has also been researched by researchers from a wide range of disciplines including those from computer vision, image processing, and traditional database areas for over a decade. The aim of our paper is to show the effects of Hellinger kernel and linear kernel when applied to image database and that Hellinger kernel gives better results when compared to that of the linear kernel.

Large collections of images are becoming available to the public, from photo collections to web pages or even video databases. To index or retrieve them is a challenge which is the focus of many research projects. A large part of this research work is devoted to finding suitable representations for the images, and retrieval generally involves comparisons of images. In this paper, we choose to use histograms as an image representation because of the reasonable performance that can be obtained in spite of their extreme simplicity. From classification trees to neural networks, there are many possible choices for what classifier to use. The support vector machine (SVM) approach is considered a good candidate because of its high generalization performance without the need to add a priori knowledge, even when the dimension of the input space is very high. Intuitively, given a set of points which belongs to either one of two classes, a linear SVM finds the hyper plane leaving the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyper plane. This hyper plane minimizes the risk of misclassifying examples of the test set. Organization unfolds as increasingly better results are obtained through modifications of the SVM architecture.

The rest of the paper is organized as follows. Section 2 describes the Theoretical Considerations Section 3 explains Database Preparation, Section 4 explains about feature extraction, Section 5 illustrates the training of classifier for specific classes of image with varied representation, Section 6 describes classification of images and assessing their performances using precision-recall plots and finally section 7 and 8 discusses the results , gives the conclusions respectively.

2 Theoretical Considerations

2.1. Overview of SVM

The Support Vector Machine (SVM) was first proposed by Vapnik and has since attracted a high degree of interest in the machine learning research community. Several recent studies have reported that the SVM generally are capable of delivering higher performance in terms of classification accuracy than the other data classification algorithms. They have been employed in a wide range of real world problems such as text categorization, hand-written digit recognition, tone recognition, image classification and object detection, micro-array gene expression data analysis, data classification. However, for some datasets, the performance of SVM is very sensitive to how the cost parameter and kernel parameters are set. As a result, the user normally needs to conduct extensive cross validation in order to figure out the optimal parameter setting. This process is commonly referred to as model selection. One practical issue with model selection is that this process is very time consuming. We have experimented with a number of parameters associated with the use of the SVM algorithm that can impact the results. These parameters include choice of kernel functions, varied representation of images and the number of training examples.

SVM's belong to a family of generalized linear classification. A special property of SVM is they simultaneously minimize the empirical classification error and maximize the geometric margin. So SVM are called Maximum Margin Classifiers. SVM is based on the Structural risk Minimization (SRM).

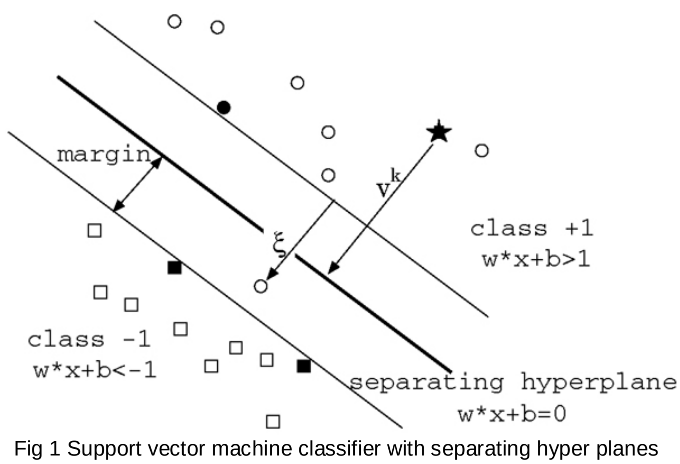

Fig 1 Support vector machine classifier with separating hyper planes and their equations

SVM maps input vector to a higher dimensional space where a maximal separating hyper plane is constructed as shown in the above fig1. The two parallel hyper planes are constructed on each side of the hyper plane that separates the data. The separating hyper plane is the hyper plane that maximizes the distance between the two parallel hyper planes. An assumption is made that the larger the margin or distance between these parallel hyper planes the better the generalization error of the classifier will be.

Large scale non linear support vector machines can be approximated by linear ones using a suitable feature map. The linear SVM's in general are much faster to learn and test than the non linear ones using a suitable feature map. To be extended upon collecting data Part of the appeal for SVMs is that non-linear decision boundaries can be learnt using the so called 'kernel trick'. Though SVMs have faster training speed, the runtime complexity of a non linear SVM classifier is high. Boosted decision trees on the other hand have faster classification speed but are significantly slower to train and the complexity of training can grow exponentially with the number of classes. Thus, linear kernel SVMs have become popular for real-time applications as they enjoy both faster training and classification speeds, with significantly less memory requirements than non-linear kernels due to the compact representation of the decision function. Discriminative approaches to recognition problems often depend on comparing distributions of features, e.g. a kernelized SVM, where the kernel measures the similarity between histograms describing the features. In order to evaluate the classification function, a test histogram is compared to histograms representing each of the support vectors.
This paper presents a comparison of a linear kernel and a Hellinger kernel classifier but the kernel values are explicitly computed so that the classifier remains linear in the new feature map and also gives better performances when compared to the linear kernel results. A comparison of the two is shown and explained in the upcoming sections.

2.2. Kernel Selection of SVM

The concept of a kernel mapping function is very powerful. It allows SVM models to perform separations even with very complex boundaries. The design of the SVM classifier architecture is very simple and mainly requires the choice of the kernel. Nevertheless, it has to be chosen carefully since an inappropriate kernel can lead to poor performance. There are currently no techniques available to "learn" the form of the kernel; as a consequence, the first kernels investigated were borrowed from the pattern recognition literature. Many kernel mapping functions can be used – probably an infinite number. But a few kernel functions have been found to work well in for a wide variety of applications. The equation of a SVM kernel function can be given as
$K (x_i, x_j) \equiv \Phi (x_i). \Phi (x_j)$ where xi and xj are feature vectors.

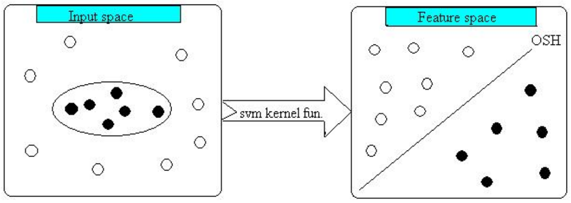

Fig 2 Mapping of input space to feature space by SVM kernel function

There are many kernel functions in SVM, so how to select a good kernel function is also a research issue. However, in our research work we have considered the two popular kinds of kernels:

Linear kernel: $K(h(i), h'(i)) = \sum h(i)\, h'(i)$

Hellinger kernel or Bhattacharya coefficient

$K(h(i), h'(i)) = \sqrt{\sum_i h(i)\, h'(i)}$

Where h and h′ are normalized histograms

2.3. Model Selection of SVM

Model selection is also an important issue in SVM. Recently, SVM have shown good performance in data classification. Its success depends on the tuning of several parameters which affect the generalization error. We often call this parameter tuning procedure as the model selection. If you use the linear SVM, you only need to tune the cost parameter C. Unfortunately, linear SVM are often applied to linearly separable problems .Many problems are non-linearly separable. For example, Satellite data and Shuttle data are not linearly separable. Therefore, we often apply non linear kernel to solve classification problems, so we need to select the cost parameter (C) and kernel parameters. We usually train and test both the kernels separately under different classes of data and varied representation and normalization and the obtained results can be seen with the help of precision recall graphs and the ranked list of images that are displayed.

3 Database Preparation

The given classes of images are tested one by one with each linear kernel and Hellinger kernel. The database consists of three image classes' airplanes, motorbikes, and people. It also contains background image which doesn't contain the above mentioned classes. The data is divided as per the below mentioned table which consists of the training and testing images for each class. The sample of Image database can be seen in fig 5 in the upcoming section 6.

TABLE 1
DISTRIBUTION OF IMAGE DATABASE

	Bikes	Airplanes	People	Background
Training	120	112	1025	1019
Testing	125	126	983	1077
Total	245	238	2008	2096

4 Feature Extraction

It computes a dense set of multi-scale SIFT descriptors from a given input image. Vocabulary learning is then used to cluster a few hundred thousand visual descriptors into a vocabulary of 10^3 visual words. A spatial histogram calculates the joint distribution of appearance and location of the visual words in an image.

The feature vector consists of SIFT features computed on a regular grid across the image (`dense SIFT') and vector quantized into visual words. The frequency of each visual word is then recorded in a histogram for each tile of a spatial tiling. The final feature vector for the image is a concatenation of these histograms.

The Scale Invariant Feature Transform (SIFT) is probably the most popular feature used in computer vision. Scale-invariant feature transform (or SIFT) is an algorithm in computer vision to detect and describe local features in images. The algorithm was published by David Lowe in 1999. It detects salient image regions (key points) and extracts discriminative yet compact descriptions of their appearance (descriptors). For any object in an image, interesting points on the object can be extracted to provide a "feature description" of the object. This description, extracted from a training image, can then be used to identify the object when attempting to locate the object in a test image containing many other objects. To perform reliable recognition, it is important that the features extracted from the training image be detectable even under changes in image scale, noise and illumination. Such points usually lie on high-contrast regions of the image, such as object edges.

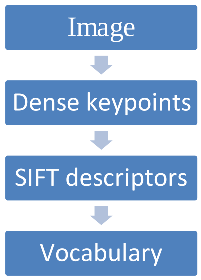

Fig 3 Feature extraction

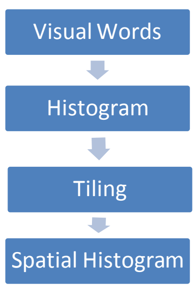

Fig 4 Obtaining Spatial Histogram

The figs 3 & 4 illustrate the process of feature extraction in the corresponding steps. This may be used as a basis for a three-dimensional reconstruction of the scene. Alternatively, key points with discredited descriptors can be used as visual words as an intermediate image characterization. Histogram of visual words can then be used by a classifier to map images to abstract visual classes (e.g. car, cow, horse). Despite its popularity, the original SIFT implementation is available only in binary format. Dense SIFT is a fast algorithm for the computation of a dense set of SIFT descriptors.

5 Classifier Training and Varied Image Representations

The spatial histograms are used as image descriptors and fed to a linear SVM classifier. Linear SVMs are very fast to train, but also limited to use an inner product to compare descriptors .Much better results are obtained by pre-transforming the data, which computes an explicit feature map that makes non linear kernel as a linear one. Each class of image is trained and tested using both linear kernel and Hellinger kernel, varying the number of training images, varied image representation which means turning off the spatial tiling and representing it with only a single histogram .This is done by merging the tiles together. The results of all the above cases are plotted using the precision and recall curve which gives statistical comparison of the methods. The precision and recall can be calculated with the below mentioned formulae.

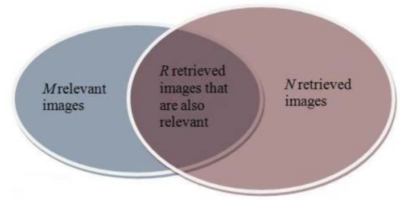

Fig 5 Precision and Recall Illustration

- Precision= R/N
- Recall= R/M

6 Results and Discussion

The below tables compare various classes of image database and their retrieval performance with linear kernel and Hellinger kernel in various conditions whose inferences are discussed.

Table 2

Results with Linear kernel

Classes	Test AP	No. of images retrieved in Top 36	PR on train	PR on test
Airplane	0.55	30	61.31	54.76
Spatial tiling off	0.48	31	50.77	47.96
Histograms Normalized	0.55	24	88.60	55.22
Varying fraction of training images	0.55	30	96.01	71.04
Bikes	0.29	15	36.74	28.65
Spatial tiling off	0.18	9	18.69	17.79
Histograms Normalized	0.52	26	94.88	51.86
Varying fraction of training images	0.29	15	36.74	28.65
People	0.61	26	68.63	61.35
Spatial tiling off	0.57	20	60.05	56.84
Histograms Normalized	0.71	30	96.01	71.04
Varying fraction of training images	0.61	26	68.63	61.35

AP → Average precision
PR → Precision recall

ISSN 2229-5518

Fig 6 Sample of Image Database containing 2491 images

As mentioned in the database section we consider a database with 2491 images spread in three categories and 2096 background images, a sample of which is shown below. The precision and recall plots and the displayed image rank list for train data of the people class are also shown for both the linear as well as Hellinger case.

It can be seen from the comparison tables 1 and 2 that Hellinger kernel for the given same database of images performs better compared to the linear kernel mapping of the SVM. We can also infer that the more the number of positive images in the database, the more the retrieval is possible for a given class of data. Here we have used the value of cost parameter to be 100 which performs well in both the cases. There is a significant difference between the training and test performance, which can often be reduced, and the test performance (generalization) improved by cross validating the SVM C parameter. The precision and recall plots of the people class for both the cases indicate the performance efficiency of Hellinger kernel as 76.86 % on testing as compared to linear kernel which is only 71.04%. Also the training values are way better for Hellinger kernel which is 99.97 % as that of 96.01 % of linear kernel SVM.

TABLE 3
RESULTS WITH HELLINGER KERNEL

Classes	Test AP	No. of images retrieved in Top 36	PR on train	PR on test
Airplane	0.66	30	100	65.62
Spatial tiling off	0.67	32	92.53	66.76
Histograms Normalized	0.65	30	100	65.23
Varying fraction of training images	0.65	30	100	65.23
Bikes	0.69	34	100	69.42
Spatial tiling off	0.61	32	86.91	60.75
Histograms Normalized	0.69	33	100	68.84
Varying fraction of training images	0.69	33	100	68.84
People	0.77	34	100	76.77
Spatial tiling off	0.78	34	90.74	77.67
Histograms Normalized	0.77	34	100	76.86
Varying fraction of training images	0.77	34	100	76.86

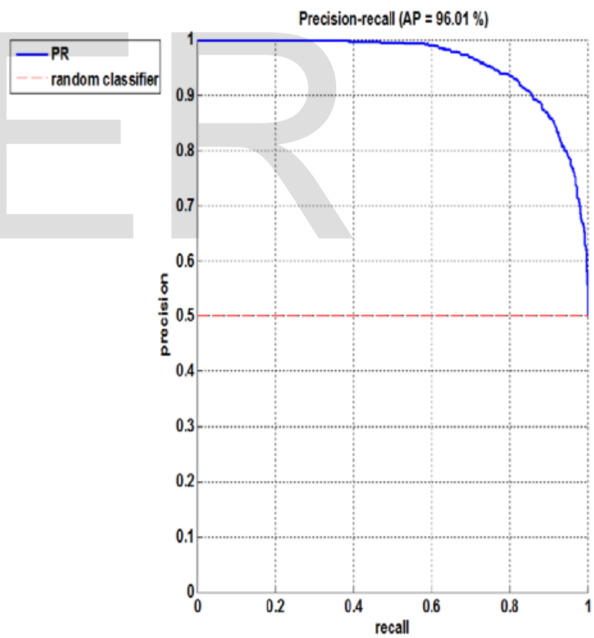

Fig 7 Precision Recall curve of training for People class with linear kernel and Histograms Normalized.

ISSN 2229-5518

Fig 8 Ranked Image List Retrieved in Top 36 over testing with linear kernel

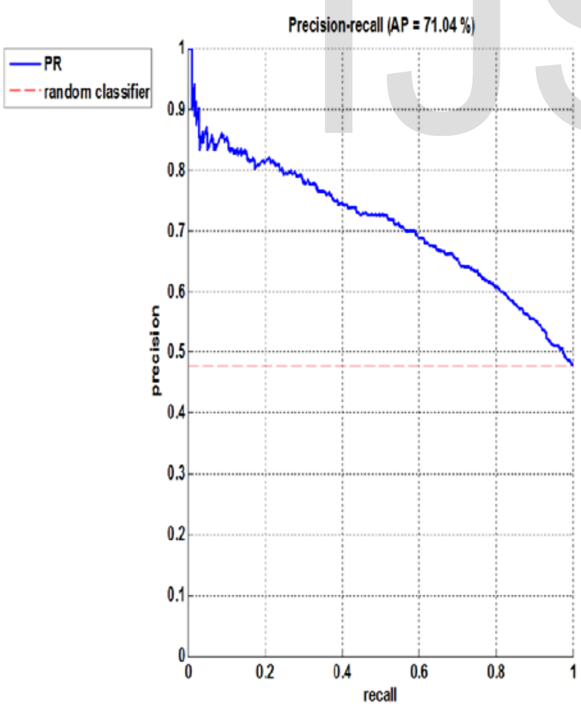

Fig 9 Precision Recall curve on Testing for people class with linear kernel and histograms normalized

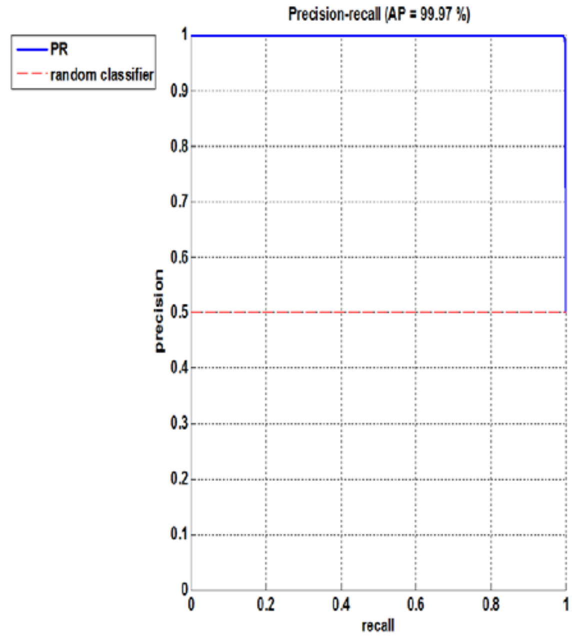

Fig 10 Precision Recall curve on training for People class with Hellinger kernel and Histograms Normalized

Fig 11 Ranked Image List Retrieved in Top 36 over testing with Hellinger kernel

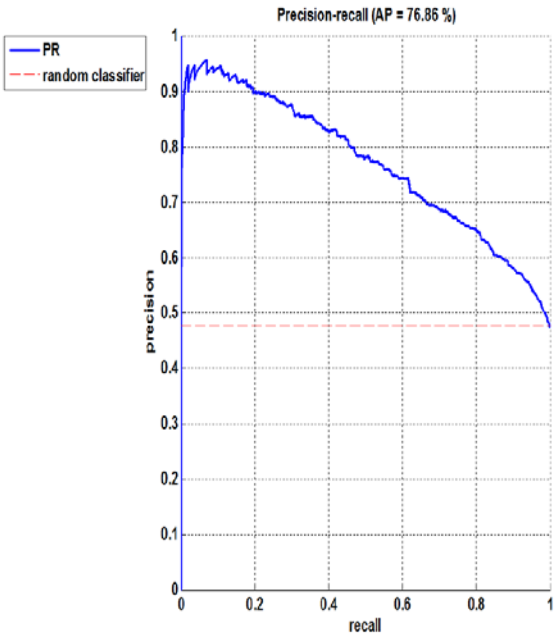

Fig 12 Precision Recall curve on Testing for people class with Hellinger kernel and histograms normalized

7 Conclusion

We have proposed a paper which compares the performance of linear kernel mapping of SVM with Hellinger kernel. The Hellinger kernel which is an explicit computation in the feature space serves better than the linear kernel mapping and the results are found to be more efficient in both training and testing phases and in different conditions. The superiority lies in the fact that Hellinger kernel values are computed as they remain linear in the new feature map so that it is put up with the advantages of linear mapping with enhanced performance.

8 References

[1] Andrea Vedaldi, Andrew Zisserman. "Efficient Additive Kernels via Explicit Feature Maps". IEEE transactions on pattern analysis and machine intelligence, vol. xx, no. xx, June 2011.

[2] U.C. Berkeley, Alexander C. Berg, Jitendra Malik. "Classification Using Intersection Kernel Support Vector Machines is Efficient".

[3] Marta Rojas, Inmaculada D´opido, Antonio Plaza and Paol Gamba. "Comparison of Support Vector Machine-Based Processing Chains for Hyper spectral Image Classification". Satellite Data Compression, Communications, and Processing VI, edited by Bormin Huang, Antonio J. Plaza, Joan Serra-Sagristà, Chulhee Lee, Yunsong Li, Shen-En Qian, Proc. of SPIE Vol. 7810, 78100B.

[4] Durgesh k. srivastava, lekha bhambhu. " Data Classification Using Support Vector Machine". Journal of Theoretical and Applied Information Technology.

[5] Yanni Wang, Bao-Gang Hu. "Hierarchical Image Classification Using Support Vector Machines". ACCV2002: The 5th Asian Conference on Computer Vision, 23–25 January 2002, Melbourne, Australia.

[6] Juan C. Caicedo, Angel Cruz and Fabio A. Gonzalez. "Histopathology Image Classification using Bag of Features and Kernel Functions". Bioingenium Research Group National University of Colombia.

[7] Claudio Cusano, Gianluigi Ciocca, Raimondo Schettini. "Image annotation using SVM".

[8] Gidudu Anthony, Marwala Tshilidzi, Hulley Gregg. "Image Classification Using SVMs One-against-One Vs One-against-All".

[9] Xiaohong Yu and Hong Liu. "Image Semantic Classification Using SVM in Image Retrieval". ISBN 978-952-5726-07-7 (Print), 978-952-5726-08-4 (CD-ROM) Proceedings of the Second Symposium International Computer Science and Computational Technology(ISCSCT '09) Huangshan, P. R. China, 26-28,Dec. 2009, pp. 458-461.

[10] Guggen Berger. "Another Introduction to Support Vector Machines".

[11] S. Boughorbel, J.-P. Tarel and N. Boujemaa. "Generalized Histogram Intersection Kernel for Image Recognition". In ICIP, Genova, Italy, 2005.

[12] M.Herbster." Learning Additive Models Online with Fast Evaluating Kernels". In Fourteenth Annual Conference on Computational Learning Theory, volume 2111, pages 444–460, 2001.

[13] S.S.Keerthi, O.Chapelle, and D.DeCoste. "Building Support Vector Machines with Reduced Classifier Complexity". JMLR, 7:1493–1515, 2006.

[14] E.E.Osuna and F.Girosi." Reducing the Run-time Complexity in Support Vector Machines". Advances in kernel methods: support vector learning, pages 271–283, 1999.

[15] Olivier Chapelle, Patrick Haffner and Vladimir Vapnik. "SVMs for Histogram-Based Image Classification".

[16]VLFeat - An open and portable library of computer vision Algorithms