International Journal on Emerging Technologies **2**(1): 67-70(2011) **ISSN : 0975-8364**

Face Recognition Using Support Vector Machines

Soheb Munir*, Vikas Gupta*, Sandeep Nemade* and M. Zahid Alam**

Department of Electronics and Communication,
**Technocrates Institute of Technology, Bhopal, (MP)*
***Lakshmi Narain College of Technology, Bhopal, (MP)*

(Received 11 January, 2011 Accepted 10 February, 2011)

ABSTRACT : An approach to multi-view face detection based on head pose estimation is presented in this paper. Support Vector Regression is employed to solve the problem of pose estimation .Three methods, the eigenface method, the Support Vector Machine (SVM) based method, and a combination of t he two methods, are investigated. The eigenface method,which seeks to estimate the overall probability distribution of patterns to be recognised, is fast but less accurate because of the overlap of confidence distributions between face and non-face classes. On the other hand, the SVM method, which tries to model the boundary of two classes to be classified, is more accurate but slower as the number of Support Vectors is normally large. The combined method can achieve an improved performance by speeding up the computation and keeping the accuracy to a preset level. It can be used to automatically detect and track faces in faceverification and identificationsystems.

Keywords : SupportVector Machine (SVM), face detection, eigen face method risk minimization.

I. INTRODUCTION

Support Vector Machines (SVM) have their roots in Statistical Learning Theory. They have been widely applied to machine vision fields such as character, handwriting digit and textrecognition and more recently to satellite image classification. Svms, like Artificial Neural Networks and other nonparametric classifiers have a reputation for being robust. SVMs function by nonlinearly projecting the training data in the input space to a feature space of higher (infinite) dimension by use of a kernel function. This results in a linearly separable dataset that can be separated by a linear classifier [1]. This process enables the classification of remote sensing datasets which are usually nonlinearly separable in the input space. In many instances, classification in high dimension feature spaces results in over-fitting in the input space, however, in svms over-fitting is controlled through the principle of structural risk minimization.

The empirical risk of misclassification is minimised by maximizing the margin between the data points and the decision boundary. In practice this criterion is softened to the minimisation of a cost factor involving both the complexity of the classifier and the degree to which marginal points are misclassified. The tradeoff between these factors is managed through a margin of error parameter (usually designated C) which is tuned through cross-validation procedures [2]. The functions used to project the data from input space to feature space are sometimes called kernels (or kernel machines), examples of which include polynomial, Gaussian (more commonly referred to as radial basis functions) and quadratic functions. Each function has unique parameters which have to be determined prior to classification and they are also usually determined through a cross validation process. A deeper mathematical treatise of svms can be found in Christianini (2002), Campbell (2000) and Vapnik (1995).

II. SVM MULTICLASS STRATEGIES

As mentioned before, SVM classification is essentially a binary (two-class) classification technique, which has to be modified to handle the multiclass tasks in real world situations e.g.derivation of land cover information from satelliteimages. Two of the common methods to enable this adaptation include the 1A1 and 1AA techniques. The 1AA approach represents the earliest and most common SVM multiclass approach (Melgani and Bruzzone, 2004) and involves the division of an N class dataset into N two-class cases. If say the classes of interest in a satellite image include water, vegetation and built up areas, classification would be effected by classifying water against non-water areas *i.e.* (vegetation and built up areas) or vegetation against non-vegetative areas *i.e.* (water and built up areas) [3]. The 1A1 approach on the other hand involves constructing a machine for each pair of classes resulting in N(N-1)/2 machines. When applied to a test point, each classification gives one vote to the winning class and the point is labeled with the class having most votes. This approach can be further modified to give weighting to the voting process. From machine learning theory, it is acknowledged that the disadvantage the 1AA approach has over 1A1 is that its performance can be compromised due to unbalanced training datasets, however, the 1A1 approach is more computationally intensive since the results of more SVM pairs ought to be computed. In this paper, the performance of these two techniques are compared and evaluated to establish their performance on the extraction of land cover information from satellite images [4].

III. METHODOLOGY

The study area was extracted from a 2001 Landsat scene (row 171 and row 60). It is located at thesource of River Nile in Jinja, Uganda. The bands used in this research consisted of Landsat's opticalbands i.e. bands 1, 2, 3, 4, 5 and 7. The classes of interest were built up area, vegetation andwater. IDRISI Andes was used for preliminary data preparation such as sectioning out of the study area from the whole scene and identification of training data. This data was then exported into a form readable by MATLAB (Version 7) for further processing and to effect the classification process. The SVMs that were used included the Linear, Polynomial, Quadratic and Radio Basis Function (RBF) SVMs. Each classifier was employed to carry out 1AA and 1A1 classification. The classification results for both 1AA and 1A1 were then imported into IDRISI for georeferencing, GIS integration, accuracy assessment and derivation of land cover maps. The following four parameters formed the basis upon which the two multiclassification approaches were compared: Number of unclassified pixels, number of mixed pixels, final accuracy assessment and the 95% level of significance of the difference between overall accuracies of the two approaches (*i.e.* |Z| >1.96).

IV. GLOBAL APPROACH

Both global systems described in this paper consist of aface detection stage where the face is detected and extracted from an input image and a recognition stage where the person'sidentity is established.

A. Face Detection

In order to detect faces at different scales we first computed a resolution pyramid for the input image and then shifted a window over each image in the pyramid.

We applied two preprocessing steps to the gray images tocompensate for certain sources of image variations [5]. A best-fit intensity plane was subtracted from the gray valuesto compensate for cast shadows. Then histogram equalizationwas applied to remove variations in the image brightnessand contrast. The resulting gray values were normalizedto be in a range between 0 and 1 and were used as inputfeatures to a linear SVM classifier. Some detection resultsare shown in Fig. 1. The training data for the face detector were generated by rendering seven textured 3-D head models. The heads were rotated between ___Æ and __Æ in depth and illuminated by ambient light and a single directional light pointingtowards the center of the face. We generated 3,590 face images of size pixels. The negative training setinitially consisted of 10,209 non-face patterns randomly extracted from 502 non-face images. We expandedthe training set by bootstrapping [19] to 13,655 non-face patterns [6].

B. Recognisation

We implemented two global recognition systems. Both systems were based on the one-vs-all strategy for SVM multi-class classification described The first system had a linear SVM for every person in the database. Each SVM was trained to distinguish between all images of a single person (labeled) and all other images [7].

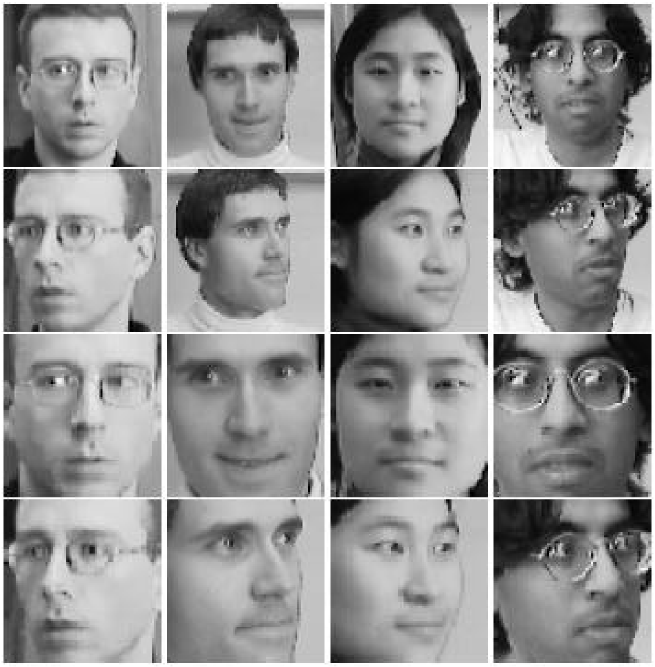

Fig. 1. The upper two rows are example images from our training set. The lower tworows show the image parts extracted by the SVM face detector.

For both training and testing we ran the face detector on the input image to extract the face. We re-saled the face image to pixels and converted the gray values into a feature vector 2. Given a set of people and a set of SVMs, each one associated to one person, the class label of a face pattern is computed as follows :

$$y = \begin{cases} n \text{ if } d_n(x) + t > 0 \\ 0 \text{ if } d_n(x) + t \pounds 0 \end{cases}$$

$$\text{with } d_n(x) = \max\{d_i(x)\}_{i=1}^{0}$$

We implemented a two-level component-based face detector which is described in detail [8]. The principles of the system are illustrated in Fig. 1. On the first level, component classifiers independently detected facial components. On the second level, a geometrical configuration classifier performed the final face detection by combining the results of the component classifiers. Given a window, the maximum continuous outputs of the componentclassifiers within rectangular search regions around theexpected positions of the components were used as inputsto the geometrical configuration classifier. The search regionshave been calculated from the mean and standard deviation of the components' locations in the training images [9]. We also provided the geometrical classifier with the precisepositions of the detected components relative to the upperleft corner of the window. The 14 facial componentsused in the detection system are shown in Fig. 4 (*a*).

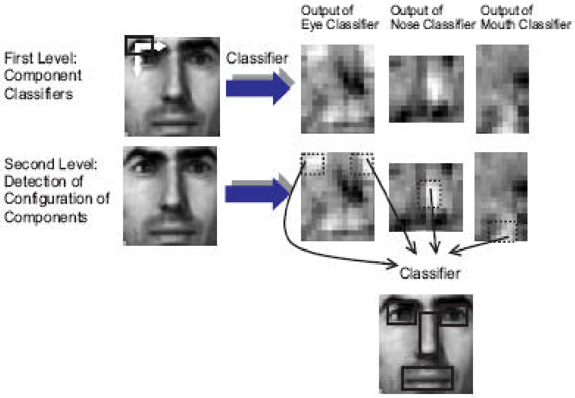
Output of Eye Classifier
Output of Nose Classifier
Output of Mouth Classifier
First Level: Component Classifiers
Classifier
Second Level: Detection of Configuration of Components
Classifier

Fig. 2. System overview of the componentbasedface detector using four components.On the first level, windows of the size of the components (solid lined boxes) are shiftedover the face image and classified by the componentclassifiers. On the second level, the maximum outputs of the component classifierswithin predefined search regions (dottedlined boxes) and the positions of the detected components are fed into the geometrical configuration classifier.

V. RESULTS

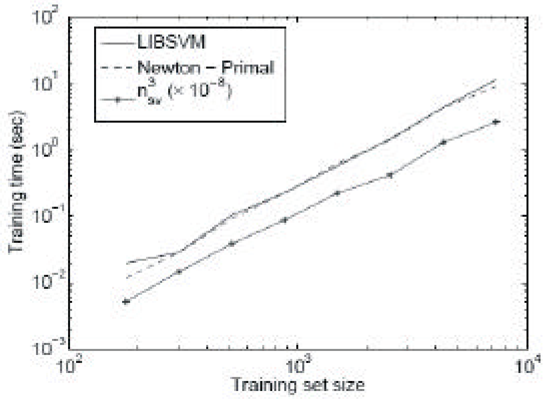

Fig. 3. With the L2 penalization of the slacks, the parallel between dualoptimization and primal Newton optimization is striking: the training timesare almost the same (and scale in O (n3sv)). Note that both solutions are exactly the same.

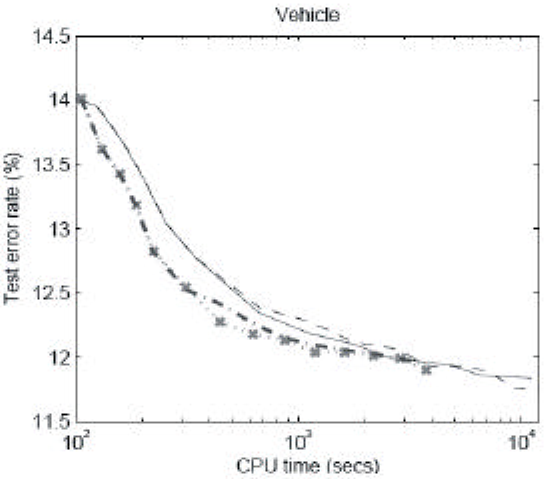

Fig. 4. KMP vs SpSVM (with/without regularization) on *M3V8, M3V other & Vehicle.*

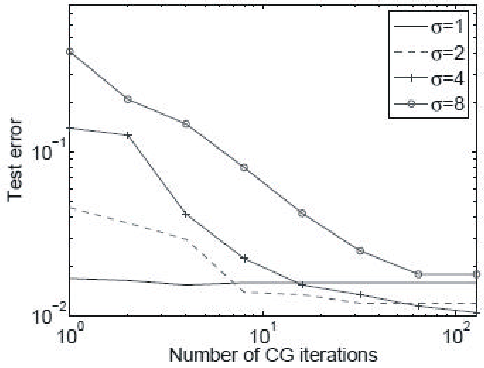

Fig. 5. Otimization of the objective function (2.13) by conjugate.

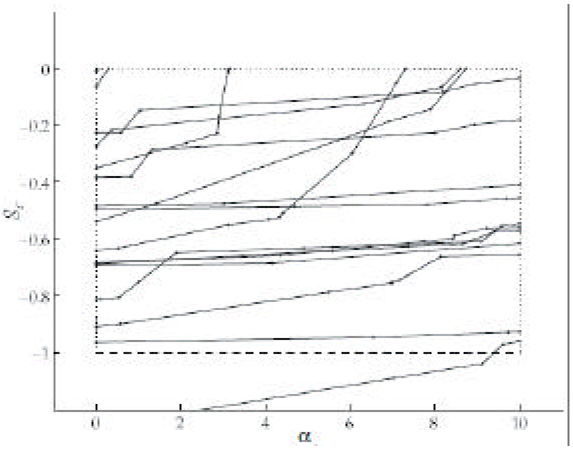

Fig. 6. Trajectory of LOO margin Idas a function of leave-one-out coefficient.

VI. CONCLUSION

We presented a component-based technique and two global techniques for face recognition and evaluated their performance with respect to robustness against pose changes. The component-based system detected and extracted a set of 10 facial components and arranged them in a single feature vector that was classified by linear SVMs. In both global systems we detected the whole face, extracted it from the image and used it as input to the classifiers. The first global system consisted of a single SVM for each person in the database. In the second system we clustered the database of each person and trained a set of view-specific SVM classifiers.

We tested the systems on a database which included faces rotated in depth up to about 0. In all experiments the component-based system outperformed the global systems even though we used more powerful classifiers (*i.e.* non-linear instead of linear SVMs) for the global system. This shows that using facial components instead of the whole face pattern as input features significantly simplifies the task of face recognition

REFRENCES

[1] P. Belhumeur, P. Hespanha, and D. Kriegman. Eigenfaces vs fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, (1997).

[2] D. J. Beymer. Face recognition under varying pose. A.I. Memo 1461, Center for Biological and Computational Learning, M.I.T., Cambridge, MA, (1993).

[3] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, (1993).

[4] R. Chellapa, C.Wilson, and S. Sirohey. Human and machine recognition of faces: a survey. *Proceedings of the IEEE*, 83(5):705–741, (1995).

[5] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:1–25, (1995).

[6] M. Fleming and G. Cottrell. Categorization of faces using unsupervised feature extraction. In *Proc. IEEE IJCNN International Joint Conference on Neural Networks*, pages 65– 70, (1990).

[7] G. Guodong, S. Li, and C. Kapluk. Face recognition by support vector machines. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, pages 196–201, (2000).

[8] B. Heisele, T. Poggio, and M. Pontil. Face detection in still gray images. AI Memo 1687, Center for Biological and Computational Learning, MIT, Cambridge, MA, (2000).

[9] K. Jonsson, J. Matas, J. Kittler, and Y. Li. Learning support vectors for face verification and recognition. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, pages 208–213, (2000).

[10] A. Lanitis, C. Taylor, and T. Cootes. Automatic terpretation and coding of face images using flexible models. *IEEE Transactions on Pattern Analysis and Machine ntelligence*, 19(7):743 –756,)1997).