COMPACT SUPPORT VECTOR REPRESENTATION

Jeff Fortuna and David Capson

Dept. of Electrical and Computer Engineering
McMaster University
Hamilton, Ontario, Canada
email: capson@mcmaster.ca

ABSTRACT

An algorithm that discovers a compact data representation for support vector classification is presented. The algorithm finds a basis which reduces the volume occupied by the coefficients in subspace. This volume reduction is driven by the support vectors of a support vector machine. A compact support vector representation (CSVR) of this form is shown to exhibit good generalization in the form of large margin and a small number of support vectors, while achieving low classification error rates. The compact nature of the data representation is shown to be particularly effective in representing correlated image sets such as those found in databases where faces and objects are imaged under varying lighting or pose.

1. INTRODUCTION

The classification of images from databases has attracted particular interest from pattern recognition researchers. The difficulty of determining salient features from images provides a challenge for feature extraction. Additionally, classification presents unique challenges for image features when little can be guaranteed about the underlying statistical distribution of the data.

Due to the large amounts of data available in raw image sets, subspace methods have become attractive for extracting features from such data. Principal component analysis (PCA) [1], independent component analysis (ICA) [2] or kernel principal component analysis (KPCA) [3] provide the advantage of re-representing the large amount of data present in images by a much smaller, statistically derived set of coefficients.

Support vector machines have emerged as a preeminent method for classification. Recently, some examination has been given to the selection of features which are particularly amenable to support vector classification. This follows the adage that improved generalization of the classification occurs if the distribution of the data is aligned with the non-linear function of the separating hyperplane. For example, in the particular case of the support vector data description [4],[5], a spherical data representation of minimal volume is desirable.

Herein, a representation of image data is found which, due to its compact size and simple shape, achieves large margins and a small number of support vectors, thus providing good generalization in classification. The compact support vector representation (CSVR) algorithm developed iteratively produces a set of basis vectors which provide subspace coefficients that re-represent the image set in this compact form. Information from the support vectors drives the modification of the basis. The algorithm is shown to often converge to the maximum achievable margin, with a very small number of support vectors. A brief background for support vector classification is provided, followed by a detailed description of the proposed algorithm. Statistical results of its operation are provided on the COIL object database [6] and Yale Face Database B [7], for two class classification problems, along with a brief discussion on convergence.

This work was supported by the Canadian Networks of Centers of Excellence Program (IRIS), the National Sciences and Engineering Research Council (NSERC) and the McMaster Manufacturing Research Institute (MMRI)

1.1. Support Vector Classification

To perform classification with a linear SVM, a labeled set of features $\{\mathbf{x}_i, y_i\}$ is constructed for all l features in the training dataset. The class of feature $\mathbf{c}_i$ is defined by $y_i = \{1, -1\}$.If the training data follows:

$$y_i(\mathbf{x}_i\mathbf{w} + b) - 1 \geq 0 \qquad \forall i \tag{1}$$

then the points for which the above equality holds lie on the hyperplanes $\mathbf{x}_i\mathbf{w} + b = 1$ and $\mathbf{x}_i\mathbf{w} + b = -1$. The SVM attempts to find the pair of hyperplanes which gives the maximum margin by minimizing $\|\mathbf{w}\|^2$ subject to constraints on $\mathbf{w}$. Reformulating the problem using the Lagrangian, the expression to optimize for a non-linear SVM can be written as [9]:

$$L(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{2}$$

$K(\mathbf{x}, \mathbf{x}')$ is a kernel function satisfying Mercer's conditions. An example kernel function (the one used herein) is the Gaussian radial basis function:

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma^2}\right) \tag{3}$$

where σ is the standard deviation of the kernel's exponential function. An estimation of classification error for the leave-one-out method of training can be made by [9]:

$$E[P(error)] = \frac{SV}{N} \tag{4}$$

where SV is the number of support vectors, N is the number of training data items, and $P()$ is the probability. Thus, if we reduce the number of support vectors, we achieve better generalization, since the reduced number of support vectors can still reproduce the same hyperplane. Error bounds based on the number of support vectors and the margin are described in [10].

1.2. Geometric Margin

For any given dataset, the geometric margin γ is the functional margin of the classifier with $\mathbf{w}$ normalized. Thus,

$$\gamma = \frac{1}{2}\left[\left[\frac{\mathbf{w}}{||\mathbf{w}||_2}\mathbf{x}^+\right] - \left[\frac{\mathbf{w}}{||\mathbf{w}||_2}\mathbf{x}^-\right]\right] = \frac{1}{||w||_2} \quad (5)$$

By reducing the volume of the dataset in such a way that the shape of the data becomes more regular, we can then increase the geometric margin and reduce the number of the support vectors. The maximum achievable geometric margin (separation on both sides of the hyperplane) is then $\frac{2}{\sqrt{2}}$ as $\mathbf{w}$ is normalized. This occurs with the hyperplane tilted at 45 degrees to the origin.

1.3. Feature Scaling By Coefficient Modification

In [4] the support vector data description was developed as a minimum volume containing all objects of the dataset. This minimum volume representation, which improves generalization, can be exploited provided that the data is rescaled, as in [5]. Another option for minimizing the volume of the feature space is to use the support vectors as an indication of the outer bounds of the feature space and to move the data toward the class mean (by an amount proportional to the support vector coefficients) to shrink the volume in this direction. A subspace that achieves this smaller volume can thus be learned from the modified coefficients. A basis can be found by linear regression. In this paper, the regression was performed by canonical correlation.

Using the class means:

$$\mathbf{C}_n = \frac{1}{l/2}\sum_{i=1}^{l/2}\mathbf{X}_{LD_{i,C_n}} \quad (6)$$

where $\mathbf{X}_{LD}$ subspace learned at each step of the iterative algorithm, a matrix of the class means,

$$\mathbf{X}_{LD_{mean}} = \left[\begin{array}{cccccc} \mathbf{C}_1^1 & \cdots & \mathbf{C}_1^{l/2} & \mathbf{C}_n^1 & \cdots & \mathbf{C}_n^{l/2} \end{array}\right] \quad (7)$$

a matrix scalar of the support vector coefficients α_i and an initial $m \times n$ matrix S_0,,

$$\mathbf{\Lambda} = \begin{bmatrix} \alpha_1 & & & \\ & \alpha_2 & & \\ & & \ddots & \\ & & & \alpha_n \end{bmatrix} \qquad \mathbf{S_0} = \begin{bmatrix} \mathbf{I} \\ \\ \mathbf{0} \end{bmatrix} \quad (8)$$

where m is the length of the basis vectors and n is the dimensionality of the subspace, boundary feature vectors can be moved toward their class means and basis vectors $\mathbf{S}$ can be learned to fit the new features.

2. CSVR ALGORITHM

1: initialize $\mathbf{S}$ as $\mathbf{S_0}$.
2: initialize:

$$\mathbf{X}_{LD_{train}} \leftarrow \mathbf{S}^T\mathbf{X}_{train}$$
$$\mathbf{X}_{LD_{test}} \leftarrow \mathbf{S}^T\mathbf{X}_{test}$$

3: initialize $\mathbf{\Lambda}$ to the identity matrix.
4: **repeat**
5: move the support vectors toward the mean by an amount proportional to the support vector α by:

$$\mathbf{X}_{LD_{train}} \leftarrow \mathbf{X}_{LD_{train}} - \mathbf{\Lambda}(\mathbf{X}_{LD_{train}} - \mathbf{X}_{LD_{mean}})$$

6: recalculate $\mathbf{S}$ by:

$$\mathbf{S} \leftarrow (\mathbf{X}^T\mathbf{X})^{-1})\mathbf{X}^T(\mathbf{X}_{LD_{train}}\mathbf{U})\mathbf{U}^+$$

where $+$ denotes pseudo-inverse and $\mathbf{U}$ are the left singular vectors of the generalized SVD of $\mathbf{X}$ and $\mathbf{X}_{LD_{train}}$ (canonical correlation regression)
7: calculate:

$$\mathbf{X}_{LD_{test}} \leftarrow \mathbf{S}^T\mathbf{X}_{test}$$
$$\mathbf{X}_{LD_{train}} \leftarrow \mathbf{S}^T\mathbf{X}_{train}$$

8: define data pairs $(\mathbf{x}_{LD_{train\,i}}, y_i)$ and apply a support vector classifier to classify $\mathbf{X}_{LD_{test}}$.
9: **until** (margin change $< .0001$) or (margin > 1.35).

2.1. Description of CSVR Algorithm

The steps of the CSVR algorithm are summarized in the steps above. To initialize, the basis vectors $\mathbf{S}$ are found from the training data $\mathbf{X}_{train}$ using principal or independent components, the low-dimensional data representations are found ($\mathbf{X}_{LD_{train}}$ and $\mathbf{X}_{LD_{test}}$) and $\mathbf{\Lambda}$ is set to the identity matrix (lines 1, 2, and 3).

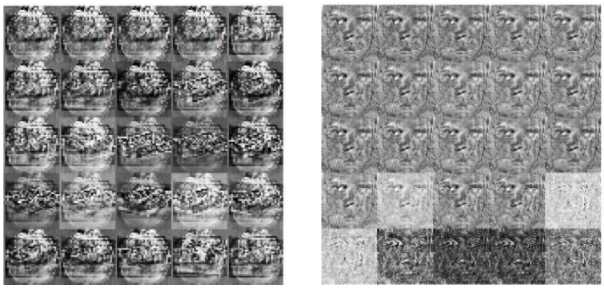
The first step in the iteration (line 5) moves all training data toward its class mean by an amount proportional to its support vector coefficient α. Support vectors will be set to the class mean for $\alpha = 1$ and the majority of the rest of the training data will be unmodified. The basis vectors $\mathbf{S}$ are then calculated to fit the modified training data set through canonical correlation regression, as shown in line 6. In line 7, the new test $\mathbf{X}_{LD_{test}}$ and training $\mathbf{X}_{LD_{train}}$ data sets are derived from their projections into the modified basis vectors. In the final step in the iteration (line 8), the newly calculated test and training subspace coefficients are classified by a SVC, which provides a new $\mathbf{\Lambda}$ for the next iteration.

After each iteration, the classes become more compact, with less effect from outlying points (which have been previously moved toward the mean) and the basis is learned from the regression on the coefficients. The compactness of the classes is illustrated by a steady increase in margin and the simplified shape of the classes is exemplified by a steady decrease in the number of support vectors. At the point where no further improvement in margin occurs, few data points are moved, since the number of support vectors has reached a small value. This condition terminates the iteration (line 9). The margin change termination threshold (0.001) was chosen empirically. In a large number of cases, the maximum achievable margin ($\frac{2}{\sqrt{2}} = 1.41$) is reached. In this case, the algorithm is terminated slightly early (1.35), which provides a significant decrease in iteration time.

3. TWO CLASS RECOGNITION WITH CSVR

3.1. Object Database - Pose Variance

The CSVR algorithm was tested on general objects from the COIL database. 30 objects under poses ranging from 0 to 355 degrees were classified with a soft margin support vector classifier with a large value of C=100. This yielded 435 two class classification



(a) COIL (b) YALE

Fig. 1. Basis images (Brightness and contrast have been enhanced)

examples. The training data consisted of objects at poses taken every 10 degrees starting from 0 degrees. The test data used objects at poses taken every 10 degrees starting from 5 degrees. The dimensionality of the learned subspace was 25. Recognition performance (margin, number of support vectors and error rate) is tested for the raw data for each subject pair for kernel σ ranging from 1 to 50.

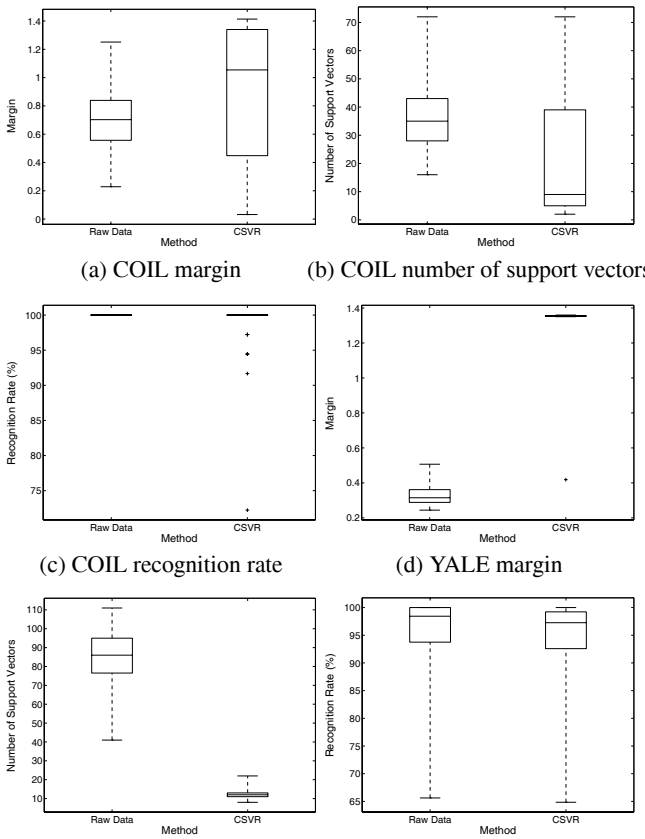
3.2. Face Database - Pose and Lighting Variance

To demonstrate the CSVR algorithm for face images, Yale Face Database B is employed. The database contains 10 subjects imaged under 9 different poses and 64 lighting positions. For this experiment, multiple 2 class recognition experiments are performed with the SVM (C = 100) over 36 pairs of subjects. For each pair of subjects, a training data set is constructed from the first 32 lighting positions for the poses 1 and 2 of each subject. The test data set comprised the same pair of subjects imaged under the last 32 lighting positions from the poses 7 and 8. As such, the recognition will therefore require some degree of both lighting and pose invariance. The training and test images were histogram equalized and mean centered before subspace calculation and classification.

The resulting basis images for the training images are shown in Figure 1 for both the COIL and Yale experiments. The recognition results for the kernel sigma which yielded the largest margin of the raw data and the results after the termination of the CSVR algorithm for both databases are shown in Figure 2.

3.3. Convergence

To illustrate the convergence of the algorithm, the volumes of the classes, margin, and number of support vectors were plotted as an average across all test cases. Convergence, of course, occurred at a different number of iterations for each test. Thus, to find an average, the last value converged to was extended to the largest number of iterations taken (401 iterations for the COIL database and 72 iterations for the Yale database). Volume for each class was estimated by sum of the the absolute distances of each subspace data point to its class center. The average convergence characteristics for the COIL and YALE test is shown in Figure 3. The maximum achievable geometric margin, $\sqrt{2}$ (see Equation 5) is shown as a dashed line on the mean margin plots.



(a) COIL margin (b) COIL number of support vectors

(c) COIL recognition rate (d) YALE margin

(e) YALE number of support vectors (f) YALE recognition rate

Fig. 2. Box plots of results of classification of all pairs of classes

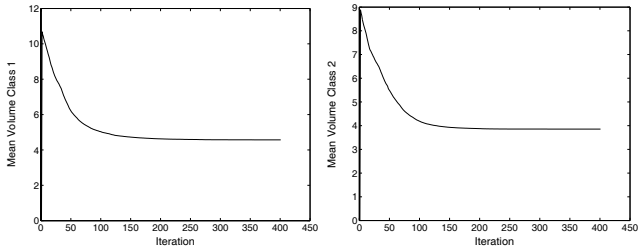
4. DISCUSSION

4.1. Choice Of SVM Parameter C

C was chosen as 100 empirically, however there was almost no change in the results (error rate, margin, or number of support vectors) over a very wide range of values, from C = 0.1 to C = 100 for both datasets. The largest value of C was chosen under the assumption that a heavy penalty for errors creates more complex decision boundaries making it easier to illustrate the reduction of complex decision boundaries into simpler ones. In any case, the same value of C was used for the raw data SVM and the CSVR SVM.

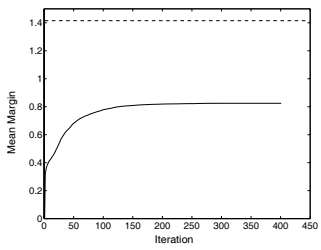
4.2. Raw Data and CSVR Results

For both databases, the average recognition rate was almost identical between the raw data and CSVR classification. However, substantial increases in margin and decreases in the number of support vectors resulted from the use of CSVR. This is a direct indication that the CSVR's ability to generalize effectively for data with characteristics typical of image databases. The basis images indicate a highly redundant coding, with a lot of the basis images exhibiting similarity. This is in sharp contrast to PCA, which provides a set of decorrelated bases. While this type of coding would be highly inefficient for image coding applications, redundant coding

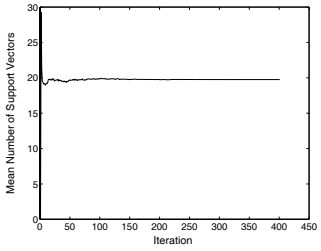


(a) COIL volume of class 1

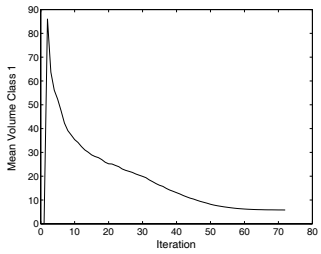
(b) COIL volume of class 2



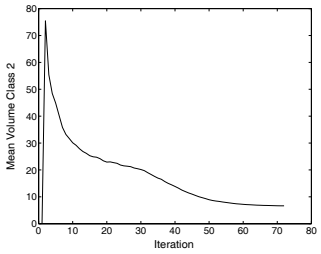
(c) COIL margin



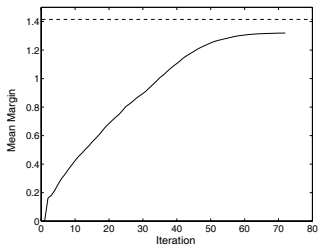
(d) COIL number of support vectors



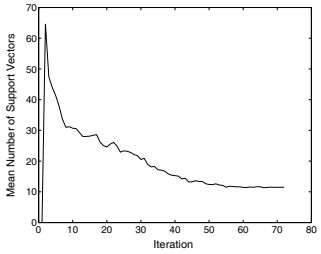
(e) YALE volume of class 1



(f) YALE volume of class 2



(g) YALE margin



(h) YALE number of support vectors

Fig. 3. Averages per iteration

is gaining ground for applications in image recognition.

4.3. Volume, Margin and Number of Support Vector Convergence

The averages of volume, margin and number of support vectors over multiple iterations show a strong relationship between these quantities. For the case of image databases, it appears that reduction in class volume in the direction of the class means provides an effective and well behaved way to regularize the class shapes. Average convergence occurred quite rapidly (exponentially) for both databases, after about 100 iterations. However, for a number of recognition tasks, where the class distributions may be highly irregular, or strongly multi-modal, it is possible that such regular shapes may not occur after iteration. It appears that the nature of correlated image sets is particularly amenable to the CSVR representation.

5. CONCLUSION

In this paper, a new iterative algorithm was developed with the intent of compacting the classes in a support vector representation. A set of basis vectors was learned from the modified support vectors to re-represent the data in a more compact form. This new representation, CSVR, was shown to exhibit better generalization characteristics than the raw data for two standard image databases. The algorithm was well behaved image datasets under a variety of pose and lighting conditions, with exponential convergence of margin and the number of support vectors.

6. REFERENCES

[1] M.A. Turk and A.P. Pentland, "Face Recognition Using Eigenfaces", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,* pp. 586–591, June 1991.

[2] M. Bartlett and T. Sejnowski, "Independent components of face images: A representation for face recognition", *Proceedings of the 4th Annual Joint Symposium on Neural Computation*, Pasadena, California, 1997.

[3] K. Kim, K. Jung and H. J. Kim, " Face Recognition using Kernel Principal Component Analysis" *IEEE Signal Processing Letters*, Vol. 9, No. 2, Feb. 2002, pp 40–42

[4] D. Tax and R. Duin, "Support Vector Domain Description",*Pattern Recognition Letters*, Vol.20, 1999, pp. 1119–1125.

[5] P. Juszczak, D. Tax, R. Duin, "Feature Scaling in Support Vector Data Description",*ASCI Conference 2002, Lochem, The Netherlands*, Vol.20, 1999, pp. 1119–1125.

[6] S. A. Nene, S. K. Nayar and H. Murase, "Columbia Object Image Library (COIL-100)", *Technical Report CUCS-006-96*, February, 1996

[7] A.S. Georghiades and P.N. Belhumeur and D.J. Kriegman, "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 6, 2001, pp 643–660.

[8] D.J. Field, "Relations between the statistics of natural images and the response properties of cortical cells", *Journal of the Optical Society of America*, Vol.4, No. 12, 1987, pp. 2379–2394.

[9] N. Cristianini and J. Shawe-Taylor, "An Introduction to Support Vector Machines", *Cambridge University Press*, Cambridge, UK, 2000.

[10] V. Vapnik, "The Nature of Statistical Learning Theory", *Springer-Verlag*, New York, USA, 2000.