Multi-Dimensional Features Reduction of PCA on SVM Classifier for Imaging Surveillance Application

Tan Chue Poh[1], Nur Fateha Muhamad Lani[1], Lai Weng Kin[1]
[1]Centre for Advanced Informatics, MIMOS Berhad,
Technology Park Malaysia, 57000 Kuala Lumpur,
MALAYSIA.
Email: chue.poh@mimos.my

Abstract: - This paper presents the application of multi dimensional feature reduction of Principal Component Analysis (PCA) and Support Vector Machine (SVM) classifier for imaging surveillance system. Recently, research in image processing has raised much interest in the security surveillance systems community. Weapon detection is one of the greatest challenges facing by the community recently. In order to overcome this issue, application of the popularly used SVM classifier is performed to focus on the need of detecting dangerous weapons. However, PCA is used to explore the usefulness of each feature and reduce the multi dimensional features to simplified features with no underlying hidden structure. In this paper, we take advantage of the simplified features and classifier to categorize images object with the hope to detect dangerous weapons effectively. In order to validate the effectiveness of the SVM classifier, several classifiers are used to compare the overall accuracy and computational speed of the system with the compliment from the features reduction of PCA. These classifiers include Neural Network, Decision Trees, Naïve Bayes and k-Nearest Neighbor methods. The final outcome of this research clearly indicates that SVM has the ability in improving the classification accuracy using the extracted features. Besides, it is also shown that PCA is able to speed-up the computational time with the reduced dimensionality of the features while maintaining the classification accuracy.

Keywords: Support Vector Machine, Classification, Imaging surveillance

1 Introduction

Security surveillance systems are becoming indispensable in scenarios where personal safety could be jeopardized due to criminal activities [1]. Conventional security surveillance systems require the constant attention of security personnel, who monitor several locations concurrently [2,3]. Hence, the advancement in image processing techniques has become an advantage to the security surveillance systems to improve on the operational activity for monitoring purpose.

Image classification is an essential process in image processing and its major issue lies in categorizing images with huge number of input features using traditional classification algorithm. These algorithms tend to produce unstable prediction models with low generalization performance [4]. To overcome high dimensionality, image classification usually relies on a pre-processing step, specifically to extract a reduced set of meaningful features from the initial set of huge number of input features. Recent advances in classification algorithm such as Support Vector Machine [5-7] has produced new methods that are able to handle more complex problems.

In this paper, we emphasize on the analysis and usage of the advanced classification method of Support Vector Machines to classify dangerous weapons within an image. In order to validate the effectiveness of the classifier, several classifiers such as Neural Network, Decision Trees, Naïve Bayes and *k*-Nearest Neighbor methods are utilized to compare the overall accuracy of the classifiers. Finally the study depicts the comparative analysis of different classification techniques with respect to the robustness of the meaningful extracted features. The classification process comprised of four steps, which are feature extraction, training, prediction and assessing the accuracy of the classification. Analysis on the features is done to ensure the robustness and usefulness of each feature to differentiate classes effectively. The details of the classification will be discussed in this paper.

This paper is divided into four sections. Section II presents the methodology and the dataset used in this paper. In this section, the basic concept of Principal Component Analysis, Support Vector Machine, Neural Network, Decision Trees, Naïve Bayes and *k*-Nearest Neighbor methods are discussed. Section III describes the results and discussion for the findings of the classification process using the aforementioned classifiers. The accuracy assessment with the comparisons between the classifiers is discussed in this section. In Section IV, we conclude this paper with the suggestion on future work.

2 Methodology

2.1 Data Description

In this paper, we utilized on a set of data which was available freely in the internet [8] to carry out some experimental research on the classification. We evaluated the selected algorithms using the training dataset which contains 13 attributes (features value of the image objects) with their associate class labels (Human, Bull, Child, Dog, Duck, Knife classes). Besides, 6 test dataset that contain the same features value of the image objects for each class have been identified. Feature extraction process was carried out to extract all useful features from 80 binary images (black and white images) for training dataset and 48 binary images for testing dataset to represent the characteristics of the image object. From the image analysis and feature extraction, 13 important and useful features of the image object as the attributes of the dataset were obtained. In this case, the extracted features must be robust enough and RST (rotation, scale and transition) invariant. A very adaptive feature would be RST-invariant, meaning that if the image object is rotated, shrunk or enlarge or translated, the value of the feature will not change. We took the invariance of each feature into consideration and the features comprised of compactness, elongation, axis ratio, hull ratio, aspect ratio, dispersedness, roughness, occupancy, breadth, moment, ratio between ellipse and area of the blob, ratio between area of the bounding box minus area of the blob and area of the bounding box, and ratio between major axis length and minor axis length. These features are obtained through OpenCV software tools.

2.2 Principal Component Analysis

Principal component analysis (PCA) is one of the most valuable multi dimensional features reduction products derived from the applied linear algebra. PCA is used abundantly because it is a simple and non-parametric method of extracting relevant information from complex data sets. The goal of PCA is to reduce the dimensionality of the data while retaining as much as possible of the variation in the original dataset.

Suppose $x_1,\ x_2,\ \dots x_N$ are N x 1 vectors

Step 1: Mean value is calculated with this equation :

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{1}$$

Step 2: Each features is used to subtract the mean value :

$$\Phi_i = x_i - \bar{x} \tag{2}$$

Step 3: Matrix $A = [\Phi_1 \quad \Phi_2 \quad \dots \quad \Phi_N]$ is generated with N x N matrix and covariance matrix with the same dimension size is computed as below:

$$C = \frac{1}{M}\sum_{i=1}^{N} \Phi_i \Phi_i^T = AA^T \tag{3}$$

The covariance matrix characterizes the distribution of the data.

Step 4: Eigenvalues is computed:

$$C = \lambda_1 > \lambda_2 > \dots > \lambda_N \tag{4}$$

Step 5: Eigenvector is computed:

$$C = u_1, u_2, \dots, u_N \tag{5}$$

Since C is symmetric, $u_1, u_2, \ldots, u_N$ form a basis, ($(x-\bar{x})$, can be written as a linear combination of the eigenvectors):

$$x-\bar{x} = b_i u_1 + b_2 u_2 + \ldots + b_N u_N = \sum_{i=1}^{N} t \quad (6)$$

Step 6: For dimensionality reduction, it keeps only the terms corresponding to the K largest eigenvalues:

$$x-\bar{x} = \sum_{i=1}^{K} b_i u_i \text{ where } K << N \quad (7)$$

The representation of $\hat{x}\,\tilde{.}\,x$ into the basis $u_1, u_2, \ldots, u_K$ is thus

$$\begin{bmatrix} b_i \\ b_2 \\ \ldots \\ b_K \end{bmatrix}. \quad (8)$$

2.3 Classification Methods

The aim is to do comparison of supervised classification methods for classification of the image object to their known class from the reduced multi-dimensional features dataset. The issue in identifying the most promising classification method to do pattern classification is still in research. Therefore, we are interested in predicting the most promising classification method for pattern classification in terms of the classification accuracy achieved in detecting dangerous weapons. The algorithms considered in this study are Support Vector Machines (SVM), Neural Network, Decision Trees, Naïve Bayes and *k*-Nearest Neighbor. The methodology for each classifier is presented with basic concept and background.

2.3.1 Support Vector Machine Classifier

The SVM algorithm is a machine learning technique based on statistical theory [5] that can be used for classification purposes. The aim of Support Vector Machine classifier is to find an ideal separating hyperplane in a higher dimensional feature space. For a given training sample belonging to two different classes, SVM derives a hyperplane, which is at a maximum distance from the closest points belonging to both the classes. To find the optimal separating hyperplane, assume that the two classes to be distinguished are linearly separable, and denote the input space X with input vectors, $\vec{x}$ and the training set $T_r = \{(x_1, y_1), \ldots, (x_N, y_N)\}$, where $x_i \in X$ and $y_i \in Y$, $Y = \{1, -1\}$. In practice, it will often be the case where the data cannot be separated linearly by means of a hyperplane. One of the basic ideas behind SVM is to have a mapping Φ from the original input space X into a high-dimensional feature space F.

The SVM method solves for

$$\min \|w\|^2 \quad (9)$$

with $\quad y_i(\langle \Phi(\vec{x}_i), w \rangle + b) \geq 1$ for $i = 1, \ldots, N \quad (10)$

where $\vec{w}$ is a vector perpendicular to the hyperplane while b determines the displacement of the hyperplane along the normal vector $\vec{w}$ [5]. To solve the constrained minimization problem, the Lagrangian dual problem method is introduced as

maximize

$$W(\lambda) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{N} \lambda_i \lambda_j y_i y_j \langle \Phi(\vec{x}_i), \Phi(\vec{x}_j) \rangle \quad (11)$$

subject to $\lambda_i \geq 0$, $i = 1, \ldots, N$ and $\sum_{i=1}^{m} \lambda_i y_i = 0 \quad (12)$

with Lagrange multipliers $\lambda_i \geq 0$ [7]. After solving this dual problem, the decision function implemented by the classifier for any test vectors x is expressed by

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{N} \lambda_i y_i \langle \Phi(\vec{x}), \Phi(\vec{x}_i) \rangle + b \right) [6] (13)$$

2.3.2 Neural Network Classifier

A neural network is a set of connected input/output units in which each connection has a weight associated with it [9]. During the learning phase, the network learns by adjusting the weights so as to be able to predict the correct class label of the input tuples. The learning of neural network is also referred to as connectionist learning due to the connections between units [10].

The classification by neural network algorithm is based on multilayer feed-forward neural network. The network topology is specified by the number of units in the input layer, the number of units in the hidden layer, and the number of units in the output layer. Then, normalization of the input values for each attribute measured in the training tuples is done to fall in the range between 0.0 and 0.1 to speed up the learning phase. The inputs to the network correspond to the features of the image objects. The inputs are fed simultaneously into the units making up the input layer. These inputs pass through the input layer and are then weighted and applied with an activation function. Then, they fed simultaneously to a second layer of units called hidden layer. The outputs of hidden layer units can be input to another hidden layer and so on. The weighted outputs of the last hidden layer are input to units making up the output layer which emits the network's prediction for given tuples.

2.3.3 Decision Tree Classifier

Decision tree induction is the learning of decision trees from class-labelled training tuples. A decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test and each leaf node (or terminal node) holds a class label [11]. Decision tree models are built by a process that is known as recursive partitioning [12].

The ultimate goal of building a tree model is to end up with the smallest tree that has the purest leaf nodes [11]. A leaf node where all the instances in it are correctly classified is superior to one where just of over half are correctly classified.

A commonly used technique is to choose a split that will create the largest and purest child nodes by only looking at the instances in that node. This technique is referred to the 'greedy' or the 'local optimization' approach [12]. Its advantage is that it is computationally efficient regardless of the size of the problem.

Each node will have an error rate *e*, which is the misclassified instance in it. The probability that a particular classification will be correct is then 1-e [11]. The probability of a correct prediction from the model is then the weighted average of these probabilities from each leaf. These estimates can be based on training data or on a separate test data used to validate the model.

2.3.4 Naïve Bayes Classifier

The naive Bayes classifier combines the naive Bayes probability model with a common decision rule known as the *maximum a posteriori* (MAP) decision rule [13]. Below is the classifier that is the function classify defined as follows:

$$\arg\max_c \; p(C=c)\prod p(F_i = f_i \mid C=c) \; [15] \quad (14)$$

where C with a small number of outcomes or *classes*, conditional on several feature variables F.

Naïve Bayesian classifier is considered 'naïve' because it assumes that the effect of a feature value on the respective class is independent of the values of the other features.

2.3.5 K-Nearest Neighbor Classifier

K-nearest neighbor algorithm is one of the simplest of all machine learning algorithms. A point in the space is assigned to a particular class if it is the most frequent class label among the *k* nearest training samples. In this study, Euclidean distance is used.

The training process in the algorithm involves storing the feature vectors and class labels of the training samples. During the classification stage, the test sample is represented as a vector in the feature space. Distances from the new vector to all stored vectors are computed and *k* closest samples are selected [14]. We predict the new vector to the most common class amongst the K-nearest neighbors to classify the new vector to a particular class.

3 Results and Discussion

The supervised classification algorithms (Support Vector Machine, Neural Network, Decision Trees, Naïve Bayes and k-Nearest Neighbor, PCA + Support Vector Machine, PCA + Neural Network, PCA + Decision Trees, PCA + Naïve Bayes and PCA + k-Nearest Neighbor) are applied to the training and test dataset. The classifiers are analyzed and compared and the accuracy assessment is shown in Figure 1. In this study, the model with the highest classification accuracy is considered as the best model for pattern classification of this dataset.

We identify the best model to predict which model is the best to be used in predicting the

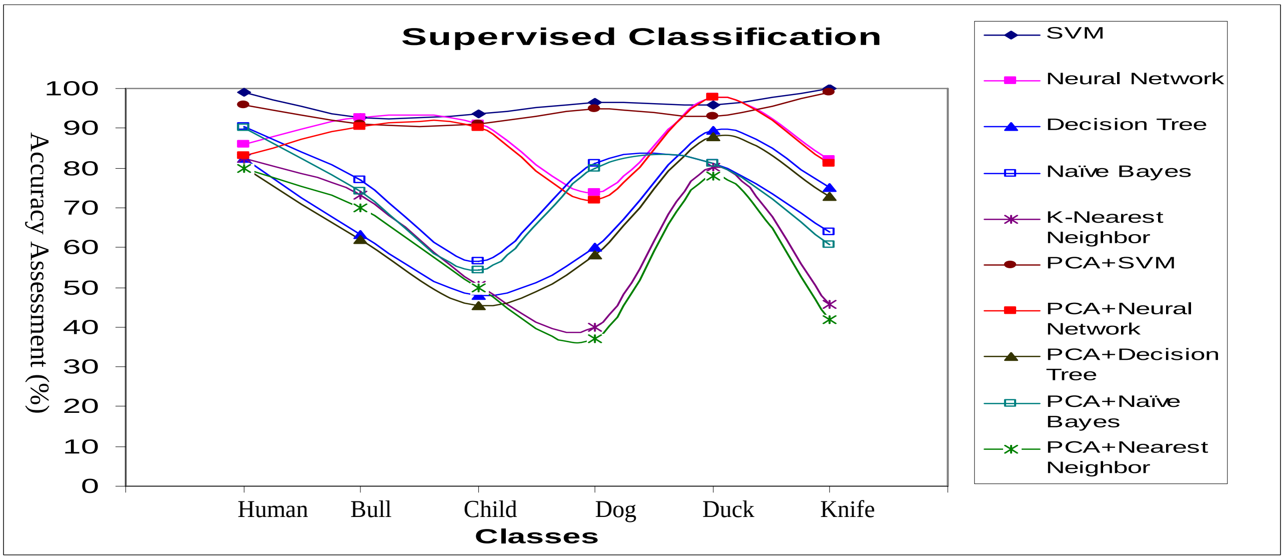

Fig. 1: Accuracy Assessment of the classifiers

future trends of the dataset. Table 1 shows impact of PCA for all the aforementioned classifiers by comparing the accuracy and computational speed (shown in bracket).

Table 1: Difference of accuracy (%) and computational speed (ms) for each classifier

Classifier (Difference in Computational Speed)	**Human**	**Bull**	**Child**	**Dog**	**Duck**	**Knife**
SVM (5.32ms)	1.98	1.75	1.69	1.42	1.51	1.15
Neural Network (3.38ms)	1.87	1.92	0.91	1.81	0.26	1.02
Decision Tree (2.55ms)	1.77	1.28	1.89	2.00	1.30	1.58
Naïve Bayes (2.32ms)	0.33	1.91	1.87	1.12	0.20	1.93
K-Nearest Neighbor (2.15ms)	2.35	3.21	0.46	2.91	2.37	3.83

Based on Fig. 1, we can see that Support Vector Machine classifier achieve the highest overall classification accuracy of all classes followed by Neural Network classifier,Naïve Bayes Classifier, Decision Tree Classifier,and K-Means K-Nearest Neighbor achieves the least average accuracy for all six classes. The dataset we used in this study is quite small and based on our research, the Support Vector Machines classifier is best applied to small dataset [6]. The knife class has been classified accurately by Support Vector Machine classifier with overall accuracy of 100%.On the other hand, classifiers with features generated from PCA provide slightly less accuracy and computational speed compared to the classifiers using the predefined number of features. This is due to the reduced dimensional features offered by PCA which allow only the useful key features to participate in the classification process.

4 Conclusion

The project is aimed to investigate the performance and impact of PCA on classification in the aspect of accuracy and computational speed. The potential of each classifier has been demonstrated and Support Vector Machine has shown a desirable result in detecting weapons compared to other classifiers. Our future work shall extend this work to multiple types images, extraction of appearance features and real-time signal data.

References

[1] A.T. Ali, and E.L. Dagless , "Computer vision for security surveillance and movement control", *IEE Colloquium on Electronic Images and Image Processing*

in Security and Forensic Science, May 1990, pp. 1-7.

[2] A.C.M. Fong, "Web-based intelligent surveillance systems for detection of criminal activities", *Journal of Computing and Control Engineering*, Vol 12, Issue 6, 2001, pp. 263-270.

[3] Yi-Tsung Chien, Yea-Shuan Huang, Sheng-Wen Jeng, Yao-Hong Tasi, and Hung-Xin Zhao, "A real-time security surveillance system for personal authentication", *IEEE 37th Annual 2003 International Carnahan Conference on Security Technology 2003*, 14-16 October 2003, pp. 190-195.

[4] P. Geurts, Contribution to decision tree induction: bias/ variance tradeoff and time series classification, PhD. Thesis, Department of Electrical Engineering and Computer Science, University of Liege, May 2002.

[5] Vapnik, V. The Nature of Statistical Learning Theory, *Springer Verlag*, 1999.

[6] Bernhard Schölkopf and Alexander J.Smola, Learning With Kernels, The MIT Press, Cambridge, Massachusetts, London, England., 2002.

[7] Nello Cristianini, and John Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, 2000.

[8] http://www.cs.sdce.edu/Matcher/.

[9] Y. Tang, G. W. William, and Y. H. Chia, "Application of a multilayer network in image object classification", *Proceedings of SPIE*, Vol 1469, 1991, pp. 113-120.

[10] O'Keefe, S.E.M. Austin J, "Image object labelling and classification using an associative memory", *5th International Conference of Image Processing and its Applications*, 1995, pp. 286-290.

[11] J. Zhang A. Silvescu and V. Honavar, Driven Induction of Decision Trees at Multiple Levels of Abstraction, Reformulation, and Approximation, *Lecture Notes in Artificial Intelligence 2371*, 2002, pp. 316-323.

[12] M. Núñez, The Use of Background Knowledge in Decision Tree Induction, *Machine Learning*, Vol. 6, 1991, pp. 231-250.

[13] I.H. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, October 1999.

[14] J. Han and M. Kamber, Data Mining Concepts and Techniques, 2nd Edition; Morgan Kauffman Publisher, 2000.

[15] N. Michael, Introduction to Artificial Intelligence, 2nd Edition, Addison Wesley, 2002.