Asian Journal of Information Technology 7 (5): 183-190, 2008
ISSN: 1682-3915

Boosting with Kernel Base Classifiers for Human Object Detection

[1]M. Rahmat Widyanto and [2]Chastine Fatichah
[1]Faculty of Computer Science, University of Indonesia, Depok Campus, West Java, Indonesia
[2]Department of Informatics, Faculty of Information Technology,
Institut Teknologi Sepuluh Nopember, East Java, Indonesia

Abstract: To improve the accuracy of Boosting for human object detection, Boosting with kernel base classifiers, called K-Boosting, is proposed. The proposed method uses kernel function rather than linear function, as in conventional Boosting, for base classifiers. The use of kernel function makes a better decision function therefore the accuracy is improved. Experiments on human object detection application show that the accuracy is 16% improved comparing to that of conventional Boosting. The accuracy of the proposed method is comparable to that of Support Vector Machine but the computational time is comparable to that of conventional Boosting. This proposed method is very useful for development of a real time human object detection.

Key words: Boosting, kernel, support vector machine, human object detection

INTRODUCTION

Human object detection is an important issue to address because of the many applications of human object detection system, i.e., surveillance system, visual search engine and intelligent vehicles (Sun *et al*., 2002; Papageorgiou and Poggio, 1999). Many interesting human object detection methods have been proposed in the literatures. Oren *et al*. (1997) use wavelet templates to detect human object. Papagergiou and Poggio (2000) pioneered the use of overcomplete sets of Haar wavelet features in combination with a Support Vector Machine (SVM). Mohan (1999), create 4 component classifiers for detecting heads, legs and left/right arms separately. The research in Sun *et al*. (2002), Papageorgiou and Poggio (1999, 2000), Oren *et al*. (1997) and Mohan (1999) used Support Vector Machine (SVM) method for human object detection. SVM method has high accuracy for human object detection but the speed of testing is slow.

Other method for human object detection is Boosting (Laptev, 2006; Arras *et al*., 2007; Viola *et al*., 2003). Laptev (2006) improves the accuracy object detection using boosted histograms. Arras *et al*. (2007) used boosted features for detecting people in 2D range data. Viola *et al*. (2003) used Boosting for movement human object detection. Boosting method has comparably low accuracy for human object detection but the speed of testing is fast. The real time applications of human object detection requires high accuracy of detecting human and high speed of testing. To overcome the problem, this research proposes Boosting with kernel base classifier for human object detection in order to improve the accuracy of the conventional boosting classifier.

DISCUSSION ON BOOSTING

Boosting (Bishop, 2006) is a powerful technique for combining multiple base classifiers to produce a form of committe whose performance can be significantly better than that of any the base classifier. The most widely used form of boosting algorithm called *AdaBoost*, shot for 'adaptive boosting', developed by Freud and Scaphire (1999). Boosting can give good results even if the base classifiers have a poor performance and hence sometimes the base classifers are known as *weak learners* (Bishop, 2006).

The principal of boosting is that the base classifiers are trained in sequence and each base classifier is trained using a weighted form of the data set where the weighting coeffecient associated with each data point depends on the performance of the previous classifiers.

In particular, points that are missclasiffied by one of the base classifiers are given greater weight when used to train the next classifier in the sequence. Once all the classifiers have been trained, their predicitons are then combined through a weighted majority voting scheme, as illustrated schematically in Fig. 1.

Corresponding Author: M. Rahmat Widyanto, Faculty of Computer Science, University of Indonesia, Depok Campus, West Java, Indonesia

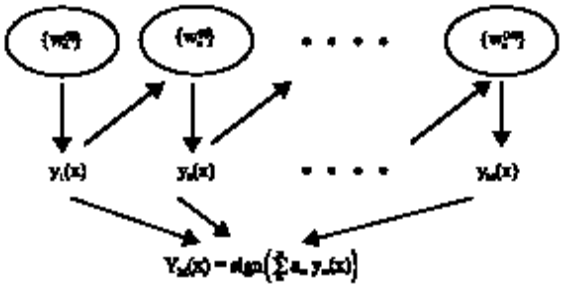


Fig. 1: Schematic illustration of the boosting framework

Consider a two-class classification problem, in which the training data comprises input vectors $x_1, \ldots, x_N$ along with corresponding binary target $t_1, t_2, \ldots, t_N$ where, $t_n \in \{-1,1\}$. Each data point is given an associated weighting parameter w_n which is initially set to 1/N for all data points.

At each stage of the algorithm, AdaBoost trains a new classifier using a dataset in which the weighting coefficients are adjusted according to the performance of the previously trained classifier so as to give greater weight to the misclassified data points. Finally, when desired number of base classifiers has been trained, they are combined to form a committee using coefficients that give different weight to different base classifiers. The precise form of the AdaBoost algorithm is given in Table 1.

Freund and Scaphire (1999) say that AdaBoost uses l_∞ norm for the y_m vector and l_1 norm for the weight vector. The l norms are instead of Euclidean to maximize a minimal margin. The difference between the norms can result in very large differences in the margin value. In this case, it can be shown that if the number of relevant weak classifiers m is a small fraction of the total number of weak classifiers then the margin associated with AdaBoost will be much large. Beside that AdaBoost corresponds only to linear programming for maximizing the margin. So, that the margin of hyperplane is not optimal especially to solve nonlinear problems.

This problem can decrease the accuracy of classification. As an example to see how the accuracy classification of AdaBoost, consider a human object detection problem. The training data of human object detection problem is shown in Fig. 2.

The training data is resulted from capturing image with 3-5 meter distance and cropping image with size 128×64 pixels. The positive samples are images of human with variability in different colors and garment types. The negative samples are images of natural scenery and buildings that are not contain any human. To evaluate the Boosting performance, the experiment uses 450 training dataset and 175 testing dataset. The experiment is

Table 1: AdaBoost algorithm

AdaBoost algorithm:

1. **Input**: a set of training data with labels
2. **Initialize**: the weight of training data: $w_n^{(n)} = 1/N$ for $n = 1, 2, \ldots, N$.
3. **Do For** $m = 1, \ldots, M$

(a) Fit classifier $y_m(x)$ to the training data by minimizing the weighted error function

$$J_m = \sum_{n=1}^{N} w_n^{(m)} I(y_m(x_n) \neq t_n) \quad (1)$$

where, $I(y_m(x_n) \neq t_n)$ is the indicator function and equals 1 when $y_m(x_n) \neq t_n$ and 0 otherwise.

(b) Calculate the training error of y_m:

$$\varepsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} I(y_m(x_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}} \quad (2)$$

(c) Set weight of base classifier y_m:

$$\alpha_m = \ln\left\{\frac{1-\varepsilon_m}{\varepsilon_m}\right\} \quad (3)$$

(d) Update the weight of training data:

$$w_n^{m+1} = w_n^m \exp\{\alpha_m I(y_m(x_n) \neq t_n)\} \quad (4)$$

3. **Output**:

$$Y_M(x) = \text{sign}\left(\sum_{m=1}^{M} \alpha_m y_m(x)\right) \quad (5)$$

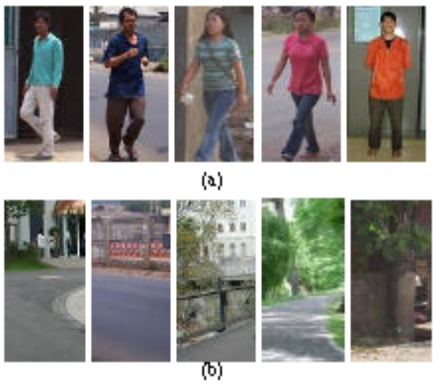


Fig. 2: The training data of human object detection problem. (a) positive samples and (b) negative samples

simulated using Boosting algorithm in Matlab (http://people.csail.mit.edu/torralba/shortCourseRLOC/boosting/boosting.html). And the result shows that the Boosting method can classify image correctly about 67%. This result is not significant for human object detection system that requires much higher accuracy.

DISCUSSION ON KERNEL METHOD

The support vector machine (SVM) algorithm uses structural risk minimization to find the hyperplane that optimally separates two classes of objects (Asano, 2004). This is equivalent to minimizing a bound on generalization error. The optimal hyperplane is computed as a decision surface of the form:

$$f(x) = sgn(g(x)) \quad (6)$$

Where,

$$g(x) = \sum_{i=1}^{N_s} y_i \alpha_i K(x, x_i) + b \quad (7)$$

In Eq. 7, *K* is one of many possible kernel functions, $y_i \in \{1,-1\}$ is the class label of the point x_i and $\{x_i\}$ is subset of the training data set. The x_i are called support vectors and are the points from the data set that closest to the separating hyperplane. Finally, the coefficients α_i and b determined by solving a large-scale quadratic programming problem. The kernel function K that is used in the component classifiers is a quadratic programming problem and has the form shown:

$$K(x, x_i) = (x \cdot x_i + 1)^2 \quad (8)$$

$f(x) \in \{1,-1\}$ in Eq. 6 is referred to as the binary class of the data point x which is being classified by the SVM. Values of 1 and -1 refer to the classes of the positive and negative training examples respectively. As Eq. 6 shows, the binary class of a data point is the sign of the raw output g(x) of the SVM classifier. The raw output of an SVM classifier is the distance of a data point from the decision hyperplane.

SVM corresponds to quadratic programming in maximizing the margin. To solve the high dimensional problem, SVM use kernels which allow algorithms to perform low dimensional calculations that are mathematically equivalent to inner products in a high dimensional "virtual" space.

Using quadratic programming to compute optimal hyperplane as decision surface, SVM results a high accuracy of classification. Consider the same human object detection problem as described in this study, SVM can classify object correctly about 81%. But the testing time of this method is much slower than Boosting method. Therefore, SVM can not be applied for real time human object detection in which much faster testing time is needed.

PROPOSED K-BOOSTING: BOOSTING WITH KERNEL BASE CLASSIFIERS

This study discusses the new proposed method for human object detection called K-Booting (or Kernel Boosting). Human object detection is a real time application that requires high accuracy and fast testing time. The proposed K-Boosting employs the quadratic kernel as base classifiers for Boosting, therefore it has high accuracy and fast testing time.

The quadratic kernel of SVM is a quadratic function of dot product of the training data samples, Eq. 8. Having quadratic function, the kernel is able to form a complex non-linier decision function so that SVM has high classification accuracy. To show the high classification accuracy of SVM, consider the performance of SVM to some data benchmarks (i.e. iris12, iris23 and Usps (http://asi.insa-rouen.fr/enseignants/~gloosli/ simpleSVM.html)). The performance results are shown in Table 2. The average accuracy of SVM to the benchmarks is 90.3%. The results show that SVM is appropriate to be applied for real time human object detection in terms of its high classification accuracy.

Otherwise, the Boosting method has fast classification time. The decision function of Boosting method is a linear combination of prediction of its base classifiers, Eq. 5. Having this simple linear combination, this method requires much less classification time. To show the fast classification time of Boosting, consider the performance of Boosting to some data benchmarks (as in the previous experiments with SVM). The performance results are shown in Table 3. The average testing time is 0.015 sec. The results show that Boosting is appropriate to be applied for real time human object detection in terms of its fast classification time.

Base on the above analysis, a new boosting with kernel function as base classifiers (called K-Boosting) is proposed. The kernel function in K-Boosting is used to improve the classification accuracy. Meanwhile the simple linear combination of base classifier function of Boosting makes the classification time of K-Boosting is fast (Table 4). The schematic illustration of the proposed method is represented in Fig. 3.

First, initialize the weight $w_n^{(1)}$ of training data with the same value. Then, do iterate from 1 until M of base classifiers with kernel function. The first iteration, $y_1(x)$, is trained use kernel function and the result is used for compute error in Eq. 2 with weight coefficient $w_n^{(1)}$. In Eq. 4, update weight coefficient w_n^m that increase weight coefficient value for misclassify the training data and decrease weight coefficient value for correct classify the training data. Finally, when desired number of base

Table 2: Testing accuracy of data benchmarks using SVM

Data benchmarks	Size of testing data	Testing time (sec)	Testing accuracy(%)
iris12	50	0.39	94
iris23	75	0.59	97
Usps	450	8.1	80
Average	90.3%		

Table 3: Testing time of data benchmarks using Boosting

Data benchmarks	Size of testing data	Testing accuracy (%)	Testing time (sec)
Iris12	50	77	0.01
Iris23	75	81	0.01
Usps	450	56	0.03
Average	0.015		

Table 4: K-Boosting algorithm

1. **Input**: a set of training data with labels
2. **Initialize**: the weight of training data: $w_n^{(m)} = 1/N$ for n = 1, 2, ..., N.
3. **Do For** m = 1, ..., M

(a) Use quadratic kernel of *SVM* to train classifier $y_m(x)$ on weighted training set

(b) Calculate the training error of y_m:

$$\varepsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} I(y_m(x_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}} \quad (2)$$

(c) Set weight of base classifier y_m:

$$\alpha_m = \ln\left\{\frac{1-\varepsilon_m}{\varepsilon_m}\right\} \quad (3)$$

(d) Update the weight of training data:

$$w_n^{m+1} = w_n^m \exp\{\alpha_m I(y_m(x_n) \neq t_n)\} \quad (4)$$

3. **Output**:

$$Y_M(x) = \text{sign}\left(\sum_{m=1}^{M} \alpha_m y_m(x)\right) \quad (5)$$

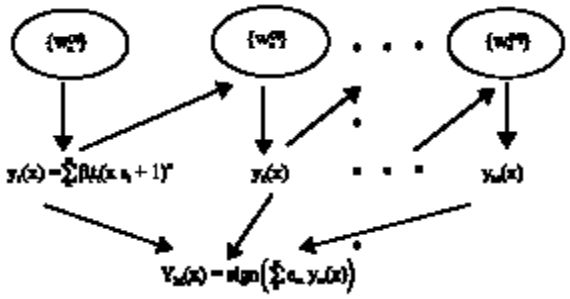


Fig. 3: Schematic illustration of the K-Boosting framework

classifiers has been trained, they are combined to form a committee using coefficients that give different weight to different base classifiers.

HUMAN OBJECT DETECTION SYSTEM

This study explains the architecture of developed human object detection system. The system is divided into training and testing phase. Figure 4 is the graphical representation of the developed system architecture. In training phase, the training data samples those are extracted by Haar wavelet features extraction are used for training the classification method, i.e., Boosting, SVM and the proposed K-Boosting. In testing phase, the system is ready to classify whether the object in the image is "humans" or "non-humans".

Feature extraction: Feature extraction in human object detection is a process to extract information such as color,

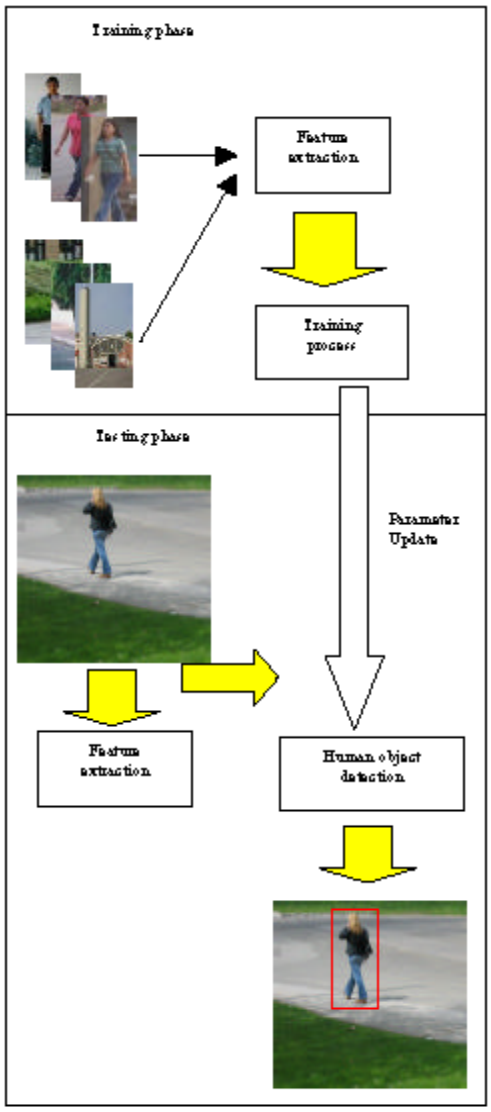


Fig. 4: System architecture of human object detection

texture and shape of human from images. Because human are highly non-rigid objects with a high degree of variability in size, shape, color and texture, in the developed system Wavelet transform is used as feature extraction method. Wavelet transform is a multi resolution function approximation that allows for the hierarchical decomposition of a signal therefore it can extract complex information of human.

Wavelet transform computation involves recursive filtering and sub sampling; and at each level, it decomposes a 2D signal into four sub bands, which are often referred to as LL, LH, HL and HH (L = Low, H = High) according to their frequency characteristics (Unser, 1995). The system uses Haar wavelet transform that it represents the features by the energy in the high frequency bands. The reason for choosing Haar wavelet transform is that it has better reflection of texture properties (Unser, 1995) where the coefficient in different frequency bands signal variations in different directions, such as horizontal, vertical and diagonal.

In addition, Haar transform requires less computation compared to other wavelet transform with longer filters (Unser, 1995). Haar Wavelet of image representation maintains high inter-class and low intra-class variability. Thus, it captures the defining details of the human object in image while distinguishing it from all other objects. Figure 5 represents the 3 types of 2-dimensional Haar wavelet. The 3 types of 2-dimensional Haar wavelet are used to represent features of image. The results that are obtained by applying the Haar wavelets are classified as either "humans" or "non-humans".

To provide more robust features to the classification process, the overcomplete sets of Haar wavelet features (Papageorgiou and Poggio, 2000) is used. This method transform image from pixel space to the space wavelet coefficients, resulting an overcomplete dictionary of features that are then used as training for a classifier. To obtain a denser set of basis functions that provide a richer model and finer spatial resolution, a set of redundant basis functions (called overcomplete dictionary) is needed, where the distance between the wavelets at level n is ¼ 2^n. The illustration of standard transform compared to quadruple dense shift resulting an overcomplete dictionary of wavelets is represented in Fig. 6.

Training and testing phase: In the training phase, the system takes as input a set of images of the human object that have been aligned and scaled so that they are all in approximately the same position and the same size. The training data is resulted from capturing image with 3-5 m distance and cropping image with size 128×64 pixels. To train this system, it uses the training data set of positive

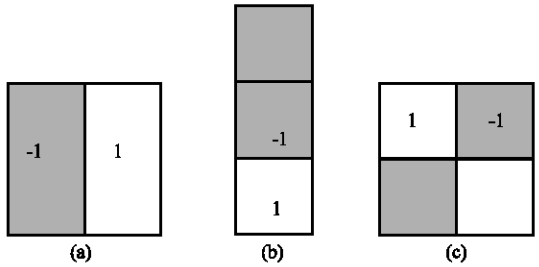


Fig. 5: The 3 types of 2-dimensional Haar Wavelet (a) "vertical" (b) "horizontal" and (c) "diagonal"

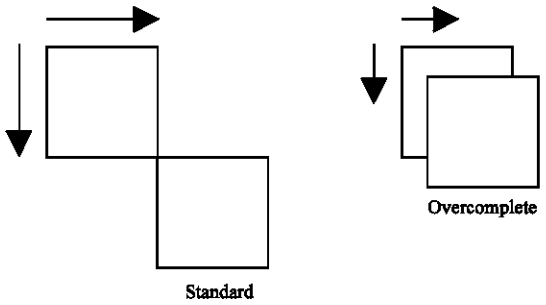


Fig. 6: The shifts in the standard transform and overcomplete dictionary of wavelet

samples and negative samples. The positive samples are images of human with variability in different colors and garment types. The negative samples are images of natural scenery and buildings that are not contained any human.

In testing phase, the system slides a fixed window over an image and uses the trained classifier to decide which patterns show the human objects or not. At each windows position, the system extracts the same set of features as in the training phase. The system starts detecting human object in images by selecting 128×64 windows from top left corner of the image as an input. The window shifts a step by step about 16 pixel from top left corner until bottom right corner. Once the training stage is completed, the system is able to detect a human object at arbitrary positions by shifting the 128×64 window, thereby scanning all possible locations in the image. The results of testing phase are two classes, i.e., positive and negative class. If the result is a positive class then the system draws a red box of location of human object.

EXPERIMENTAL RESULTS AND EVALUATION PERFORMANCE

In the experiment, a dataset of 250 positive samples and 375 negative samples is used. These samples are obtained from capturing image with 3-5 m distance with

variability in different locations. The positive samples are images of human with variability in different colors and garment types. The negative samples are images of natural scenery and buildings that are not contain any human. Therefore the negative samples need many of variation of samples. The positive samples are divided two parts, i.e., 150 samples as training dataset and 100 samples as testing dataset. The negative samples are divided two parts, i.e., 300 samples as training dataset and 75 samples as testing dataset.

Performance of the proposed K-Boosting, Boosting and SVM for human object detection problem are compared. To evaluate the system performance, the 2 scenarios are carried out. The first scenario uses a training dataset with 450 samples and uses a testing dataset with 175 samples that the sizes of images are 128×64 pixels. The first scenario aims to measure the accuracy and computational time of training process and testing process of the proposed method that is compared with Boosting and SVM. The second scenario uses a training dataset with 450 samples and uses a testing dataset with the size of images those are larger than samples of training dataset.

In the 1st scenario, the result of accuracy and computational time of training and testing process is presented in Table 5.

This result shows that the testing accuracy of the propose method is comparable to that SVM method but higher about 16% than Boosting method. The testing time of the propose method is comparable to that of conventional Boosting but higher than SVM method.

In the 2nd scenario, the system starts detecting human object in images by selecting 128×64 windows from top left corner of the images. The system detects area of image of active window. If the result of human object detection are positive class then the system draw a red box. The result of the second experiment is shown in Fig. 7-9.

Figure 7 shows the result of experiment of image with size 180×135 and consists of one object human. The area of image that consist of human object is signed with a red box. The results human object detection of third methods are same.

Figure 8 show the result of image with size 176×234 and consist of five human objects. The area of image that consist of human objects are signed with red boxes. In Fig. 8c, there are three red boxes that are false detection. The false detection is caused area of false detection in image has similarity features with human object. This shows that the proposed K-Boosting has better performance than the conventional Boosting.

Table 5: The accuracy and computational time of training and testing process of K-Boosting, Boosting and SVM

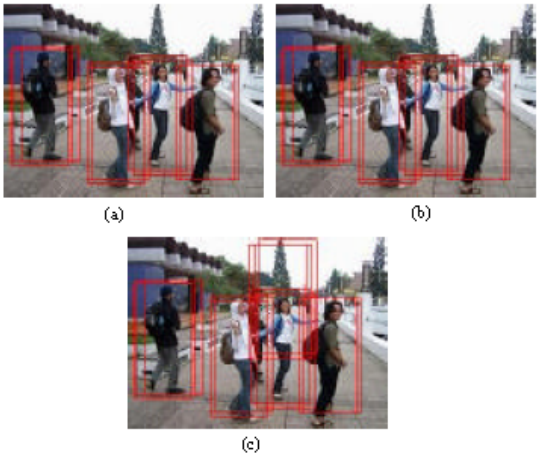
	Accuracy of (percentage)		Computational time	
Methods	Training (%)	Testing (%)	Training	Testing
K-Boosting [proposed]	97	83	161.52	0.01
SVM (http://asi.insa-rouen.fr/enseignants/~gloosli/simpleSVM.htm)	97	82	110.9	10.7
Boosting (http://people.csail.mit.edu/torralba/shortCourseRLOC/boosting/boosting.html)	87	67	11.328	0.01

Table 6: The testing time rate of K-Boosting, Boosting and SVM methods of large image

	The testing time of experiment of Fig. 5-7 with large image		
Methods	Size 180×135	Size 176×234	Size 231×155
K-Boosting [proposed]	4.69	9.63	9.45
SVM (http://asi.insa-rouen.fr/enseignants/~gloosli/simpleSVM.htm)	147.45	172.13	171.80
Boosting (http://people.csail.mit.edu/torralba/shortCourseRLOC/boosting/boosting.html)	4.69	9.63	9.45

(a) (b) (c)

Fig. 7: The result of human object detection of image with size 180×135 using (a) K-Boosting (b) SVM and (c) Boosting



(a) (b)

(c)

Fig. 8: The result of human object detection of image with size 176×234 using (a) K-Boosting (b) SVM and (c) Boosting

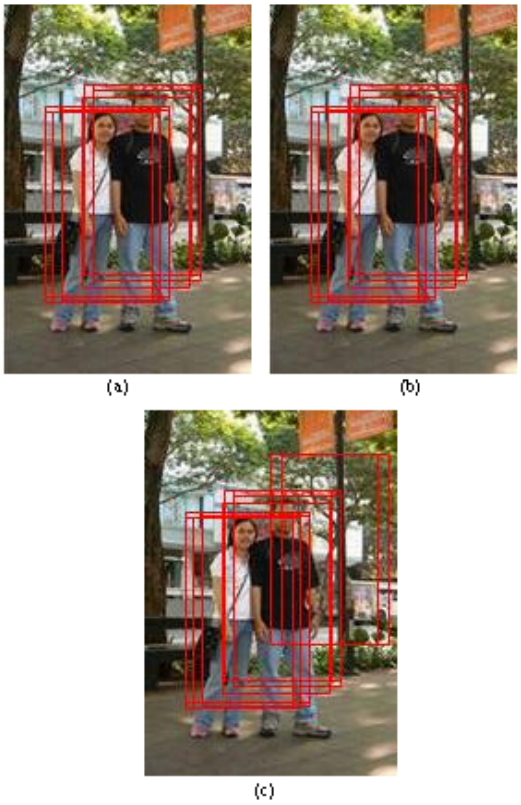


Fig. 9: The result of human object detection of image with size 231×155 using (a) K-Boosting (b) SVM and (c) Boosting

Figure 9 show the result of image with size 231×155 and consist of 2 human objects that the system draw some red boxes. In Fig. 9c, there are two red boxes that are false detection. The false detection is caused area of false detection in image has similarity features with human object. This shows that the proposed K-Boosting has better performance than the conventional Boosting.

The testing time of experiment of Fig. 7-9 are presented in Table 6. The testing time of K-Boosting method is comparable to that Boosting method, but is much lower then SVM method.

CONCLUSION

Human object detection is an important issue to address because of the many applications of human object detection system, i.e., surveillance system, visual search engine and intelligent vehicles. The real application of human object detection required high accuracy and fast testing time. This research proposes K-Boosting method that employs the quadratic kernel as base classifiers for Boosting, therefore it has high accuracy and fast testing time.

In the experiment, the testing accuracy of the proposed K-Boosting method, that it uses testing dataset of 100 positive samples and 75 negative samples, is about 83%. This result improves the accuracy about 16% from that of the conventional Boosting that is about 67%. The accuracy detection of large image of K-boosting is comparable to that of SVM method. In Boosting method, the result of human object detection of large image show that there are some false detection in image test. The false detection is caused area of false detection in image has similarity features with human object. The experiment shows that the proposed K-Boosting has a high accuracy that is comparable to SVM and a fast testing time that is comparable to Boosting, therefore the proposed method can be applied for a real time application.

The research can be extended to the Internet domain to create a visual web search engine. Besides that the proposed method can be further used in state of the art surveillance systems and car driver assistance systems.

REFERENCES

Arras, K.O., O´ scar Mart´ınez Mozos and W. Burgard, 2007. Using Boosted Features for the Detection of People in 2D Range Data. IEEE International Conference on Robotics and Automation.

Asano, A., 2004. Support Vector Machine and Kernel Method. Pattern Information Processing.

Bishop, C.M., 2006. Pattern Recognition and Machine Learning. Springer.

Freud, Y. and R.E. Scaphire, 1999. A Short Introduction To Boosting. J. Japanese Soc. Artif. Intelligence, 14 (5): 771-780.

Laptev, I., 2006. Improvements of Object Detection Using Boosted Histograms. IRISA/INRIA Rennes.

Mohan, A., 1999. Object Detection in Images by Components. MIT AI Memo, 1664 (CBCL Memo 178).

Oren, M., C. Papageorgiou, P. Sinha, E. Osuna and T. Poggio, 1997. Pedestrian Detection Using Wavelet Templates. IEEE Proceeding of Conference on Computer Vision and Pattern Recognition (CVPR).

Papageorgiou and T. Poggio, 2000. A Trainable System for Object Detection. Int. J. Comput. Vision Kluwer Acad. Publishers. Manufactured in The Netherlands, 38 (1): 15-33.

Papageorgiou, C. and T. Poggio, 1999. Trainable Pedestrian Detection. Center for Biological and Computational Learning Artificial Intelligence Laboratory MIT.

Sun, Z., G. Bebis, R. Miller, 2002. Quantized Wavelet Features and Support Vector Machines for On-Road Vehicle Detection. Computer Vision Laboratory, Department of Computer Science, University of Nevada, Reno.

Unser, M., 1995. Texture classification and Segmentation Using Wavelet Frames. IEEE Transaction Image Processing, 4 (11): 1549-1560.

Viola, P., M. Jonesa and D. Snow, 2003. Detecting Pedestrians Using Patterns of Motion and Appearance. Mitsubishi Electric Research Laboratories Cambridge, Massachusetts, USA.