Evaluating Feature Importance for Object Classification in Visual Surveillance

Masamitsu Tsuchiya, Hironobu Fujiyoshi
Dept. of Computer Science, Chubu University
Aichi 487-8501 Japan
{tsuchiya, hf}@vision.cs.chubu.ac.jp

Abstract

Feature-based object classification, which distinguish a moving object to human or vehicle, is important in visual surveillance. In order to improve classification performance, in addition to choosing between the classification (such as SVM, ANN etc), we have to pay attention to which subset of features to employ in the classifier. This paper describes a method to evaluate the relative importance of various features for object type classification. Starting with a given set of features, we apply the AdaBoost method and then we compute a metric which enables us to choose a good subset of the features. We apply our method to the task of distinguishing whether an image blob is a vehicle, a single human, a human group, or a bike, and we determine that shape-based feature, texture-based feature, and motion-based feature are reliable for this classification task. We validate our method by comparing with performance of ANN-based classification.

1 Introduction

Feature-based methods are commonly used for object recognition and type classification in visual surveillance[1]. For robustness, we need to select features that are invariance to various changes caused by environment, scaling, viewpoint, and lighting. In order to improve classification performance, in addition to choosing between the classifier (such as SVM, ANN, and etc.), we have to pay attention to which subset of features to employ in the classifier. Especially, in outdoor surveillance, we have to carefully design good features for input vectors of the classifier.

Previous work in this area has focused on producing descriptors and classification method that are invariant to scaling and viewpoint of the detected objects. Lipton et. al [2] have proposed a binary classification method which uses two feature vectors, e.g., dispersedness and area, to distinguish the image blob detected by the adaptive background detection. The automated video surveillance system called VSAM[1] uses artificial neural network(hereafter referred to as ANN) based classification which enables classification robust to size changes (by using information about the zoom parameter of camera). Since both these features are only shape-based features, the performance is not high.

Texture-based features such as histograms of oriented gradients for human detection have been proposed[3]. They compute high dimensional features based on edges and use SVM (binary classification) to detect human regions. Viola and Jones have proposed a pedestrian detection system that integrates intensity information with motion information[4]. Although the systems are aimed for human detection, these methods can be applied to multi-class classification. However, in these methods, the relative contribution of features is not considered because it is difficult to measure how effective each feature vector is.

Further, a common problem in using a learning based classifier such as ANN is that they act essentially as a black box that performs the assigned task for the user. Additionally, if we have too many training sets, it is hard to make a desired combination of feature sets.

This paper describes a method to evaluate the relative importance of features for object type classification. Starting with a given set of features, we apply the AdaBoost method and then we compute a metric which enables us to choose a good subset of the features. We apply our method to the task of distinguishing whether an image blob is a vehicle (hereafter referred as VH), a single human (SG), a human group (HG), or a bike (BK), and we determine that shape-based feature, texture-based feature, and motion-based feature are reliable for this classification task.

2 Features

We use temporal differencing to detect moving objects and for each detected region, we compute feature vectors for object type classification. To classify objects in a video stream, it is important to use a classification metric which is computationally inexpensive, reasonably effective for small numbers of pixels on object, and invariant to lighting conditions or viewpoint. This study introduces three different kinds of features, viz., shape-based feature, texture-based feature, and motion-based feature.

2.1 Shape-based Feature

(1) Aspect ratio Aspect ratio is determined relative size to measure two extensions of the object. We apply ellipse fitting onto detected regions as shown in Figure 1.

$$Aspect\ ratio = \frac{length\ of\ minor\ axis}{length\ of\ major\ axis}. \qquad (1)$$

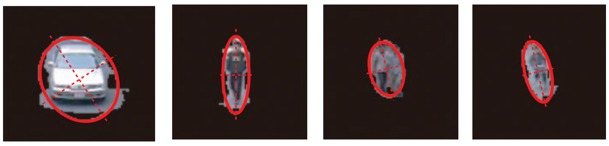

(a) VH (b) SH (c) HG (d) BK

Figure 1. Examples of aspect ratio

(2) Complexity of shape Dispersedness is a measure of the complexity of shape,

$$Dispersedness = \frac{Perimeter^2}{Area}. \qquad (2)$$

A human blob, having a more complex shape, tend to have larger dispersedness than a vehicle blob.

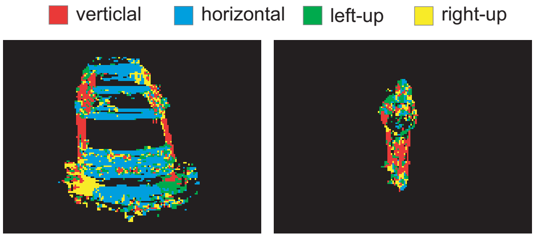
2.2 Texture-based Feature

(3) Edge magnitude and connectivity (8-dimensional vector) The gradient magnitude $g(x, y)$ and the gradient direction $\theta(x, y)$ are computed for each pixel using equation (3).

$$g(x,y) = (I_X^2 + I_Y^2)^{\frac{1}{2}},\ \theta(x,y) = atan(\frac{I_Y}{I_X}) \qquad (3)$$

The simplest implementation of this would be to convolve the 3×3 mask with the image, aligning the mask with the x and y axes to compute the values of I_x and I_y. The gradient direction is divided into four directions such as horizontal, vertical, right-up, and left-up directions as shown in Figure 2. We collect all the pixels which have the same direction (only the four directions) and we compute the normalized sum of their gradient magnitude to get four values.

Connectivity of edge segment for each direction is computed by counting the number of edges which share the same direction and are connected to each other. These four values along with the previous 4 values give us the 8 dimensional texture-based vector.



(a) VH (b) SH

Figure 2. Examples of edge information

2.3 Motion-based Feature

(4) Variance of optical flow vectors A moving vehicle gives rigid motion and a walking human gives non-rigid motion. In order to measure the rigidity of the motion for a moving object, we employ variances of optical flow distribution for 6 blocks. Figure 3 shows examples of optical flow vectors for each category obtained by the KLT method[5]. Given the histogram of the optical flow vectors

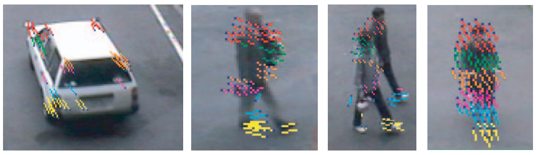
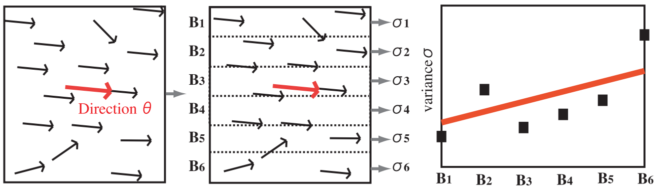

(a) VH (b) SH (c) HG (d) BK

Figure 3. Examples of optical flow vector



(a) Step1: optical flow vector
(b) Step2: variance of each region
(b) Step3: gradient of the variance

Figure 4. Variance of optical flow vector

of a moving object as shown in Figure 4(a), the moving direction θ is selected as median of the histogram. Variances of each region σ_i are computed as shown in Figure 4(b). The least-square method is applied to compute the gradient of the variance by the following equation,

$$G = \frac{n\sum_{i=1}^{n} x_i\sigma_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} \sigma_i}{n\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2} \qquad (4)$$

where $n = 6$ in this study. The gradient value G nearly equals to 1 for rigid motion.

The above mentioned 11 features are normalized to be between 0 to 1.

3 Evaluating the Relative Contribution of Features

We use the AdaBoost method to construct a binary classifier for each category, and then we compute a metric which evaluates importance of the feature.

3.1 Dataset

We collected 200 learning sample images for each category (VH, SH, HG, BK) from a video database of 23 hours. So, total of 800 images is used for training the AdaBoost classifier. A human operator collects sample image and assign class labels to them. Another 800 images are used in the discriminant experiments described in section 4.

3.2 Training Process and Discrimination

In this paper, the training process of AdaBoost is used to obtain a metric for evaluating the contribution of a feature, not used as a classifier.

In training process, Adaboost selects a subset of features in order to construct a robust classifier from training dataset $\{(\boldsymbol{x}_i, y_i) : 1 < i < n\}$ where $\boldsymbol{x} = (x^1, \cdots x^P)$ is 11-dimensional feature vector and $y \in \{-1, +1\}$ is class label. For example, y is set by the following equation in the case of training the VH Classifier.

$$y_i \quad = \quad \begin{cases} +1, if\ \boldsymbol{x}_i \in VH \\ -1, otherwise \end{cases} \tag{5}$$

In this implementation, n is 800 (200 positive examples plus 600 negative examples).

In each round, the learning algorithm chooses from the 11 features described in section 2. The AdaBoost algorithm picks the optimal threshold th for each feature p by the following equation.

$$h_t^{p,th}(\boldsymbol{x}) =_{(0\ \leq\ th\ \leq\ 1\ ,1\ \leq\ p\ \leq\ P)}^{argmin} \{\sum_i I(y_i = \text{sgn}\ (x_i^p - th))\} \tag{6}$$

The output of the Adaboost after the learning process is a binary classifier that consists of a linear combination of the selected features with weighs α_t. Therefore, the final classifer H_T is given by the following equation,

$$H_T(\boldsymbol{x}) = \sum_{t=1}^{T} \alpha_t h_{(t)}(\boldsymbol{x}). \tag{7}$$

The AdaBoost algorithm is given below.

We trained four classifiers one for each class (VH, SH, HG, BK) using the method described above.

```
Algorithm The Adaboost algorithm
1. Input: $n$, Training dataset $(\mathbf{x}_i, y_i)$
2. Initialize: $w_1(i) = 1/n(i = 1 \cdots n),\ h_0(\mathbf{x}) = 0$
3. Do for $t = 1, \ldots, T.\ \ \epsilon_t(h) = \sum_i^n I(y_i \neq h(\boldsymbol{x}_i))w_t(i)$
     (a) $\epsilon_t(h_{(t)}) = min \epsilon_t(h)$
     (b) $\alpha_t = \frac{1}{2} ln(\frac{1-\epsilon_t(h_{(t)})}{\epsilon_t(h_{(t)})})$
     (c) $w_{t+1}(i) = w_t(i) \exp(-\alpha_t h_{(t)}(\boldsymbol{x}_i) y_i))$
4. Output: Final hypothesis with weights $\alpha_t$
        $sign(H_t(\mathbf{x})),\ where\ H_T(\boldsymbol{x}) = \sum_{t=1}^{T} \alpha_t h_t(\boldsymbol{x})$
```

In the discriminant process, we have to merge the results of four trained classifiers as a multi-class classification. The final decision is made by searching the maximum value of the outputs obtained by each classifier $H^c()$ for the category c using normalized weights α'_t.

$$Class =_{c\ \in\ \{VH,\ SG,\ HG,\ BK\}}^{argmax} H^c(\boldsymbol{x}) \ = \ \sum_{t=1}^{T} \alpha'_t ht(\boldsymbol{x}) \tag{8}$$

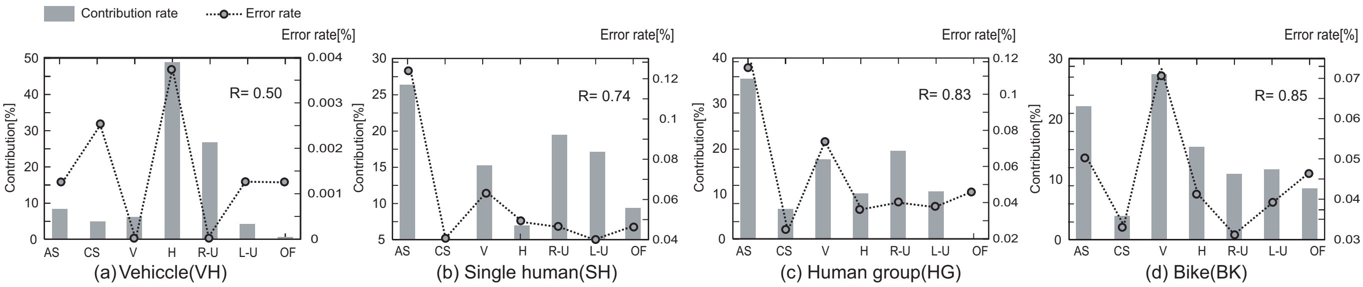
where the normalized weights is obtained by $\alpha'_t = \frac{\alpha_t}{\sum_{i=0}^{T} \alpha_i}$.

3.3 Contribution of Feature

Each round of Adaboost chooses from the total set of the various features, the feature with lowest weighted error on the training examples. The final classifier balances the 11 features in order to maximize classification performance. The weight α, the selected feature, and the thereshold th are chosen at each round are very important factors for classification performance. This introduces a metric which indicates how well the features "contribute" to the classification performance. We define a contribution ratio C_p for each feature p by the following equation,

$$C_p = \sum_{t=1}^{T} \alpha'_t \cdot \delta_K[P(h_t) - p] \tag{9}$$

where p is a kind of feature, and $P()$ is a function to output the feature chosen at round t in the AdaBoost training process, let δ_K be Kronecker delta. This contribution ratio C_p becomes a metric to measure the contribution of the feature vector p. This enables us to determine which subset of the features should be chosen in a given classification task.



(a) Vehiccle(VH) (b) Single human(SH) (c) Human group(HG) (d) Bike(BK)

AS:Aspect Ratio, CS:Complexitiy of Sahpe, V:Vertical edge, H:Horizontal edge, R-U:Right-up edge, L-U:Left-up edgeV, OF:ariance of Optical Flow

Figure 5. Contribution ratio and error ratio

4 Experiment

4.1 Contribution Ratio

Figure 5 shows the contribution rate C_p of feature vector p and the error rate in the discriminant experiments using ANN when it is trained without the feature p. We tested ANN-based classification with 200 sample images for each class, which are not contained in the training sets. Table 1 shows a confusion matrix of the ANN-based classification results. The R in the figure denotes a correlation coefficient between them. This is a measure of how well the contribution ratio “fits” with the classification performance. It is clear that there is a strong correlation between the contribution ratio and the classification performance except vehicle class (VH). This is due to the fact that the “vertical”, “horizontal”, “right-up”, and “left-up” features work in a complementary manner.

Table 1. Classification results by ANN

		Out					
		VH	SH	HG	BK	correct	[%]
In	VH	200	0	0	0	200	100.0
	SH	0	181	11	8	181	90.5
	HG	0	6	189	5	189	94.5
	BK	3	7	0	190	190	95.0
	Sum					760	95.0

4.2 Selected Features

From the Figure 5, we make the following observations about the features for object type classification. For vehicle class(VH), we see that the edge information , especially horizontal component, is effective as a feature for the classification. On the other hand, for other categories, i.e., single human class (SH), human group class (HG), bike class (BK), the shape-based feature such as aspect ratio and complexity of the shape as well as texture-based features are important. For bike class, the motion-based feature such as a variance of optical flow vectors is also important to distinguish a single human class and a bike class. Although the appearances of single human and bike can be very similar from some viewpoints, the motion-based feature distinguishes between the walking human’s non-rigid motion and the bike’s rigid motion.

5 Conclusion

This paper presents a metric, contribution ratio, to evaluate how important a feature is for the task of object type classification. The contribution ratio is obtained from the selected features and weights in the AdaBoost training. The ratio is experimentally validated by demonstrating positive correlation with the performance of a ANN-based method. This helps us to find which feature should be chosen, in a general learning-based classifier.

References

[1] R. Collins, A. Lipton, H. Fujiyoshi, and T. Kanade, “Algorithms for cooperative multisensor surveillance,” Proc. of the IEEE, Vol. 89, No. 10, October, 2001, pp. 1456 - 1477.

[2] A. Lipton, H. Fujiyoshi, and R.S. Patil. “Moving target detection and classification from real-time video.” Proc. of the 1998 Workshop on Applications of Computer Vision, 1998.

[3] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, pp.886-893, 2005

[4] P. Viola, M. J. Jones, D. Snow. “Detecting Pedestrians Using Patterns of Motion and Appearance,” Proc. of the Ninth IEEE International Conference on Computer Vision (ICCV’03) - Volume 2, 2003.

[5] B. D. Lucas and T. Kanade., “An Iterative Image Registration Technique with an Application to Stereo Vision.” International Joint Conference on Artificial Intelligence, pp. 674-679, 1981.