International Journal of Soft Computing 3 (2): 128-133, 2008
ISSN: 1816-9503

Invariant Moments to Scene Categorization Using Support Vector Machines

[1]V. Devendran, [2]Amitabh Wahi and [3]Hemalatha Thiagarajan
[1]Department of Computer Applications,
[2]Department of Information Technology, Bannari Amman Institute of Technology,
Sathyamangalam, Tamil Nadu, India
[3]Department of Computer Applications, National Institute of Technology, Trichy, Tamil Nadu, India

Abstract: Thousands of images are generated every day, which implies the necessity to classify, organize and access them using an easy, faster and efficient way. Scene classification, the classification of images into semantic categories (e.g., coast, mountains, highways and streets) is a challenging and important problem nowadays. Many different approaches concerning scene classification have been proposed in the last few years. This study presents a different approach using invariant moments and support vector machines to scene classification. Radial basis kernel function with $p_1 = 10$ used for SVM. The results are proving the efficiency of this work with 83% classification rate. This complete study is carried out using real world data set.

Key words: Invariant moments, scene categorization, support vector machine

INTRODUCTION

Understanding the robustness and rapidness of human scene categorization has been a focus of investigation in the cognitive sciences over the last decades (Guerin-Dugue and Oliva, 2000; Chella *et al*., 2000; Autio and Elomaa, 2003). At the same time, progress in the area of image understanding has prompted computer vision researchers to design computational systems that are capable of automatic scene categorization. Classification is one of several primary categories of machine learning problems (Serrano *et al*., 2004; Martin, 2005). This study gives a systematic overview of moment invariants for several combinations of deformations and photometric changes (Mindru *et al*., 2004). This study give very promising results in the classification of indoor-outdoor scene image and manmade-natural classification (Pietikainen *et al*., 2004; Boutell and Luo, 2005; Payne and Singh, 2005). Ian Stefan Martin (2005) presents in his doctoral work, the techniques for robust learning and segmentation in scene understanding. Moment invariants are important shape descriptors in computer vision. There are 2 types of shape descriptors: Contour-based shape descriptors and region-based shape descriptors. Regular moment invariant, one of the most popular and widely used contour-based shape descriptors, is a set derived by Hu (1962). Bicego *et al*. (2006) give a new approach to scene analysis under unsupervised circumstances. Bosch *et al*. (2004) present a scene description and segmentation system capable of recognizing natural objects (e.g., sky, trees, grass) under different outdoor conditions. In this study, a computer vision system recognizing objects in captured images is established using Geometric Moment (GM).

INVARIANT MOMENTS FEATURES

Invariant Moment feature descriptors (Rizon *et al*., 2006) were derived from the theory of algebraic invariants and are used to evaluate seven distributed parameters of an image. This technique is chosen to extract image features since the features generated are Rotation Scale Translation (RST) invariant. In any process, the images are processed to extract features that uniquely represent properties of a given category. Invariant moment was successfully applied in texture classification (Rizon *et al*., 2006). The set of seven invariant moments (ϕ_1-ϕ_7) was first proposed by Hu (1962) for 2D images. Two-dimensional moments of a digitally sampled M×M image that has gray function f(x,y) (x, y = 0,…,M-1) is given as:

$$m_{pq} = \sum_{x=0}^{x=M-1} \sum_{y=0}^{y=M-1} (x)^p \bullet (y)^q f(x,y) \quad (1)$$

Where, p, q = 0, 1, 2, 3.

Corresponding Author: V. Devendran, Department of Computer Applications, Bannari Amman Institute of Technology, Sathyamangalam, Tamil Nadu, India

The moments f(x,y) translated by a position (a, b) are defined as:

$$\mu_{pq} = \sum_x \sum_y (x+a)^p \bullet (y+b)^q f(x,y) \quad (2)$$

Thus the central moments μ_{pq} can be computed from (2) on substituting $a = -\bar{x}$ and $b = -\bar{y}$ where

$$\bar{x} = \frac{m_{10}}{m_{00}} \text{ and } \bar{y} = \frac{m_{01}}{m_{00}} \text{ as}$$

$$\mu_{pq} = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q f(x,y) \quad (3)$$

When a scaling normalization is applied the central moments change as,

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \text{ where } \gamma = \left[\frac{(p+q)}{2}\right] + 1 \quad (4)$$

In particular, Hu (1962) defines 7 values, computed by normalizing central moments through order three, that are invariant to object scale, position and orientation. In terms of the normalized central moments, the seven moments are given:

$$\begin{aligned}
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
\phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
&\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
\phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
&\quad + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\end{aligned} \quad (5)$$

SUPPORT VECTOR MACHINES

Support vector machine is a relatively new pattern classifier introduced by Vapnik (1998). A SVM classifies an input vector into one of two classes, with a decision boundary developed based on the concept of structural risk minimization (of classification error) using the statistical learning theory. The SVM learning algorithm directly seeks a separating hyperplane that is optimal by being a maximal margin classifier with respect to training data. For non-linearly separable data, the SVM uses kernel method to transform the original input space, where the data is non-linearly separable, into a higher dimensional feature space where an optimal linear separating hyperplane is constructed. On the basis of its learning approach, the SVM is believed to have good classification rate for high-dimensional data. Consider the problem of image classification where X is an input vector with n dimensions. The SVM performs the following operation involving a vector $W = (w_1,...,w_n)$ and scalar b:

$$f(X) = \text{sgn}(W \bullet X + b) \quad (1)$$

Positive sign of f(X) may be taken as MIT-street images and negative value of f(X) may be regarded as MIT-highways images. Consider a set of training data with *l* data points from 2 classes. Each data is denoted by (X_i, Y_i), where $i = 1, 2, \ldots, l$, $X_i = (x_{i1}, \ldots, x_{in})$ and $y_i \in \{+1, -1\}$. Note that y_i is a binary value representing the two classes. The task of SVM learning algorithm is to find an optimal hyperplane (defined by W and b) that separates the two classes of data. The hyperplane is defined by the equation:

$$W \bullet X + b = 0 \quad (2)$$

Where, X is the input vector, W is the vector perpendicular to the hyperplane and b is a constant. The graphical representation for a simple case of two-dimensional input (n = 2) is illustrated in Fig. 1. According to this hyperplane, all the training data must satisfy the following constraints:

$$\begin{aligned}
W \bullet X_i + b &\geq +1 \text{ for } \forall_i = +1 \\
W \bullet X_i + b &\geq -1 \text{ for } \forall_i = -1
\end{aligned} \quad (3)$$

which is equivalent to:

$$y_i(W \bullet X_i + b) \geq 1 \ \forall_i = 1, 2, \ldots, l \quad (4)$$

There are many possible hyperplanes that separate the training data into 2 classes. However, the optimal separating hyperplane is the unique one that not only separates the data without error, but also maximizes the margin, i.e., maximizes the distance between the closest vectors in both classes to the hyperplane (Burges, 1998). As shown in Fig. 1, the margin, ρ, is the sum of the absolute distance between the hyperplane and the closest data points in each class. It is given by:

$$\rho = \min \frac{|W.X_i + b|}{\|W\|} + \min \frac{|W.X_i + b|}{\|W\|} = \frac{2}{\|W\|}$$

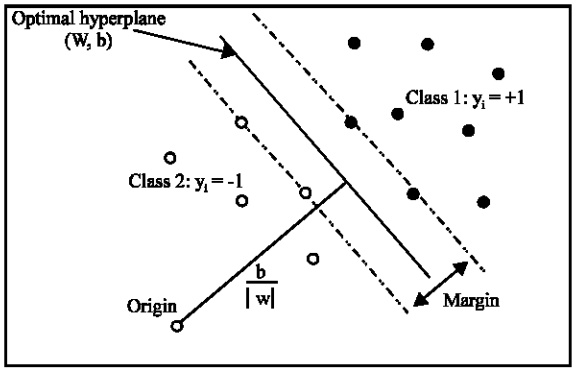


Fig. 1: Optimal separating hyperplane for 2-Dimensional two-class problem

Here, the first min is over X_i of one class and the second min is over X_i of the other class. Therefore, the optimal separating hyperplane is the one that maximizes 2/||W||, subject to constraints Eq. 4. It is mathematically more convenient to replace maximization of 2/||W|| with the equivalent minimization of $||W||^2/2$ subject to constraints Eq. 4, which can be solved by the Lagrangian formulation:

$$\min\ L = \frac{1}{2}\| W \|^2 - \sum_{i=1}^{l} \alpha_i \left[y_i (W.X_i + b) - 1 \right] \qquad (6)$$

Where, α_i is the Lagrange multiplier ($\alpha_i >= 0$, i = 1, 2,.., *l*). The Lagrangian has to be minimized with respect to W and b and maximized with respect to α_i. The minimum of the Lagrangian with respect to W and b is given by:

$$\frac{\partial L}{\partial W} = 0 \Rightarrow W = \sum_{i=1}^{l} \alpha_i X_i y_i \qquad (7)$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{l} \alpha_i y_i = 0 \qquad (8)$$

Substituting Eq. 7 and 8 into Eq. 6, the primal minimization problem is transformed into its dual optimization problem of maximizing the dual Lagrangian L_D with respect to α_i:

$$\max L_D = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j (X_i \bullet X_j) \qquad (9)$$

subject to

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \qquad (10)$$

$$\alpha_i \geq 0 \ \ \forall\, i = 1, \ldots, l \qquad (11)$$

Thus, the optimal separating hyperplane is constructed by solving the above quadric programming problem defined by Eq. 9-11. In this solution, those points have non-zero Lagrangian multipliers ($\alpha_i > 0$) are termed support vectors. Support vectors satisfy the equality in the constraint Eq. 4 and lie closest to the decision boundary (they are circles in Fig. 1, lying on the dotted lines on either side of the separating hyperplane). Consequently, the optimal hyperplane is only determined by the support vectors in the training data. Based on the α_i values obtained, W can be calculated from Eq. 7. b can be obtained by using the Karush-Kuhn-Tucker (KKT) complementary condition for the primal Lagrangian optimization problem:

$$\alpha_i \left[y_i (W.X_i + b) - 1 \right] = 0 \ \ \forall_i = 1,..,l \qquad (12)$$

One b value may be obtained for every support vector (with $\alpha_i > 0$). Burges (1998) recommends that the average value of b be used in the classification. With this solution, the SVM classifier becomes

$$f(X) = \text{sgn}(W \bullet X + b) = \text{sgn}\Big(\sum_{\forall i, \alpha_i > 0} y_i \alpha_i (X_i \bullet X) + b \Big) \qquad (13)$$

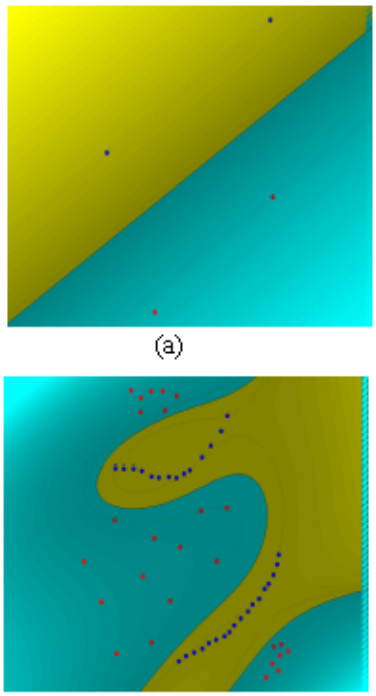
Note that, in Eq. 13, one only needs to make use of X_i, y_i and α_i of the support vectors, while X is the input vector to be classified. When a linear boundary is inappropriate (i.e., no hyperplane exists to separate the two classes of data), the extension of above method to a more complex decision boundary is accomplished by mapping the input vectors $X \in R^n$ into a higher dimensional feature space H through a non-linear function ϕ: $R^n \rightarrow H$. In H, an optimal separating hyperplane is then constructed using training data in the form of dot products $\phi(X_i) \bullet \phi(X_j)$ instead of the $X_i \bullet X_j$ term in Eq. 9. To avoid the expensive computations of $\phi(X_i) \bullet \phi(X_j)$ in the feature space, it is simpler to employ a kernel function such that

$$K(X_i.X_j) = \phi(X_i) \bullet \phi(X_j) \qquad (14)$$

Thus, only the kernel function is used in the training algorithm and one does not need to know the explicit form of ϕ. The computation in (15) results in some restrictions on the form and parameter values of non-linear functions that can be used as the kernel functions. Detailed discussions can be found in Vapkin (1998) and Burges (1998). Some commonly used kernel functions are:

Polynomial function:

$$K(X_i, X_j) = \left(X_i.X_j + 1 \right)^d \qquad (15)$$



(a)

(b)

Fig. 2: Representation of (a) linearly separable (b) non-linearly separable

Radial basis function:

$$K(X_i, X_j) = e^{\frac{\|x_i - x_j\|^2}{2\sigma^2}} \quad (16)$$

Sigmoid function:

$$K(X_i, X_j) = \frac{1}{1 + e^{[\nu(X_i \cdot X_j) - \delta]}} \quad (17)$$

Where, d is a positive integer and σ, ν and δ are real constants. These four parameters must be defined by the user prior to SVM training. With the use of a kernel function, the SVM capable of performing non-linear classification of input X becomes,

$$f(X) = sgn\left(\sum_{\forall_i, \alpha_i > 0} y_i \alpha_i K(X_i, X) + b\right) \quad (18)$$

The hyperplane and support vectors used to separate the linearly separable data are shown in Fig. 2a. And the hyperplane and support vectors used to separate the non-linearly separable data are shown in Fig. 2b. Radial basis kernel function with p1 = 10 used for this non-linear classification. Individual colors represents particular each class of data.

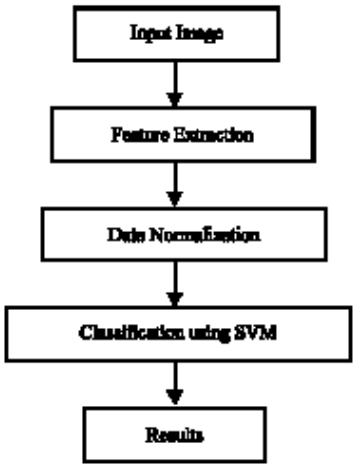


Fig. 3: Detailed description of proposed work

PROPOSED WORK

In classification, a system is trained to recognize a type of example or differentiate between examples that fall in separate categories. In the case of computer vision, the examples are representations of photographic images and the task of the classifier is to indicate whether or not a specific object or phenomena of interest is present in the image. In order to successfully accomplish this, the classifier must have sufficient prior knowledge about the appearance of the object. This study is trying to recognize the scenes of two different categories called MIT-street and MIT-highways. The detailed description of our proposed work is shown in Fig. 3.

The sample images of scenes are taken from the Ponce Research Group (www-cvr.ai.uiuc.edu/ponce_grp/data) which contains 15 different scene categories with 250 samples each. Invariant moment is used for extracting the features from the images/scenes. Normalization is then applied using Zero-mean normalization method in order to maintain the data within the specified range and to improve the performance of the classifier. Support Vector Machine is trained to recognize the scene categories.

IMPLEMENTATION

Radial basis kernel function is used in SVM for scene classification with p1 = 10. Kernel function is trained to find the optimal hyperplane to separate two different categories of scenes and maximize the margin between the two classes of data. In Training phase, 200 samples are used including 100 samples from MIT-street and 100 samples from MIT-highways. In testing phase, 200 more samples are used including 100 samples from

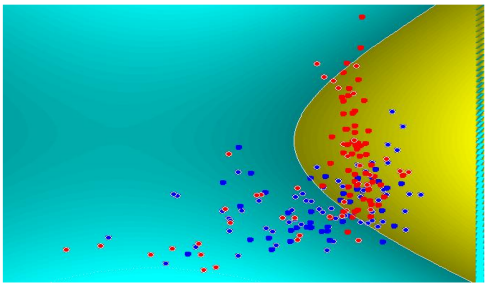


Fig. 4: Optimal Hyperplane that separates MIT-street and MIT-highways scene categories

MIT-street and 100 samples from MIT-highways. The input images are of (256×256) pixel size and it is divided into 4 blocks of sub-images. Invariant Moments are calculated from each single block of sub-image using the equations mentioned in section II. From each single block seven moments are calculated. Thus (4×7 =) 28 values are calculated and used as features for each image in each scene category. Zero-mean normalization method is applied to the extracted features. Normalized features are given as input to Support Vector Machine for training to recognize the scene category. The optimal hyperplane of trained radial basis kernel function that separates two different categories of data is shown in Fig. 4.

DISCUSSION

This study discusses the scene/image classification problem using invariant moments and SVM. Image Data is taken from Ponce Research Group (www-cvr.ai.uiuc.edu/ponce_grp/data). Invariant Moment is applied in all the scene categories by dividing the image into 4 blocks of sub-images without any preprocessing. Support Vector Machine is trained to recognize scene categories called MIT-street and MIT-highways. Radial basis kernel function finds its optimal hyperplane with the following data:

Execution time	=	10.6 sec
Status	=	OPTIMAL_SOLUTION
Hyperplane	=	13009.216875
Margi	=	0.017535
Support Vectors	=	79 (39.5%)

The final classification results are like this: True Positive is 81%; True Negative is 19% and False Positive 85%; False Negative is 15%. Average classification rate is 83.0%. Total time taken including data preparation, training and testing phase is 38.265 sec.

CONCLUSION

This study concentrates on the categorization of images as MIT-street scenes or MIT-highways scenes. Support Vector Machine with Radial basis kernel function are applied together to solve this classification problem. We have achieved 83.0% as an overall classification rate. This research can be further extended to classify other categories (www-cvr.ai.uiuc.edu/ponce_grp/data) of scenes (industrial, kitchen, inside-city, mountain, forest and etc.) with other kernel functions in SVM. This complete work is implemented using SVM Toolbox in Matlab 6.5.

REFERENCES

Autio, I. and T. Elomaa, 2003. Flexible view recognition for indoor navigation based on Gabor filter and support vector machines. Pattern Recognition, 36: 2769-2779.

Bosch, A., X. Munoz and J. Freixenet, 2007. Segmentation and description of natural outdoor scenes. Image and Vision Computing, 25: 727-740.

Bosch, A., X. Munoz and R. Marti, 2007. Which is the best way to organize/classify images by content? Image and Vision Computing, 25: 778-791.

Boutell, M. and J. Luo, 2005. Beyond pixels: Exploiting camera metadata for photo classification. Pattern Recognition, 38: 935-946.

Burges, C.J.C., 1998. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, http://svm.first.gmd.de., 2: 121-167.

Chella, A., M. Frixione and S. Gaglio, 2000. Understanding dynamic scenes. Artificial Intelligence, 123: 89-132.

Florica Mindru *et al*., 2004. Moment invariants for recognition under changing viewpoint and illumination. Computer Vision and Image Understanding, 94: 3-27.

Guerin-Dugue, A. and A. Oliva, 2000. Classification of scene photographs from local orientations features. Pattern Recognition Lett., 21: 1135-1140.

Ian Stefan Martin, 2005. Robust Learning and Segmentation for scene Understanding. Ph.D. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.

Mohamed Rizon *et al*., 2006. Object Detection using Geometric Invariant Moment. Am. J. Applied Sci., 6: 1876-1878.

Navid Serrano, Andreas E. Savakis and Jiebo Luo, 2004. Improved scene classification using efficient low-level features and semantic cues. Pattern Recognition, 37: 1773-1784.

Payne, A. and S. Singh, 2005. Indoor vs. outdoor scene classification in digital photographs. Pattern Recognition, 38: 1533-1545.

Pietikainen, M., T. Nurmela, T. Maenpaa and M. Turtinen, 2004. View-based recognition of real-world textures. Pattern Recognition, 37: 313-323.

Vapnik, V.N., 1998. The support vector method of function estimation. In: Generalization in Neural Network and Machine Learning. Springer-Verlag, New York, pp: 239-268.