Image Database Classification and Retrieval Using Support Vector Machines (SVM)

Wut Yee Thant, Thiri Thitsar Khaing
Computer University (Myitkyina), Myanmar
wutyeethant@gmail.com, thirithitsarkhaing@gmail.com

Abstract

In this paper, we propose a support vector machines (SVMs) method of classifying image regions hierarchically based on their semantics rather than on low-level features. First, image regions are segmented using the hill-climbing method. And then, the support vector machines classify these regions. The SVMs learn the semantics of specified classes from a database of image regions. A support vector machine was used as the classifier. We developed a new way to assign probability after multi-class SVM classification. Our approach achieved approximately 90% accuracy on a collection of images with minimal noise. A support vector machine (SVM) is used to classify the feature vectors. To reduce the computation time and improve the classification accuracy. We also developed a new way to compute probabilistic outputs from a multi-class support vector machines.

1. Introduction

We propose to extend the approaches of the automatic hierarchical classification of image regions into a more detailed classification hierarchy based on the semantics of the region's content. Therefore, content-based image classification and retrieval systems has been the subject of many research works in recent years. It is hard to classify images based on low-level features by non-expert users due to the semantic gap that exists between user needs and system requirements.

The gap between the low-level descriptions of image content and the semantic needs of users can be reduced if image search systems adopt a semantics based approach to organize and index the image content. For example, the color distribution in terms of color histogram can be simply extracted for any image [4]. It classifies the landscape images into sunset, forest, and mountain classes.

In section 2, the rest of this paper is presents a simple and fast nonparametric segmentation method based on hill-climbing algorithm to determine the clusters of an image. In section 3, the extracted features from the image regions. Then the support vector machines tool is explained in Section 4. In section 5, we discuss how the image regions are semantically classified. The proposed system design is discussed and results are expressed in section 6. Finally, conclusion and future work concerning this research are described in section 7.

2. Hill-Climbing Method for Image Segmentation

In this case, the system will perform image region segmentation by using Hill-Climbing method. This algorithm detects the peaks of clusters in the global three-dimensional color histogram of an image. Then, the algorithm is outlined as follows:

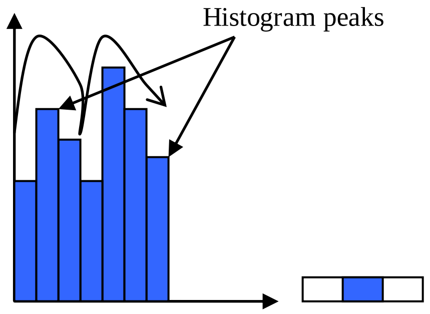
1. Compute the global 3-dimensional RGB color histogram of the image.
2. Start at a non-zero bin of the color histogram and make uphill moves until reaching a peak (local maximum) as follows:
    - 2.1. Compare the number of pixels of the current histogram bin with the number of pixels of neighboring bins. Note that the feature space is 3-dimensional, thus each bin in the color histogram has $3^d-1=26$ neighbors, where d is the dimensionality of the feature space $(d=3)$.
    - 2.2. If the neighboring bins have different numbers of pixels. The algorithm makes an uphill move towards the neighboring bins with larger number of pixels.
    - 2.3. If the immediate neighboring bins have the same number of pixels, the algorithm checks the next neighboring bins, and so on, until a set of neighboring bins with different number of pixels are found.

Then, an uphill move is made towards the bin with larger number of pixels.

2.4. The hill climbing is continued (repeat steps 2.1-2.3) until reaching a bin from where there is no possible uphill movement. That is the case when the neighbori9ng bins have smaller numbers of pixels than the current bin. Hence, the current bin is identified as a peak.

3. Select another unclimbed bin as a starting bin and perform step 2 to find another peak. This step is continued until all non-zero bins of the color histogram are climbed (associated with a peak).
4. The identified peaks represent the initial number of clusters of the input image; therefore, these peaks are saved.
5. Finally, neighboring pixels that lead to the same peak are grouped together. This associates every pixel with one of the identified peaks. Consequently, the segments of the input image are formed [4].

The image is segmented using a hill-climbing algorithm in the three-dimensional histogram of the image can be seen as a search window to find the largest bin within that window. Figure 1 explains the algorithm for a one-dimensional case. In three dimensional, each bin in the color histogram has 3^d-1=26 neighbors where d is the number of dimensional of the feature space[8].



(a) (b)

Figure 1. (a) Finding Peaks in a Histogram using Hill-Climbing Method. (b) For a One Dimensional Histogram.

The following figures are the segmented image from the original input nature image using hill-climbing method.

Figure 2. The Semantic of Nature Input Image.

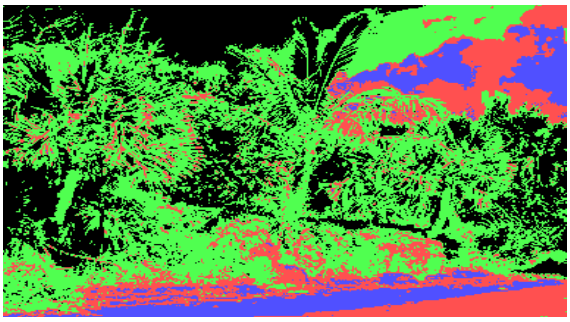


Figure 3. The Segmented Image from Nature Image

3. Features Extrection

These features were extracted from the image regions as we discuss in following subsections.

3.1 Color Histogram

Color provides and extremely powerful for the distinguishing of different entities in the scene. As color features, we use the mean RGB values over a block[1]. Color histograms are widely used in image retrieval. Color histogram gives an estimation of the distribution of the colors in the image. The color space is partitioned and for each partition the pixels within its range are counted, resulting in a representation of the relative frequencies of the occurring colors [2].

3.2 Edge Direction Histogram

Edge direction is a fundamental tool used in most image processing applications to obtain information from the frames as a precursor step to feature extraction and object segmentation. The basic edge-detection operator is a matrix area gradient operation that determines the level of variance between different pixels [3].

The edge-detection operator is calculated by forming a matrix centered on a pixel chosen as the center of the matrix area. The Canny algorithm uses an optimal edge detector based on a set of criteria which include finding the most edges by minimizing the error rate. According to Canny, the optimal filter that meets all three criteria above can be efficiently approximately using the first derivative of a Gaussian function.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{\frac{-x^2+y^2}{2\sigma^2}} \quad (1)$$

$$\frac{\partial G(x,y)}{\partial x} \alpha x e^{-\frac{x^2+y^2}{2\partial^2}} \; \frac{\partial G(x,y)}{\partial y} \alpha^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2)$$

Results can be further improved by performing edge detection at multiple resolutions using multi-scale representations [3].

3.3 Higher Order Autocorrelation Edge Vector

The higher order autocorrelation features are the primitive edge features that are shift-invariant (irrelevant to where the objects are located in the image), which is a useful property in image querying. Before extracting these features an edge image is constructed using the Canny method [4].

4. Support Vector Machines (SVM)

A support vector machines (SVM) performs classification by constructing an N-dimensional hyperplane that optimally separates the data into two categories.

A support vector machines is one of the famous supervised machines learning algorithms for binary classification in all various dataset and it gives the best results where the data set is a separable and especially when the training data set is a few, and with extended algorithms it can be used in multi-class problems [5].

In binary classification, SVMs try to find a hyperplane to separate the data into two classes. In this case in which all the data are well separated, the margin is defined as two lines the distance between the hyperplane with the largest margin, which provides good generalization ability. To increase the classification ability, SVMs first map the data into a higher dimension feature space with Ǿ(x), then use a hyperplane that features space to sperate the data[9].

The SVMs is the compact representation of the decision boundary. Only the training points that lie on, or between, hyperplanes H_1 and H_2 are considered as the support vectors and give non-zero real values. The other points are ignored.

A kernel function acts as a similarity function between points. A simple form of similarity metric is the dot product between two vectors. In a space in which the positive members of a class form more than one cluster as in image databases, a Gaussian classifier is known to be more accurate.

$$K(x,y) = \exp\left[\frac{-\|x-y\|^2}{2\sigma^2}\right]$$

Where σ is the width of the Gaussian. Thus in this paper, we use a Gaussian kernel, which is a technique used by most radial basis function classifiers [4].

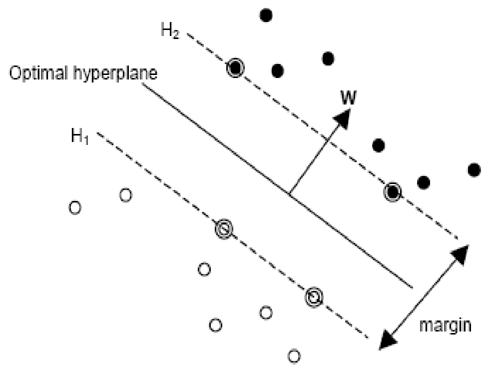


Figure 4. A Linear Hyperplane Separating the Members of Two Classes.

Kernel method work by embedding data items into a vector space *F*, called a feature space and searching for linear relations in such a space. This embedding is defined implicitly, by specifying an inner product for the feature space via a positive semi-definite kernel function:

$$K(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle$$

Where (x_1) and (x_2) are the embeddings of data items x_1 and x_2 [6]. Steps for constructing a SVM are non linear mapping of input vectors into a high dimensional feature space that is hidden from both the input and output and construction of an optimal hyperplane for separating for separating the features.

5. Image Regions are Semantically Classified

To classify image regions into semantic classes, which humans understand, we first manually defined a hierarchy that reflects the semantics in the Nature images as shown in figure 1.

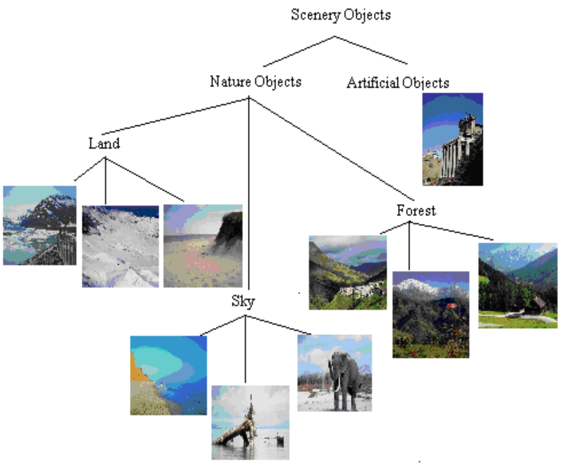


Figure 5. A Class Hierarchy that Reflects the Semantics of Nature Regions.

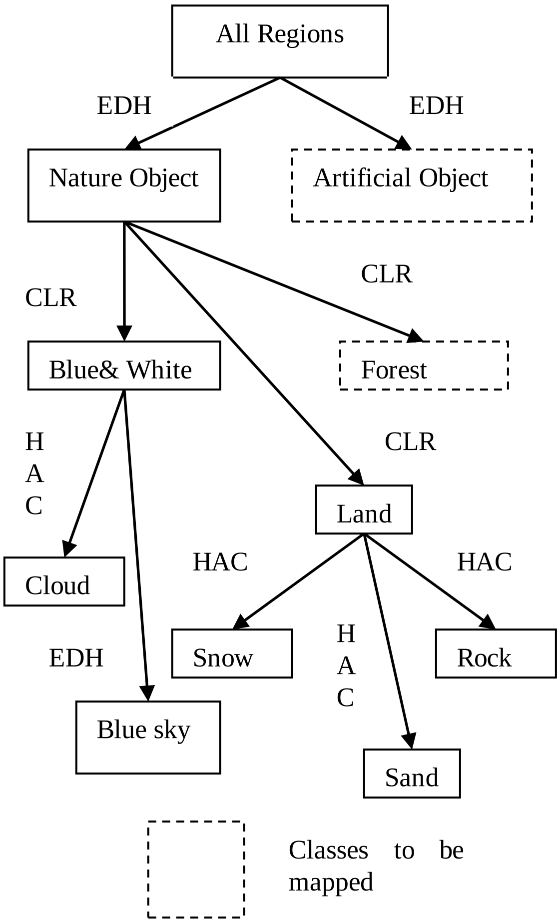


Figure 6. A Class Organization Used by the SVMs to Distinguish the Members of a Class from Others. Where CLR= color, EDH= edge direction histogram and HAC= higher-order autocorrelation edge feature.

6. The System Design and Results

This system will load images from an image database; we are able to represent an image by the classes, in which the image regions are classified. The classes' names is represented each image [4]. The choice of classes name allow for image classification. By using Hill-Climbing method, the system will perform image region segmentation. The image will be classified using SVM classifier and display the result image three categories: Sky, Land and Forest. The system flow diagram is as shown in figure 3.

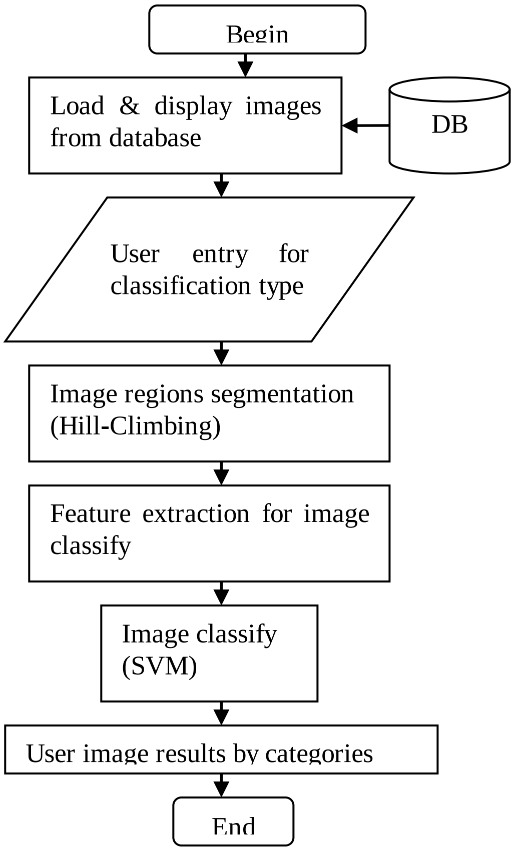


Figure 7. Flow Chart of the System Design.

7. Experiments

The system using NetBeans 5.5 on a PC running Windows XP with Pentium 1 GHz CPU and 512MB RAM. The 50 images used in the experiments were selected from the image collection of Stanford

Table1. Classification of Nature Regions

	Nature
EDH	90.5%
HAC	90.1%

Then, the best classification results were obtained using the color feature due to the nature of class that reflects the color similarity and dissimilarity between the classes at this level. The high precision values (90.7% - 95.7%) of classifications are given in Table 2.

Table 2. Classification After Grouping Classes as Shown in Figure 2 using Color

Class Name	Precision	Subclasses
Blue & White	93.3%	Blue sky, Cloud, Snow
Sand& Rock	90.7%	Sand, Rock
Forest	95.37%	Green Forest

Table 3.Classification of Low-level Classes using Different Features

Class	Feature	Precision
BlueSky	EDH	85.3%
Snow	HAC	87.8%
Sand	HAC	89.5%
Rock	HAC	88.6%
Cloud	HAC	81.2%

These low –level classes are describing by using the features extraction. The subclasses of Blue&White class were classified using the following features: EDH for BlueSky, HAC for Cloud. Similarly, each of the Land and Forest classes were divided into two subclasses each. The HAC feature leads to the best classification precision for both Sand and Rock subclasses. On the other hand the EDH feature leads to the best classification precision for the BlueSky and Cloud.

8. Conclusion and Future Work

This paper presented an approach to classify image regions. Flexible and intuitive query compositions can be based on each semantic hierarchy. The use of the binary support vector machines classifiers to classify image regions using different features at different levels. A final goal is to generate semantic indices into image database.

The weak point of this system is that it cannot classify the ocean. Future work will lead to classify multiple nature objects and using multiple extracted features.

9. References

[1] H.Permuter, J.Francos "Gaussian Mixture Models of Texture And Color For Image Database Retrieval" Ben-Gaurion University, Electrical and Computer Engineering Department, Israel.

[2] Thomas Deselaers, Daniel Keyesers, and Hermann Ney " Features for Image Retrieval: A Quantitative Comparison" Lehrstuhl fur Informatik VI – Computer Sciences Department, RWTH Aachen University- D-52056 Aachen, Germany.

[3] Hong Shan Neoh, Altera Corporation "Adaptive Edge Detection for Real-Time Video Processing using FPGAs" Asher Hazanchuk, Altera Corporation. 101 Innovation Dr. San Jose, CA 95134 , (408) 544 70000.

[4] Zaher AGHBARI " Semantic Approach to Image Database Classification and Retrieval " Graduate School of Information Science and E.E., Kyushu University, Akifumi MAKINOUCHI , Gradute School of Information Science and E.E., Kyushu University.

[5] Alireza Mansouri, Lilly Suriani Affendey, Ali Manat " Named Entity Recognition Using a New Fuzzy Support Vector Machine" Faculty of Computer Science & Information Technology, University Putra Malaysia, 43400 Serdong, Malaysia.

[6] G.R.G. LANCKRIET, Division of Electrical Engineering, University of California, Berkeley "Kernel-Based Data Fusion and Its Application To Protein Function Prediction In Yeast".

[7] C.J.C.Burges, “A Tutorial On Support Vector Machines for Pattern Recognition”, Data Mining and Knowledge Discovery.

[8] Radhakrishna Achanta, Francisco Estrade, Patricia Wils, and Sabine Susrtrunk .”Salient Region Detection and Segmentation”, School of Computer Communication Sciences (I & C). Ecole Polytechnique Federale de Lausanne (EPFL).

[9] Tong Luo, Kurt Kramer, Dmitry Goldgof, Lawrence O.”Learning to Recognize Plankton”. Hall Dept. of Computer Science & Engineering University of South Florida Tampa, FL 33620.