IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Unsupervised Quality Based Image Segmentation

R.Navaneethakrishnan[*1], S.Pradeep[2]

[*1]Department of MCA, Bharathiyar College of Engineering and Technology, Karaikal, India

[2]Dept of Electronic and Communication Engineering Bharathiyar College of Engineering and Technology, Karaikal, India

msgtokrishnan@gmail.com

Abstract

This paper presents an efficient technique for unsupervised segmentation of textured images that aims at incorporating the advantages of supervision for discriminating texture patterns. First, a pattern discovery stage that relies on a clustering algorithm is utilized for determining the texture patterns of a given image based on the outcome of a multichannel Gabor filter bank. Then, a supervised pixel-based classifier trained with the feature vectors associated with those patterns is used to classify every image pixel into one of the sought texture classes, thus yielding the final segmentation. Multi-sized evaluation windows following a top-down approach are utilized during pixel classification in order to improve accuracy both inside and near boundaries of regions of homogeneous texture.. The proposed technique is compared with alternative segmentation approaches.

Keywords: Unsupervised quality segmentation, Supervised pixel-based quality classification, Multi-sized evaluation windows, Gabor filters, Support Vector Machine

Introduction

Image segmentation consists of partitioning given image into disjoint regions of uniform features. It is a complex task, since it requires the detection of those features within each region, as well as the location of the boundaries that separate the different regions. Segmentation is usually a preliminary stage of further processing and analysis tasks, such as classification and interpretation.

In order to segment an image, it is necessary to extract features and derive measures from them that enable segregation of the distinctive regions contained in the image. One of the most important features is texture, which is a major visual cue utilized in a wide variety of applications, since it allows distinguishing among different objects or surfaces with similar shape or color. Unfortunately, it is the visual cue most difficult to model, being intrinsically noisy by nature and affected by various external factors, such as illumination, rotation and scale, which alter its perception. This complexity has fostered a large amount of research during the last.

Texture segmentation can be supervised or unsupervised, depending on whether prior knowledge about the image or its texture classes is available or not. Supervised texture segmentation identifies and separates regions that match texture properties previously learned in training samples. In turn, unsupervised texture segmentation has to discriminate the texture classes of the image as well as separate them into regions. Despite the success of many of the unsupervised texture segmentation algorithms proposed in the literature, their supervised counterparts usually perform better in terms of segmentation accuracy as demonstrated in previous works [2]. The reason is that, by definition, unsupervised texture segmentation algorithms do not take any advantage of any prior knowledge concerning the texture patterns to be discriminated, since this kind of information is not available. On the contrary, supervised algorithms are specifically trained to identify a set of patterns. Hence, they are more likely to succeed, especially when those patterns are difficult to be separated.

Although there have been some attempts to incorporate a supervised classifier into an unsupervised algorithm, those previous approaches only classify a small number of pixels that correspond either to the boundaries between regions or to regions of low-confidence obtained after applying an unsupervised algorithm. However, if that algorithm fails in the remaining pixels, which is highly likely due to its unsupervised nature, they cannot be corrected by the supervised classifier.

As a solution, this paper proposes a two-stage unsupervised texture segmentation technique that applies a supervised classifier in order to completely classify a given input image by taking into account the texture patterns initially discovered by an unsupervised algorithm. Since the only objective of the pattern discovery stage is to find out a set of suitable patterns and not to accurately define the image regions, this process is carried out by using a small number of feature samples, thus significantly reducing the computational cost of the whole process. In the second stage, a supervised classifier is trained with samples of the previously obtained texture patterns and then applied in order to perform pixel- based classification of the complete input image, yielding the final segmentation. Another key point of the proposed technique is that multi-sized evaluation windows following a top-down approach are used during classification in order to improve the accuracy inside regions of homogeneous texture, as well as near boundaries. The texture features for both stages are obtained by means of a Gabor filter bank. Experiments show that the proposed technique is effective in terms of both segmentation quality and computation time.

Related Work

There is a vast amount of literature on texture segmentation and it is beyond the scope of this paper to review it. We shall concentrate instead on the methods that are most relevant to our approach. The basic idea of our method is to use an unsupervised segmentation to learn the local classes and then proceed to use a supervised segmentation to obtain the final result. Similar ideas have been proposed by other researchers and here we review these works.

Some methods are concerned with the refinement of the boundary of the segmentation. For example, Ojala and Pietikainen in [4] performed pixel wise classification of boundary pixels after segmentation with a split and merge algorithm based on local binary pattern (LBP) histograms. They utilized a discrete disk of radius equal to 11 pixels and the G statistic as a dissimilarity measure, and treated the LBP histograms of the image segments as texture models. Camilleri and Petrou in [5] refined boundaries by extending the ideas of linear spectral unmixing, as applied to remote sensing, to the case where the local energy of a boundary pixel is a linear combination of the local energies of the pixels on either side of the boundary, with surrounding windows that do not touch the boundary.

Although in terms of functionality the proposed technique resembles the approaches in [4, 5], it is conceptually different. While the latter apply a supervised classifier in order to refine the boundaries of a complete segmentation produced by a previous unsupervised stage, the methodology proposed in this paper aims at classifying completely the input image, using the patterns obtained by an approximate unsupervised segmentation of the same image. In other words, while the main task in previous approaches is performed through unsupervised clustering, the main task in the presented approach is carried out by supervised classification.

Hence, clustering is only a means for obtaining training patterns and it may be replaced by any suitable, alternative method. Other methods use the learned classes from the unsupervised classifier as seeds for growing finer regions. In [6], Mirmehdi and Petrou blurred the image according to the blurring filters that model the way humans perceive colors from different distances, and clustered the highly blurred image (coarsest level) to determine the so called „„core clusters"", i.e., sets of pixels that can confidently be associated with the same region, as their affinity persists for very large scales of smoothing, that eliminate variation due to texture. Then, those clusters were used as the basis for classifying pixels at more detailed scales using probabilistic relaxation.

Unsupervised Texture segmentation Scheme

The unsupervised texture segmentation scheme proposed in this work is as follows. During the initial step, a given image is filtered by applying a multichannel Gabor filter bank, thus obtaining a feature vector for every pixel. Then, a reduced number of these feature vectors is selected by uniform sampling of their associated image pixels and passed to a clustering algorithm in order to determine which texture patterns are present in the image. After pattern discovery, the pixels associated with the feature vectors are used to build a labeled image of the same size as the original.

From this labeled image, a number of pixels corresponding to each texture pattern are selected, again by uniform sampling, but avoiding pixels close to boundaries between regions. This contributes to selecting pixels assumed to belong to „„„pure"""" texture patterns. Next, the feature vectors associated with the selected pixels are used to train a supervised classifier based on support vector machines (SVMs). This classifier produces the final segmented image after performing pixel-based classification by considering the complete set of feature vectors.

The classification methodology follows a top-down approach with multi-sized evaluation windows. Since the classes that make up the classified image are spatially updated after each window size is applied, pixels selected as training samples for the next classification level are updated as well. In cases where it is not possible to take enough training samples for a given class, since all of its associated pixels are in a boundary zone, the class is discarded and not taken into account in the remaining classification levels.

Figure 1 shows how the proposed scheme processes a given image.

A. Texture Feature Extraction

Texture feature extraction is performed by means of a multichannel Gabor filter bank. In particular, the well-known design strategy proposed in [8] is followed. After filtering an input image, the texture features that will characterize every pixel are the mean and standard deviation of the module of the resulting coefficients evaluated over that pixel and its surrounding neighborhood (evaluation window). Hence, every pixel will have an associated vector with a total of M = 2 x S xK dimensions, where S and K are the number of scales and

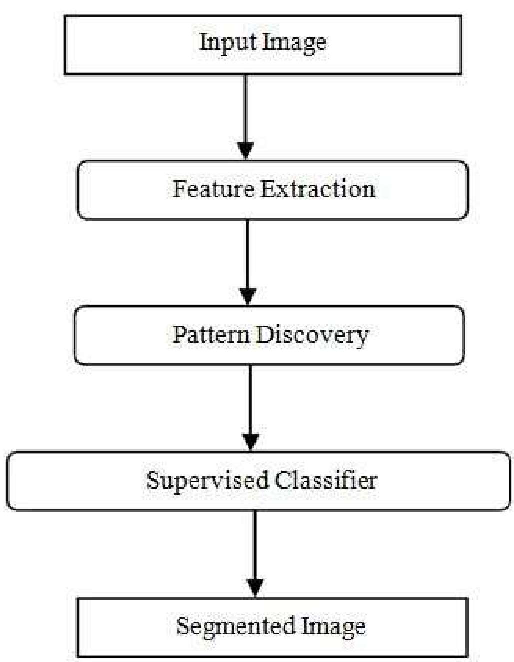

Figure 1. Steps performed by the proposed scheme

orientations used to configure the filter bank, respectively. All dimensions have been normalized in the [0,1] interval, thus avoiding any bias. In order to take advantage of the output produced by the filter bank, the texture features mentioned above have been computed for W different evaluation window sizes as suggested in [9,10].

In this way, a multilevel characterization of each of the texture patterns is obtained during the pixel-based classification stage, which improves the accuracy of the segmentation both inside and near boundaries of regions of homogenous texture.

B. Pattern Discovery

In this stage, a subset of the previously obtained feature vectors is passed to a clustering algorithm in order to discover the texture classes present in the processed image. Since clustering is a usual approach related to many applications in different domains, there are several alternatives regarding the clustering algorithm to use. In this work, three baseline algorithms that are thought to be representative of the existing clustering approaches have been considered for the core of the pattern discovery stage: k-means, mean shift clustering and graph clustering based on the normalized cut.

C. Supervised Pixel-Based Classification

At this stage, the set of texture patterns found by the previous stage are used as texture models for a supervised pixel-based classifier, thus effectively transforming the original unsupervised problem into a supervised one. As its name suggests, a pixel-based classifier aims at determining the class to which every pixel of an input image belongs, which leads to the segmentation of the image as a collateral effect. In order to achieve this objective, several measures are computed for each image pixel by applying a number of texture feature extraction methods.

D. Classification With Multiple Evaluation Window Sizes

Although previous works on supervised pixel-based classification have already shown the benefits of utilizing multiple evaluation window sizes, which approach is the best for combining these different sources of information is still an open issue. For instance, in [9], different window sizes were integrated by assigning a weight to their corresponding probabilities according to how well each window size separates a given training pattern from the others. However, since the training patterns are single textured images, the assigned weight is not representative of the structure of the test image, which in turn is composed of multiple texture patterns. Furthermore, this method may be biased to the largest window, as it captures more information and, hence, has better capabilities of distinguishing between texture classes.

Later, in [10], improved classification rates were obtained by directly fusing the outcome of multiple evaluation window sizes using the KNN rule. The main problem with this approach is that it does not guarantee that the most appropriate window size will always receive the majority of votes. Ideally, the strategy for classifying a test image using multiple evaluation window sizes should apply large windows inside regions of homogeneous texture in order to avoid noisy classified pixels and small windows near the boundaries between those regions in order to define them precisely. Unfortunately, that kind of knowledge about the structure of the image is only available after it has been segmented.

A priori approximation of that strategy can be devised through the following steps:

Step 1: Select the largest available evaluation window and classify the test image pixels labelled as unknown (initially, all pixels are labelled as unknown).

Step 2: In the classified image, locate the pixels that belong to boundaries between regions of different texture and mark them as unknown, as well as their neighbourhoods. The size of the neighbourhood corresponds to the size of the window used to classify the image.

Step 3: Discard the current evaluation window.

Step 4: Repeat steps 1–3 until the smallest evaluation window has been utilized.

This scheme, which can be thought of as a top-down approach, has been used during the supervised classification stage of the proposed segmentation technique. In addition to closely approximating the previously described ideal strategy for using multiple evaluation window sizes, this approach avoids the classification of every image pixel with all the available windows. Hence, it leads to a lower computation time than previous approaches.

E. Support Vector Machine-Based Classifier

Pixel classification in the previous scheme is performed by means of a support vector machine-based classifier. SVMs have been selected due to their excellent discriminating capabilities and low computation time in both training and testing. Since an SVM is a binary classifier, an extension is needed in order to solve multiclass problems. Preliminary experimentation, as well as comparative results ,suggest that one-against-one SVMs, which separate every one class from each other, are the best alternative both in terms of classification accuracy and computation time, as only "small", two-class problems need to be solved, which, in addition, yields a reduced number of support vectors.

Under that extension, a binary classifier is required for each pair of classes. Hence, the total number of SVMs given a problem with T classes is T (Tx1)/2. The final classification considering every pair of classes j and k is obtained by the following rule: if according to the sign of the decision function, point x belongs to the jth class, then the jth class receives one more vote; otherwise, the votes for the kth class are increased by one. Finally, x is assigned to the class with the largest number of votes.

Since only the sign of the decision function is considered, the above strategy is discrete, i.e., each binary classifier directly chooses the best alternative and the aggregated score only accounts for that alternative, completely disregarding the other. Clearly, this may lead to wrong decisions in cases where both alternatives are almost equally likely. Furthermore, two or more classes may have identical votes. Therefore, an additional untying mechanism would be necessary. In order to alleviate these problems, probability estimates have been included instead of discrete votes. Thus, for the final classification, probabilities for each class are added and x is assigned to the class with the highest probability.

An independent set of experiments has shown that this strategy yields better classification results than its discrete counterpart.

Experimentation

Experiments have been carried out in order to validate the performance of the proposed unsupervised segmentation technique. The following sections describe the experimental setup and show and discuss the obtained results.

A. Test Images

A broad collection of images of 256 x 256 pixels has been considered in this work. Some images may have several possible ground-truths, as the goodness of the segmentation depends on the desired level of detail, which in turn depends on the context of the application for which the segmentation technique is required. This is especially true for some real scenes in which the number of different textured regions is not clear.

B. Segmentation Quality Measure

In order to measure the quality of the segmentation maps produced by the evaluated approaches, a segmentation quality factor has been defined. It is an improvement over the one proposed in [2] and is inspired by the classification rate that measures the performance of supervised pixel-based classifiers, which is simply the ratio between the number of correctly classified pixels and the number of valid pixels in the ground-truth. However, in the unsupervised case, it is not possible to determine which pixels are correctly classified due to the lack of correspondence between the labels of the segmentation map and the labels of the ground-truth.

Hence, in order to adapt the above performance measure, a region based quality factor instead of a pixel-based one has been utilized.
The basic idea consists of comparing the segmentation map with the corresponding ground-truth and, for every region in the latter, determining the region with the largest overlap in the segmentation map. The ratio between the area of the overlapping portion and the area of the corresponding ground-truth region is an indicator of how good the segmentation of that particular region is. Afterwards, a global score is obtained by aggregating the partial scores:

$$Q \quad \sum_{r=1}^{R} \frac{A_r^g \cap A_r^s}{A_r^g}, \tag{1}$$

A_r^s Where A_r^g is the area of one of the R regions in the ground-truth, is are of its corresponding region $A_r^g \cap A_r^s$ in the evaluated segmentation map and is the area of the portion between both regions. The area of a region is defined as the number of its pixels. Associations between pairs of regions are required to be unique. In cases where it is not possible to associate a region in the ground-truth with any of the regions in the segmentation map due to under segmentation or over

Segmentation, the partial score for that region is zero.
Figure 2 and table1 shows the segmentation results.

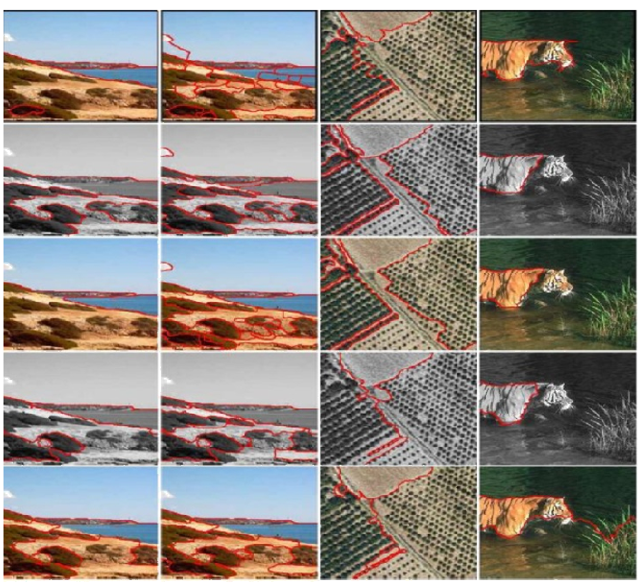

Figure 2. Segmentation results from the experiments. Results by: CTM (color) (first row), OWT-UCM (second row), OWT-UCM (color) (third row), Proposed method (fourth row), Proposed method (color) (last row).

Segmentation Technique	Segmentation Quality	CPU Time(s)
CTM (color)	78.4	8.7
OWT-UCM	82.1	7.8
OWT-UCM (color)	80.3	6.7
Proposed method (fourth row)	90.4	4.5
Proposed method (color) (last row)	90.3	5.2

Conclusions And Future Work

This paper presents a new unsupervised texture segmentation technique based on a supervised pixel-based classifier that achieves good segmentation quality with low computational time.
A pattern discovery stage is applied in order to identify the texture patterns of a given image by means of a clustering algorithm, thus effectively transforming the initial unsupervised problem into a supervised one, which allows the proposed technique to benefit from the advantages of a supervised classifier.
Accuracy inside and near boundaries of regions of homogeneous texture is improved by utilizing multiple evaluation window sizes according to a top-down approach that also contributes to speeding up the segmentation process, as only a reduced number of image pixels needs to be classified by all the available window sizes. The computational cost is further reduced by considering a subset of feature vectors as input for the pattern discovery stage, as only a rough approximation of the texture patterns is necessary.
The proposed technique has been compared with the basic clustering algorithms applied during the pattern discovery stage and with alternative segmentation techniques. Results in terms of segmentation quality and computation time have always been favorable.
Future work will consist of investigating different multi-sized window schemes in order to better utilize these information sources when classifying a given feature vector and thus, improving both the accuracy and the computation time of the pixel-based classification stage of the current algorithm. Another way to improve this time is to reduce the number of support vectors evaluated during classification.

References

[1] M. Petrou, P. Garcia Sevilla, Image Processing: Dealing with Texture, John Wiley, 2006.

[2] M.A. Garcia, D. Puig, Pixel classification by divergence-based integration of multiple texture methods and its application to fabric defect detection, Lect. Notes Computer Science. pp.132–139,2003.

[3] T. Randen, J.H. Husoy, “Filtering for texture classification: a comparative study”, IEEE Trans. Pattern Analysis, Vol. 21, pp. 291–310, 1999.

[4] T. Ojala, M. Pietikainen, “Unsupervised texture segmentation using feature distributions”, Pattern Recognition,Vol. 32 , pp. 477–486,1999.

[5] K.P. Camilleri, M. Petrou, “Spectral unmixing of mixed pixels for texture boundary refinement”, Proceedings of the ICPR, 2000, pp. 1096–1099.

[6] M. Mirmehdi, M. Petrou, “Segmentation of color textures”, IEEE Trans. Pattern Analysis, Vol. 22, 142–159, 2000.

[7] J. Luo, A.E. Savakis, “Two-stage texture segmentation using complementary features”, Proceedings of the IEEE ICIP, 2000, pp. 564–567.

[8] B.S. Manjunath, W. Y. Ma, “Texture features for browsing and image retrieval”, IEEE Trans. Pattern Anal.,Vol.18 pp. 837–842, 1996.

[9] D. Puig, M.A. Garcia, “Automatic texture feature selection for image pixel classification, Pattern Recognition., Vol. 39, pp. 1996–2009, 2006.

[10] J. Melendez, M.A. Garcia, D. Puig, “Efficient distance-based per-pixel texture classification with Gabor wavelet filters”, Pattern Anal. Appl. 11, pp. 365–372, 2008