ORIGINAL RESEARCH PAPER
Volume-8 | Issue-10 | October - 2019 | PRINT ISSN No. 2277 - 8179 | DOI : 10.36106/ijsr
INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH

DECISION TREES BASED IMAGE SEGMENTATION USING ENSEMBLE CLUSTERING

Engineering

Lavi Yaron Department of Information Systems, The Max Stern Yezreel Valley Academic College

Abdallah Loai* Department of Information Systems, The Max Stern Yezreel Valley Academic College *Corresponding Author

ABSTRACT

In this study, we propose an improvement on a commonly used classification algorithm – Decision Tree. The mentioned improvement has been observed during segmentation of colored images. The method in which we improved the performance of a regular decision tree, involved clustering the data which was then used as the training set for the decision tree. The clustering algorithm used was K-Means. After running both regular and improved decision tree algorithms several times, on different images of animals in a diverse set of backgrounds, we concluded that the decision tree algorithm that was empowered by the clustering of K-Means, had higher confidence levels and performance.

KEYWORDS

decision trees, k-means, computer vision, ensemble clustering

INTRODUCTION

Computer vision is by no means a new field. In fact, it goes back as far as the late 1960's when the first "boom" of artificial intelligence research had occurred. Our study focused on image segmentation in general using the Decision Trees classification algorithm. In the image processing field, an image is treated as a collection of units, each called a pixel, each color of an image consists of three layers: red, green and blue, where each layer Contains a matrix of pixels that describes the intensity of a certain color, and the classification of a decision tree does not know how to deal with such data. But what if we organized the data in such a way, that helps the decision tree predict which pixel belongs to a certain group, and which doesn't? in the following sections, we explain how we did just that, and what conclusions can be drawn from this research.

Image segmentation can help us cope with various challenges such as detection of cancer cells, identification of diseased / infected plants in field crops, change focus between background and main object, detect burglars in security footage, robot-vision and so forth.

In this study we suggest an innovative approach to segment an object in colored images. We then compare its performance against commonly used algorithms. The top performer was indeed the new and improved decision tree that we trained, both in its depth, complexity and ultimately in its ability to predict the labels it was trained upon. The core of our approach is combining the strengths of a decision tree's classification, together with that of k-means ensemble clustering.

This paper is organized as follows: **Related work** on Image segmentation and decision trees is discussed in next Section, followed by a Section which elaborates on **Our method**. The Section named **Ensemble Clustering based Decision Trees** describes the ensemble clustering method using the ***K*-means clustering algorithm**. The Results of **Experiments on real colored images** are presented afterwards. And Finally, our **Conclusions** are presented right before the **References**.

Related works

Shotton et al. (2008) used semantic Texton Forests to segment objects in an image, claiming that the method they used is extremely fast to both train and test. Especially compared with k-means clustering and nearest neighbor assignment of feature descriptors.

Gupta et al. (2014) used semantically rich image and depth features to cope with the problem of object detection for RGB-D images. They proposed a new geocentric embedding for depth images that encodes height above ground and angle with gravity for each pixel in addition to the horizontal disparity.

In our study, we use a different technique to segment objects in an image. We propose an improvement to the traditional decision tree by using an ensemble clustering algorithm(k-means) on pre-digested data of a picture (raw pixels).

Permuter et al. (2005) used gaussian mixture models (GMMs) of colored texture on several feature spaces, in contrast to our usage of decision trees. Although, they did, quite similarly to our case, compare the performance of these models with other popular models in the literature.

Viola et al. (2004) used cascade detection processes, which is essentially that of a degenerate decision tree. As such, it resembles our work with decision trees, but differs in approaches: theirs has a mechanism that discards parts of the image which it deems irrelevant, while ours retains the information of it all.

Heumann et al. (2014) had a similar approach to our study, in which they combined two machine learning algorithms. In their case: A decision tree (DT), together with support vector machine (SVM), in ours: A decision tree (DT), improved by K-Means (KMC).

Schroff et al. (2008) used pixel-wise segmentation of images like our study, but with the use of random forests instead of decision trees. They showed that combining several features can improve the performance of the algorithms, and we showed that combining algorithms can improve performance.

Im et al. (2008) stated that per-pixel analysis may not function successfully in satellite imagery and used contextual information to overcome the problem. Their approach is like our study in a way, where we use K-Means as our contextual information classifier.

Our method

The main goal of our research was to differentiate(segment) a given image into two distinct units – A Main Object (like a rabbit, see Figure 1) in that image, and its corresponding Background (grass, trees etc.).

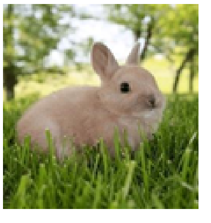

Figure 1: Rabbit

Running Decision Trees for image segmentation

In the image processing field, an image is treated as a collection of units, each called a pixel. Each colored image consists of three layers, represented by matrices. As a first step in the study presented, we converted the mentioned three matrices of a picture into one matrix, in which, every column stands for a different property of a pixel: a row number, marked with X, a column number, marked by Y, a red level (or intensity) - a number between 0 and 255, a number for green and a number for blue. Finally, we kept another detail indicating the type of each pixel. If it represented an object - 1, and if it represented a background - 0.

Thanks to the above conversion, we were able to run the decision tree algorithm on the pixels of the images that we "fed" it with, which gave us good results.

The decision tree was built using the image's data which was structured as the matrix described in the previous section, and its training set which was represented as a vector of the corresponding pixel's class (labeled 0 for background and 1 for object). See Figures 2A, B to witness the improvement of our method in comparison to regular decision trees.

Ensemble Clustering based Decision Trees

We used a method that improves the representation of the proposed data, namely – ensemble clustering, which is a method for grouping similar points in space. We used the k-means clustering algorithm. Since K-means requires you to set a certain number of clusters, and because the exact number of clusters required is unknown, we decided to run the algorithm several times while setting the number of clusters randomly and saving all the results, and by doing so, we could determine how many clusters are needed. Furthermore, the basic assumption was that pixels in the same cluster are pixels with similar traits. Figures 2A, B demonstrate the improvement of our method on regular decision tree classification approach. This improvement came to fruition in deduction of pruning levels from 6(Figure 2B) to 4(Figure 2A).

Figure 2A: improved decision tree's pruning levels

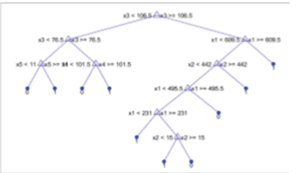

Figure 2B: regular decision tree's pruning levels

Objects which were always clustered together in the same clusters are defined as members of an equivalence class respectably. As a result, the decision tree algorithm now runs on clustered data rather than on raw pixel information as its input. This clustering process is in effect a smart data reduction technique, which can help many different algorithms in computer vision specifically, and in machine learning in general. We observed that above a certain number of clusters, our algorithm would continue running indefinitely, since it works until an iteration with no moving centroids (convergence).

The clustering results are the inputs for the Decision Tree algorithm. We ran the algorithm on 6 different and difficult images and saw that the proposed algorithm (Decision Trees using the ensemble clustering) which utilized the clustered data, outperforms the basic Decision Trees algorithm in every single image.

While being a rather fast and lightweight performant algorithm, the decision tree's key disadvantage comes in the form of its input data, which it tries to classify. When a decision tree is given with raw input, it would usually have a hard time figuring out how to classify it. As mentioned previously, our approach in tackling this problem, was using the K-Means algorithm to improve the decision tree's prediction capabilities.

K-Means clustering algorithm

K-Means starts by randomly defining k centroids. From there, it works in iterative (repetitive) steps to perform two tasks:

Assign each data point to the closest corresponding centroid, using the standard Euclidean distance. In layman's terms: the straight-line distance between the data point and the centroid.

For each centroid, calculate the mean of the values of all the points belonging to it. The mean value becomes the new value of the centroid. Once step 2 is complete, all the centroids have new values that correspond to the means of all their corresponding points. These new points are put through steps one and two producing yet another set of centroid values. This process is repeated over and over until there is no change in the centroid values, meaning that they have been accurately grouped. Or, the process can be stopped when a previously determined maximum number of steps has been met.

Experiments on real colored images

Figure 3B demonstrates two learning curves of both the regular and the improved decision trees. The orange curve refers to the regular decision tree algorithm, while the blue one refers to our improved decision tree. The X axis represents the number of pixels used in the training set of the corresponding decision tree, while the Y axis represents the confidence level of the classifier.

Table 1. Numerical image properties

Dataset	Image size	K's values for the K means	Cluster matrix size	# of the object points	# of the background points
Rabbit Image	364x435	2-50	158340x26	120459	37881
Bird Image	1799x1200	2-50	2158800x26	1985573	173227
Panda Image	900x610	2-50	549000x26	431050	117950
Platypus Image	816x490	2-50	399840x26	346651	53189
Tiger Image	612x423	2-50	258876x26	197064	61812
Weasel Image	500x333	2-50	166500x26	156405	10095

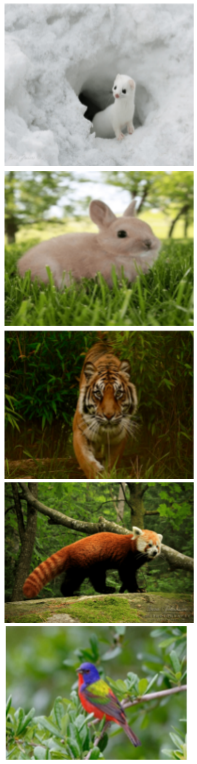

Figure 3A: Animal Image

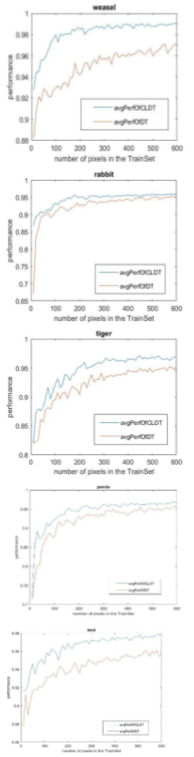

Figure 3B: Learning curves

Figure 3C: Learning curves

CONCLUSIONS

In this study we implemented an innovative approach to segment an object in colored images. Our approach includes the combination of two algorithms, Decision Trees and K-Means clustering.

Comparing the newly combined Decision Tree's performance against commonly used algorithms, yielded better results regarding both its depth, complexity and ultimately its ability to predict the labels it was trained upon. As presented in our research, neither Decision Trees, nor K-Means are flawless. However, combining the strengths of them both, did in-fact yield better performance and higher confidence for prediction of the classes (object or background) of the individual pixels of a colored image. This novel approach is especially good on images with a higher difficulty in segmentation of the desired main object, such as the white weasel on top of a snowy background shown earlier. A proposal for future research would be to include colored images which contain multiple objects, that are spread around the background and are separated from each other.

REFERENCES

[1] AbedAllah, L., & Shimshoni, I. (2012, September). k Nearest neighbours using

ensemble clustering. Proceeding of International Conference on Data Warehousing and Knowledge Discovery. Vienna, Austria, pp. 265-278.
[2] Gupta, S., Girshick, R., Arbeláez, P., & Malik, J. (2014, September). Learning rich features from RGB-D images for object detection and segmentation. In European Conference on Computer Vision (pp. 345-360). Springer, Cham.
[3] Shotton, J., Johnson, M., & Cipolla, R. (2008, June). Semantic texton forests for image categorization and segmentation. In 2008 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.
[4] Permuter, H., Francos, J., & Jermyn, I. (2006). A study of Gaussian mixture models of color and texture features for image classification and segmentation. Pattern Recognition, 39(4), 695-706.
[5] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. CVPR (1), 1(511-518), 3.
[6] Heumann, B. W. (2011). An object-based classification of mangroves using a hybrid decision tree—Support vector machine approach. Remote Sensing, 3(11), 2440-2460.
[7] Schroff, F., Criminisi, A., & Zisserman, A. (2008, September). Object Class Segmentation using Random Forests. In BMVC(pp. 1-10).
[8] Im, J., Jensen, J. R., & Tullis, J. A. (2008). Object-based change detection using correlation image analysis and image segmentation. International Journal of Remote Sensing, 29(2), 399-423.