IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Efficient Image Classification Using Sum Product Tree Based Parallel Computing Approach Using Large Feature Set

B Venkat Raman[1], Vaidya Saikumar[2] and Naresh Ashetti[3]

Lecturer, CSE, RGUKT Basar, Hyderabad, India [1]

Student, CSE, RGUKT Basar, Hyderabad, India [2, 3]

Abstract: To create an efficient classification method using a multi structural analogy for images. Classification is a division of data into groups of similar objects. Representing the data by fewer clusters necessarily loses certain minor details, to minimize the minute error, come up with a theme is to increase and create an efficient classification technique. When we have a large number of data to be processed and it needs to be clustered for simplification. We have to use efficient approach to consume less time should be taken into consideration. This paper represents creating such methodology in keeping mind all the pros and cons occurred in clustering .We have come up with an idea, using multi features like image feature extraction, parallel computing and distance finding algorithms which bind with sum product tree will results in achieving an efficient classification.

Keywords : Parallel computing, K nearest neighbor, canny edge detection, sobel edge detection, sum product tree.

I. INTRODUCTION

Generally if we want to divide large number of images/data into required number of clusters we use image processing techniques and cluster but processing time is main factor which needs to be reduced. Image Processing with parallel computing is an alternative way to solve image processing problems that require large times of processing or handling large amounts of information in acceptable time. This make use of less resources and results in efficient output. In parallel computing method different objects of the image will be processed at the same time this multitasking makes CPU to be utilized up to our need.

Classification can be done by the idea of sum product trees, the results for nodes of tree will be obtained from parallel computing techniques where we use different image feature techniques like shape, color, texture and distance finding algorithm(nearest neighbor) and with set of rules and procedure we cluster images. Because of the parallel computing we can save time and/or money, we can solve larger problems in very short time periods.

II. LITERATURE REVIEW

Classification : Classification the groups (or classes) are specified before hand, with each training data instance belonging to a particular class. It is the association between the instances features and the class they belong to that classification algorithms are supposed to learn.

1 .K Nearest Neighbor algorithm

The K-nearest neighbor algorithm stores all available cases and classifies new cases based on a distance functions. KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970's as a non-parametric technique.

Algorithm

A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor.

Distance functions

Euclidean
$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Manhattan
$$\sum_{i=1}^{k}|x_i - y_i|$$

Minkowski
$$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$$

Hamming Distance

$$D_H = \sum_{i=1}^{k}|x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

X	Y	Distance
Male	Male	0
Male	Female	1

 DOI 10.17148/IJARCCE.2016.53180

It should also be noted that all three distance measures are only valid for continuous variables. In the instance of categorical variables the Hamming distance must be used. It also brings up the issue of standardization of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset.

Choosing the optimal value for K is best done by first inspecting the data. In general, a large K value is more precise as it reduces the overall noise but there is no guarantee. Cross-validation is another way to retrospectively determine a good K value by using an independent dataset to validate the K value. Historically, the optimal K for most datasets has been between 3-10. That produces much better results than 1NN [4].

2. Image Feature Extraction Techniques

Color :

Using color feature we can compare two images to determine difference between them. This can be achieved by using Pixel concept where pixel is the basic part of an image which is compose of three part values Red, Green, Blue. By analyzing RGB values image processing and analyzing can be done. We have Histogram Concept which helps in determining and having work with image processing.

A histogram simply plots the frequency at which each grey level occurs from 0 (black) to 255 (white) (in black and white images). In color images it plots the frequency of red, green and blue levels in the image. In “bad” images, the majority of these frequencies lies in the middle of the histogram (between100 – 175). Hence, the contrast isn’t very good. Histogram equalization fixes these errors.

Histogram concept available in different types.

1. Grey Histogram, 2. Red Histogram, 3. Blue Histogram, 4. Green Histogram, 5. Histogram Oriented Gradient

Shape Detection

Canny edge detection :

Canny’ s intention was to create a perfect edge detector. Unlike Sobel, Canny extracts thin, clear edges. It works by a two step algorithm in which firstly an edge filter just like Sobel is applied 3 and secondly a non-maxima suppression is applied to get thin lines that represent the edges. The second step is essential to obtain thin lines as edges. Starting from a point of greater gradient magnitude than a threshold value T1, the algorithm follows a ridge, that’s to say a path of local maxima perpendicular to the edge Detection. Points that are not along that ridge are suppressed, the path itself is the final edge. The path stops as soon as the gradient magnitude falls below a second threshold T2. The introduction of a second threshold was to avoid “dashed” edges where the edge’s gradient magnitude is close to one threshold, noise would make it pass the threshold frequently. An Issue with the method as explained here, by the way, is that Y-junctions of edges are impossible, as only linear paths are followed. Where three edges meet in a point, two would connect and the third one would stop just before the actual junction because it is suppressed as the first two edges are tracked [3].

Texture Detection

An **image texture** is a set of metrics calculated in image processing designed to quantify the perceived texture of an image. Image texture gives us information about the spatial arrangement of color or intensities in an image or selected region of an image [1].

Image textures can be artificially created or found in natural scenes captured in an image. Image textures are one way that can be used to help in segmentation or classification of images. For more accurate segmentation the most useful features are spatial frequency and an average grey level [2].To analyze an image texture in computer graphics, we use LBP concept here.

The LBP feature vector, in its simplest form, is created in the following manner:

- Divide the examined window into cells (e.g. 16x16 pixels for each cell).
- For each pixel in a cell, compare the pixel to each of its 8 neighbors (on its left-top, left-middle, left-bottom, right-top, etc.). Follow the pixels along a circle, i.e. clockwise or counter-clockwise.
- Where the center pixel's value is greater than the neighbor's value, write "0". Otherwise, write "1". This gives an 8-digit binary number (which is usually converted to decimal for convenience).
- Compute the histogram, over the cell, of the frequency of each "number" occurring (i.e., each combination of which pixels are smaller and which are greater than the center). This histogram can be seen as a 256-dimensionalfeature vector. Optionally normalize the histogram.
- Concatenate (normalized) histograms of all cells. This gives a feature vector for the entire window.

The feature vector can now be processed using themachine-learning algorithm to classify images. Such classifiers can be used for texture analysis.

3. Image Processing With Parallel Computing

Parallel computing provides concurrency and by this we can use non– local recourses very efficiently. It also removes the limit of `serial computing. Image Processing with Parallel computing is an alternative way to solve image processing problems that require large times of processing or handling large amounts of information in "acceptable time" (according to each criterion).

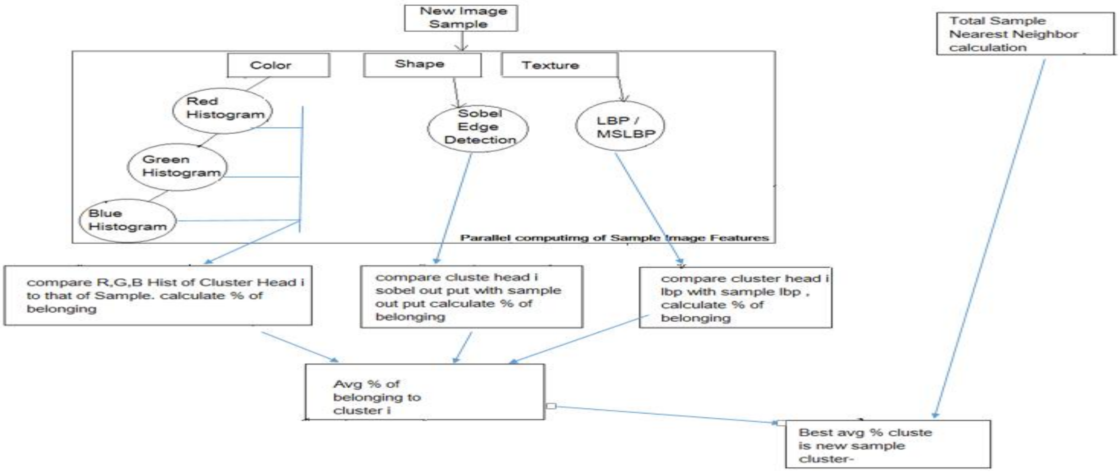
As we can see that high number of data sample imaging requires lots of memory space and time to process so by parallelizing we can find efficient and fast result. In parallel processing, a program is able to create multiple tasks that work together to solve a problem. The main idea of parallel image processing is to divide the problem into simple tasks and solve them concurrently, in such a way the total time can be divided between the total tasks (in the best case). Parallel Image processing cannot be applied to all problems, in other words we can say that not all the problems can be coded in a parallel form. A parallel program should must have some features for a correct and efficient operation; otherwise, it is possible that runtime or operation does not have the expected performance [1].

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

III. PROPOSED METHOD

3.1 Architecture of the System



3.2 Methodology

Hardware Requirements

System : Intel® core™ i3-4005U CPU @1.70 GHz.
Hard Disk : 500GB
Cache Memory : 3 Mb.
Ram : 4GB

Software Used

Mpj-v0_44 Express software
Java jdk_1.7.0

3.3 Implementation Procedure

This Classification takes input a large input image set and performs following procedure on it and cluster those images. Image set is randomly taken around 1000 images.

- Initially cluster heads will be defined for each cluster and their properties will be recorder and saved for further usage.
- When a sample image comes from images set to detect into which cluster it belongs,
- Through parallel computing multitasking at a time its image features like color, shape, texture will be acquired.
- Using sum product tree structure defined as in architecture these properties will be compared using image comparison technique to that of each cluster head properties and percentage of matching will be recorded,
- Finally for the cluster which has percentage of belonging is high sample image will go into that cluster.

VI. CONCLUSION

Our proposed method is a simple mechanism for high efficient clustering . It works effectively and is a combination of various existing methods proposed for the process. However, to reduce the complexity in identification and verification we used simple base algorithms in each related field and parallel computing also done in one system with mpj-express software which is specialist in parallel computing tasks.

VII. LIMITATIONS

1. To reduce the complexity in identification and verification we used simple base algorithms in each related field and
2. Parallel computing also done in one system with mpj-express software which is specialist in parallel computing tasks.

REFERENCES

[1] Linda g. Shapiro and George c. stockman, computer visio, Upper saddle River:Prentice-hall,2001
[2] Trambitskiy k.v.,Anding K.,Polte G.A., Garten D., Musalimov V.M (2015)." OUT-OF-FOCUS REGION SEGMENTATION OF 2D SURFACES IMAGES WITH THE USE OF TEXTURE FEATURES"
[3] André Aichert, CAMP MEDICAL SEMINAR WS0708, "Feature extraction techniques", January 9, 2008
[4] http://www.saedsayad.com/ K_Nearest_Neighbors.html

BIOGRAPHIES

Venkat Raman B, is working as teaching faculty in the department of Computer Science and Engineering, Rajiv gandhi University of Knowledge Technologies-Basar, India. His passions include teaching, seminars and conducting workshops. His research interest is in the area of data mining and machine learning.

Naresh Ashetti is an undergraduate student at Rajiv gandhi University of Knowledge Technologies-Basar, India. His passions include developing innovative software. His research interest is in the area of data mining and machine learning.

Vaidya Saikumar is an undergraduate student at Rajiv gandhi University of Knowledge Technologies-Basar, India. His passions include developing innovative software. His research interest is in the area of data mining and machine learning.