COMPARITIVE ANALYSIS OF FLLT AND JSEG

Ms AKILA VICTOR

Asst. PROFESSOR, SCSE
VIT UNIVERSITY, VELLORE
TAMILNADU, INDIA.

ABSTRACT

The main key point of preferential image segmentation is to segment object of user interest based on intensities, boundaries and texture and ignoring the remaining portions. It explains the trees of shapes to represent image content. In the tree of shape we use algorithms called as FLLT (Fast Level Line Transform) and JSEG (Jsegmentation). Here both the algorithm are compared and an analysis is done to display the result of objects selected from prior images.

Key Words: FLLT, JSegmentation, Texture Classification, Peer Group Filter.

I. INTRODUCTION

Segmentation plays a major role in digital image processing. Image segmentation algorithms are designed to segment an image into several regions so that the contents of each region represent meaningful objects. The segmentation results can therefore be utilized for post processing stages such as object recognition. Image segmentation simplifies post processing stages by focusing attention on each individual segment. Edge detection methods such as the Canny detector [1] were widely applied for this task. Edge detection methods utilize intensity gradients to detect the boundaries of objects.

However, edge detection methods usually generate edges that are not closed contours, and this causes difficulties for later processing such as object recognition. Curve evolution methods [2][6] have been popular for image segmentation since the early 1990s.

These methods evolve the initialized curve(s) to the boundaries of objects in an image. The above methods tend to segment the whole image into several regions, which is challenging for images with cluttered background. On the other hand, this is not always necessary in real applications. The user may be interested in finding only the location of objects of interest. For example consider a car moving on a road then the user can preferentially segment the car image by extracting the background. This is the key idea of “preferential image segmentation,” which means to preferentially segment objects of interests from an image and ignore the remaining portions of the image for this application.

The idea of preferential image segmentation bears some similarities to the object detection methods from images [3]. These methods detect the existence and rough location of objects in an image, e.g., in using a sparse, part-based object representation and a learning method respectively. The image parsing method utilizes probabilistic methods to unify segmentation, detection and recognition. These methods, however, tend to classify/parse an whole image instead of meaningful objects. Furthermore, these methods are usually computationally intense. This method is motivated by the utilization of prior information in curve evolution models.

II. BACKGROUND

Section A shows how an image is represented using a tree of shapes. It also shows how a tree structure is introduced for image representation. Section B describes the relationship between color and geometry in natural images. Section C introduces the techniques of planar curve matching, which can be utilized to compare the boundaries of different objects. Section D introduces the technique of texture Identification, which includes the wavelet transformation which isolates the bands into many sub bands.

A. Image Representation using trees of shapes

The tree of shapes represents images based on the techniques of the contrast-invariant mathematical morphologies [4][7]. This method is based on the theory of image representation using connected components of set of finite perimeters in the space of functions with weakly bounded variations (WBV). The representation of an image using a tree of shapes utilizes the inferior or the superior of a level line to represent an object, and takes the boundary of the inferior area as the shape of the object. Therefore, only the closed shapes are generated. This representation also provides a tree structure to represent the spatial relationship for the objects in an image.

They have several advantages. First, they represent regions instead of curves in an image, which provide a way to handle the contents inside the regions. Second, they are invariant to the contrast changes in an image, which may be
caused by the change of lighting Third, closed boundaries are acquired for each upper level set or lower level set, which can be utilized for shape matching of the regions.

The nesting of level sets provides an inclusion tree for an image. The inclusion tree from the family of upper level sets and the tree from the family of lower level sets, however, can be different if the connected components are directly utilized.

The shapes in the lower level are spatially included in the shapes in the next higher level. The tree of shapes, therefore, provides a natural way to represent the spatial relationships between the shapes in the image. It is straightforward to find upper level sets and lower level sets in an image by thresholding. A tree of shapes can be further constructed by the nesting of level sets. However, this method is computationally intense. This trees of shapes uses a method called as JSegmentation(J-SEG) which is used for segmenting objects. It splits the region based on the user interest.

THE FLLT ALGORITHM

The Fast Level Line Transform (FLLT) [5] constructs a contrast-invariant representation of an image. This algorithm builds a tree which follows the inclusion of the shapes contained in an image. For an image filter, having the contrast-invariant property is interesting. For instance, in the field of document image analysis, this representation is precious to extract characters regardless of whether their are brighter or darker than their surroundings. This document presents how this algorithm is introduced in our image processing Library and shows the results of some connected filters that can be derived from this representation.

Fig.1 Original Object

Lower level set

For an image ima, we define the lower level set of value v as the following set of points:

Lv(ima) = fp 2 domain(ima) j ima(p) _ vg

We will note Lv insteed of Lv(ima) when there is no ambiguity on the input image.

Upper level set

For an image ima, we define the upper level set of value v as the following set of points: Uv(ima) = fp 2 domain(ima) j ima(p) _ vg We will note Uv insteed of Uv(ima) when there is no ambiguity on the input image.

The FLLT take an image input and builds a non-redundant tree i.e. a point of input is stored only once by the nodes of it. The hierarchy follows the inclusion between the shapes. In the following, the input image will be noted input

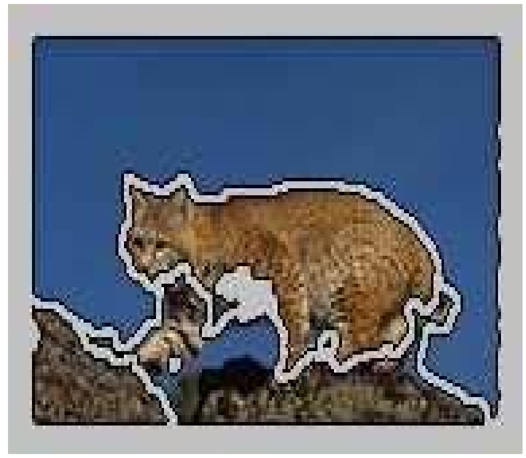


Fig 2 Segmented Animal

ALGORITHM:

For each point pmin that is a local minimum not already in the tree,

1. **Initialization.** Here, we initiate the construction of CC starting with the local minimum
pmin.
_ g = U(pmin)
_ CC a new node with g as value of the tree.
_ A fpming
_ N fg
_ CC:shape:add(pmin)

2. **Growing.**
_ If A = fg then the tree is complete.
_ Enlarge N to its neighborhood:
N N [fx; x 2 neighborhood(A) n CC:shapeg
_ Update gN:
gN min
x2N
U(x)

3. N and A.
Three cases are distinguished:
(a) If gN > g, we meet the parent component.
The value of the parent component is gN. We therefore update g.

Max tree

The Max tree is useful to apply filters on an image. It allows us to select a subset of the shapes of the image according to their characteristics. A traditional example is the area opening and closing of surface (which works on the areas of the shapes.It is also used in astronomical image processing because all the shapes are brighter than their neighborhood. We thus filter the image to clean it and to proceed to automatic classification of galaxies. The Max tree builds a tree of components modeling inclusion of shapes in an image. This tree contains only the shapes having a darker neighborhood.

Min tree

This tree is the dual of the Max tree. It thus contains only the shapes darker that their neighborhood.It is only efficient on the images having bright background and dark shapes, therefore speciallyeffective in analysis of images of documents. Applying a filter on this tree allows quickly extracting all the characters (connected), or arraying elements in an invoice.

Build the Min tree and the Max tree, keeping for each node, one point belonging to each of its holes. This is the most complicated part of the FLLT. Merge the two trees, by filling the holes of the Min tree with nodes of the Max tree and inversely.

THE JSEG METHOD

The J-SEG algorithm [5] provides a faster way to construct a tree of shapes. J-SEG is a pyramidal algorithm based on region growing. At the beginning of each iteration, a J-image is computed. New segments are denied for groups of connected pixels presenting small values and appropriate size. Next, pixels presenting values smaller than the mean J-value of the unsegmented pixels are merged with adjacent existing segments. This step is repeated for a new J-image with smaller window size until a determined minimum size is reached. Finally, a post-processing segment merging step using histogram differences is performed to reduce over-segmentation.

The essential idea of JSEG is to separate the segmentation process into two independently processed stages, color quantization and spatial segmentation.

Color Quantization

Colors in the image are reduced through peer group filtering (PGF) and vector quantization. PGF is a nonlinear algorithm for image smoothing and impulsive noise removal. The result of color quantization is a class-map which associates a color class label to each pixel belonging to the class.

In the first stage, colors in the image are quantized to several representing classes that can be used to differentiate regions in the image. This quantization is performed in the color space alone without considering the spatial distributions. Afterwards, image pixel colors are replaced by their corresponding color class labels, thus forming a class-map of the image. The main focus of this work is on spatial segmentation, where a criterion for "good" segmentation using the class-map is proposed.

The JSEG algorithm [5] is the primary color-texture segmentation tool for this system. Its primary usage is to segment a frame into homogenous regions of color and texture. These homogenous regions of color/texture correspond to the interesting objects to be tracked. JSEG performs this segmentation in a 2-pass algorithm.

The 1st pass detects uniform regions based on quantized color features only, while the 2nd pass uses the color quantization of the 1st pass in a multi-scale technique in order to detect region boundaries. The algorithm allows the user to manually set various thresholds and parameters or use the default parameter values in a more automatic mode. The output of the JSEG algorithm is a segmented image with the region boundaries highlighted as well as a gray-mask image with each region given a distinct label.Applying the criterion to local windows in the class-map results in the "J-image”, in which high and low values correspond to possible boundaries and interiors of color-texture regions. A region growing method is then used to segment the image based on
the multi-scale J-images. For video sequences, a region tracking scheme is embedded into region growing and problems of motion estimation are avoided. The video sequences can be used to achieve consistent segmentation and tracking results, even for scenes with arbitrary non-rigid object motion.

Fig.3 original figure

J-SEG Calculations

The J value can be calculated

$$m_i = \frac{1}{N_i} \sum_{z \in Z_i} z$$

Here m_i mean that is to be calculated for the given image.

$$S_T = \sum_{z \in Z} \left\| z - m \right\|^2$$

Here S_T represents the total variance of the data points.

$$S_W = \sum_{i=1}^{C} S_i$$

Where S_i can be as follows

$$S_i = \sum_{i=1}^{C} \sum_{z \in Z} \left\| z - m_i \right\|^2$$

S_W = Total variance of data point belonging to same class

$$J = S_B / S_W$$

Where

$$= (S_T - S_W) / S_W$$

Here J is the value to be evaluated.These is some of the calculations to be performed for JSegmentation. And the segmented object here below in fig 2 is an example of how regions are splitted. Using the above calculation the JSegmented values are found out.

Spatial segmentation

The second method is the spatial segmentation. It explains if two related objects are there in the image. Then the spatial segmentation is used to identify the image correctly. For example if two tree images are in the same figure it could be easily separated and splitted into two different regions. Using this the J image is classified and the J image is got . Now it is ready for region growing and the desired segmentation results are obtained.

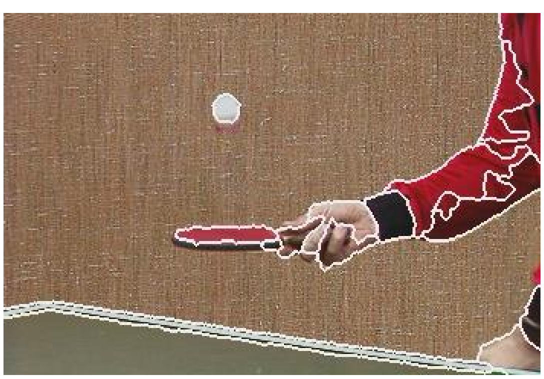


Fig 5 JSEG Figure

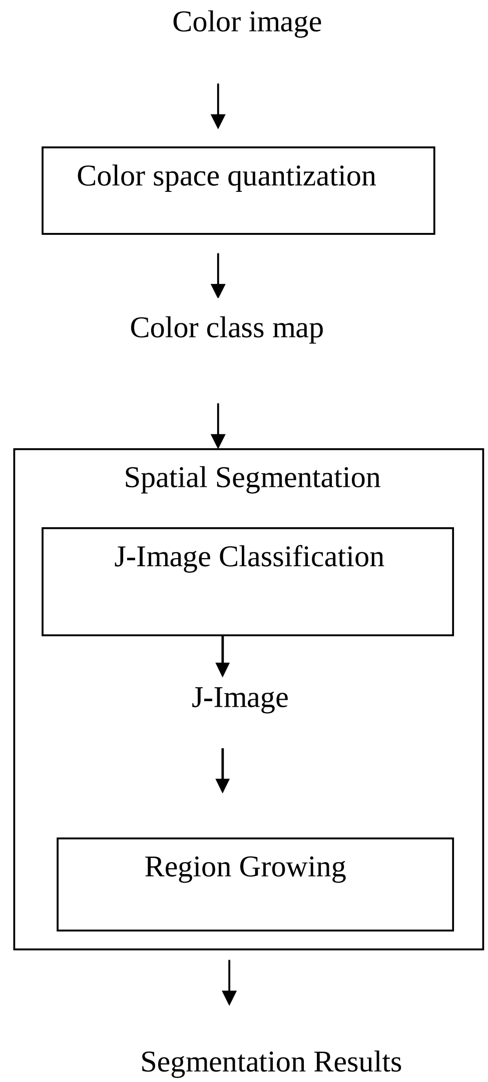


Fig 4: Schematic results of J-seg

Fig 5 explains the steps of J-SEG algorithm, it includes two important steps the first is color quantization and the next is spatial segmentation. In the first step color image is given where quantization is performed and the quantization results in finding out the color maps. The color maps are obtained by finding out the mean for the related pixels.

B. Color and Geometry in Mathematical Morphologies

An improved space HLS (IHLS) is utilized for the color model. The color in every pixel is represented with three channels (Y,S,H), which corresponds to the gray level, saturation and hue respectively. The IHLS space [10], compared to other spaces such as HLS and HSV, has the property of a “well-behaved” saturation coordinate.

The IHLS space always has a small numerical value for near-achromatic colors, and is completely independent of the brightness function.

For a pixel with color(R,G,B) in the(R,G,B) space, and the corresponding pixel with color(Y,S,H) ,

The transformation from the RGB space to the IHLS space is

Y=0.2126R + 0.7152G + 0.0722B

S=MAX(R,G,B)-MIN(R,G,B)

= { 360º-H´, if B>G

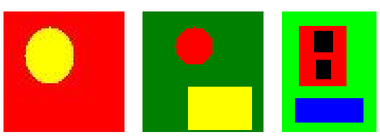
H´, OTHERWISE

Where

$$H' = \frac{R - 0.5G - 0.5B}{(R^2 + G^2 + B^2 - RG - RB - BG)^{1/2}}$$

This method uses the transformation and the total order in to extract the shapes and build the tree of shapes for color images. The inverse transformation from the IHLS space to the RGB space is not utilized. The color image is synthesized so that its grey version contains no shape information. Using the IHLS color space and the total order a tree of shapes is built for the color image.

The color matching is as follows



a b c

Fig 6a,b,c shows the identification of colors.

C. Planar Curve Matching

The method in defines the shape of a curve as a conjunction of shape elements and further defines the shape elements as any local, contrast invariant and affine invariant part of the curve. These definitions are oriented to provide invariance to noise, affine distortion, contrast changes, occlusion, and background.

The shape matching between two images are designed as the following steps.

1) Extraction of the level lines for each image. The level set representations are utilized here for the extraction. The level line is defined as the boundaries of the connected components as shown before.

2) Affine filtering [11] of the extracted level lines at several scales. This step is applied to smooth the curves using affine curvature deformation to reduce the effects of noise.

3) Local encoding of pieces of level lines after affine normalization. Both local encoding and affine normalization

are designed for local shape recognition methods. This step will help to deal with occlusions in real applications.

4) Comparison of the vectors of features of the images. Euclidean distance is utilized to compare the feature vectors.

The performance of curve matching between two curves is calculated after affine filtering, curve normalization and local encoding.

It tells how a boundary can be detected from the figure. It includes four important steps for finding out the exact boundary.

The boundary matching example is as follows

a b c

Fig 7 a, b, c represents the boundary of the images.

2.4. Texture Matching

Image transforms are very important in digital processing they allow to accomplish less with more. For example the Fourier Transform may be used to effectively compute convolutions of images or the Discrete Cosine Transform may be used to significantly decrease space occupied by images without noticeable quality loss. Wavelet Transform (WT) is a relatively new concept as a whole, even it though it incorporates some of the transforms, which have been known for long time.

Texture analysis is important in many applications of computer image analysis for classification, detection or segmentation of images based on local spatial patterns of intensity or color. Textures are replications, symmetries and combinations of various basic patterns or local functions, usually with some random variation. Textures have the implicit strength that they are based on intuitive notions of visual similarity. This means that they are particularly useful for searching visual databases and other human computer interaction applications. However, since the notion of texture is tied to the human semantic meaning, computational descriptions have been broad, vague and sometimes conflicting. The method of texture analysis chosen for feature extraction is critical to the success of the texture classification. However, the metric used in comparing the feature vectors is also clearlycritical. Many methods have been proposed to extract texture features either directly from the image statistics

The texture identification example can be as follows

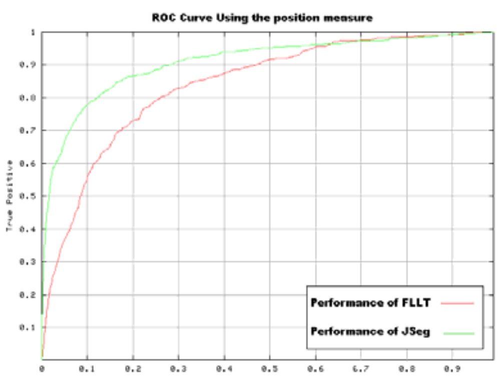
a **b** **c** **d**

Fig 8 a,b,c,d explains the wavelet identification

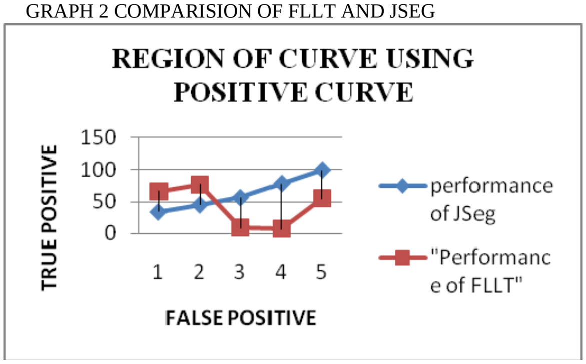
3. PERFORMANCE COMPARISON

Both the algorithms are applied for the trees of shapes and also they are compared based on the time and the quality of the image. The FLLT algorithm can be used to give a quality image and whereas the JSEG algorithm works very well on limited time. They can be done in larger databases. The performance of the Jsegmentation and Fast level line transform algorithm are compared and the performance are plotted as follows

GRAPH 1 ROC CURVE USING POSITIVE MEASURE FOR FLLT AND JSEG



GRAPH 2 COMPARISION OF FLLT AND JSEG



4. SUMMARY AND FUTUREWORK

The method utilizes both the intensity and shape prior information by means of the tree of shapes and boundary matching. It is invariant to contrast change and similarity transformations such as scale, rotation and for the translation. Future research on segmentation methods will be focused on the multi scale analysis is also applied to larger image databases to examine its performance. This can be done for the moving images in the database. A systematic evaluation of the proposed method will be performed on large image databases, and the results will be provided

REFERENCES

[1] J. F. Canny, “A computational approach to edge detection,” *IEEETrans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Jun. 1986.

[2] J. Sethian, *Level Set Methods and Fast Marching Methods*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1999, Cambridge *Monograph.on Applied and Computational Mathematics*

[3] S. Agarwal, A. Awan, and D. Roth, “Learning to detect objects in images via a sparse, part-based representation,” *IEEE Trans. Pattern AnaysisMach. Intell.*, vol. 26, no. 11, pp. 1475–1490, Nov. 2004.

[4] P. Monasse, “Morphological representation of digital images and application to registration,” Ph.D. dissertation, Ceremade, Paris, 2000.

[5] Yining Deng and B.S. Manjunath, “Unsupervised Segmentation of Color-Texture Regions in Images and Video,” IEEE Trans on Pattern Analysis and Machine Intelligence, vol.22, no. 6, p.939-954, 2001

[6] D. Cremers and G. Funka-Lea, “Dynamical statistical shape priors for level set based sequence segmentation,” *Variational and Level Set Methods Comput. Vis.*, pp. 210–221, 2005.

[7] P. Monasse and F. Guichard, “Fast computation of a contrast-invariant image representation,” *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 860–872, May 2000

[8] Y.-G. Wang, J. Yang, and Y.-C. Chang, “Color-texture image segmentation by integrating directional operators into jseg method,” Pattern Recogn. Lett., vol. 27, no. 16, pp. 1983– 1990, 2006.

[9] S. Y. Yu, Y. Zhang, Y. G. Wang, and J. Yang, “Unsupervised color-texture image segmentation,” Journal of Shanghai Jiaotong University (Science), vol. 13, no. 1, pp. 71–75, February 2008.

[10] A. Hanbury and J. Serra, A 3d-Polar Coordinate Colour Representation Suitable for Image Analysis, PRIP, T.U.Wien, Tech. Rep. PRIP-TR-77, 2002.

[11] J. L. Lisani, L. Moisan, P. Monasse, and J. M. Morel, “On the theory of planar shape,” *Multiscale Model Simul.*, vol. 1, no. 1, pp. 1–24, 2003.

Author Profile

Ms. Akila Victor received her MSc software Engineering degree in 2008 from Anna University Chennai and is also a university Rank holder and M.E. degree in Computer Science and Engineering in 2010 from Anna University Tirunelveli, Tirunelveli, India. Presently working as an Asst.Prof in VIT University, Vellore.Her areas of interest are Image Processing, Cryptography and Software

Engineering. She has presented papers in national and International Conferences in various fields. As part of this paper, she is working on developing segmentation preferentially for moving images.