MODERN IMAGE SEGMENTATION TECHNIQUES : PRINCIPLES, PERFORMANCE, AND APPLICATIONS

Paresh M. Tank
Lecturer, Computer Engineering Department,
B & B Institute of Technology, VV Nagar-388120, India.
pareshtank@gmail.com

Hitesh A. Patel
Lecturer, Computer Engineering Department,
B & B Institute of Technology, VV Nagar-388120, India.
hiteshpatelbbit@gmail.com

Abstract

Image Segmentation is a technique that partitions the input image into meaningful regions. The goal is to make the image more simplified and more meaningful to analyze. For this purpose, number of low-level feature available in feature space like Graylevel, Color, Texture, Depth, Motion are utilized. Number of segmentation techniques is available but none of them satisfy the global criteria and thus as per need of an application segmentation techniques should be selected. Many computer vision and machine learning applications involving task such as Object Detection, Pattern Recognition and Object Tracking requires image segmentation as prerequisite step and success of such algorithm depends on accuracy of image segmentation.

This Paper addresses three very popular, versatile and benchmark image segmentation techniques. These three techniques are (1) **Snakes,** the Snake is energy minimizing spline controlled by external constraint energy and influenced by image energy which pull it towards desired feature in image like lines, edges, contours etc. (2) **Normalized Cut,** this technique treats image segmentation problem as graph partitioning problem. To partition a graph Normalized Cut criteria are utilized. (3) **Mean Shift**, it is a non-parametric technique to segment image having complex multimodal background and arbitrary shaped clusters in it.

For each technique underlying concepts, mathematical framework, specific applicability's, results, limitations, and various issues are thoroughly discussed.

Introduction

Nowadays, Digital image processing and video processing are being a widely used subjects for research work in computer engineering. Various image processing techniques are mainly divided into three broad categories where first category called **image enhancement** gives more weight on basic operations on image (Input and Output both are images) and the second category is **image analysis** whose outputs are feature extracted from input image (Input is an image while output is feature). Third category is mainly associated with **computer vision** whose input is images or sequence of images and output may be some logical inference. Image segmentation is one of the important parts of image analysis. Major applications like Object Recognition, Scene Understanding and analysis, Pattern Recognition, Automatic control systems, Remote Sensing, Various Visual surveillance systems, Medical Imagining for Detection of various anomaly uses segmentation as very first step. Number of segmentation techniques is available but still it remains as a challenge for computer vision to satisfy global needs and hence selection of proper techniques is very important as per application [1].

Snakes: Active Contour Model

Introduction

M. Kass at al [3] developed a novel technique for image segmentation, which was able to solve a large class of segmentation problems that had been not possible by conventional techniques. He was interested in developing a model-based technique that could recognize familiar objects in the presence of noise and other ambiguities. Kass proposed the concept of a *snake*, which was an active contour model using an energy minimizing spline guided by external constraint forces and influenced by image forces that pull it toward features such as lines and edges. Snakes have been successful in performing tasks such as edge detection , corner detection, motion tracking, contour detection and stereo matching.

Figure 2.1, Lips modeled as Active Contour (From [5])

The goal is to find a contour that best approximates the perimeter of an object. It is helpful to visualize it as a rubber band of arbitrary shape that is capable of deforming during time, trying to get as close as possible to the target contour. Snakes do not solve the entire problem of finding contours in images. They depend on other mechanisms such as interaction with a user or with some other higher-level computer vision mechanism.

Means snakes method developed originally has been a semi automatic process which requires user interaction in some way. First, the snake is placed near the image contour of interest. During an iterative process, the snake is attracted towards the target contour by various forces that control the shape and location of the snake within the image. Example of such a snake that has attracted towards lips in the figure is show in figure 1.1.

Snake framework

Snakes are example of a more general technique of matching deformable model to an image by means of energy minimization. Mathematically snake is modelled as a spline. A spline is nothing more than a polynomial or set of polynomials used to describe or approximate curves and surfaces. Although the polynomials that make up the spline can be of arbitrary degree, the most commonly used are cubic polynomials. Higher-order polynomials can have undesirable non-local properties. Complex shapes can be decomposed into smaller regions having fewer inflection points by the introduction of appropriately placed knots.

A snake (V) is an ordered collection of **n** points on the image.

V= {v1,v2,……vn }

$\mathbf{v}_i = (x_i, y_i)$

Where $i = 0 \ldots n-1$

We can write its energy function as

$$E^*_{snake} = \int_0^1 E_{snake}(\mathbf{v}(s))\,ds = \int_0^1 \left(E_{internal}(\mathbf{v}(s)) + E_{image}(\mathbf{v}(s)) + E_{con}(\mathbf{v}(s))\right) ds$$

$$E_{external} = E_{image} + E_{con}$$

Where $E_{internal}$ represents the internal energy of the spline (snake) due to bending, E_{image} denotes the image forces acting on spline and E_{con} serves as external constraint forces introduced by user. The combination of E_{image} and E_{con} can be represented as $E_{external}$, which denote the external energy acting on the spline.

So basic snake model is controlled continuity spline under influence of image forces and external constraint forces. The internal spline forces serve to impose a piecewise smoothness constraint. The image forces push the snake toward salient image features like lines, edges, and subjective contours. The external constraint forces are responsible for putting the snake near the desired local minimum. These forces can come from a user interface, automatic attentional mechanisms or high-level interpretation [3].

Snake energy functional

A snake is an energy-minimizing parametric contour that deforms over a series of time steps. Each element along the contour **u** depends on two parameters, where the space parameter *s* is taken to vary between 0 and *N*–1, and *t* is time (iteration).

$$\mathbf{u}(s,t) = (\ x(s,t), y(s,t)\ )$$

At a given time, this function represents a mapping from the parametric domain s e [0, N] into the image plane $\mathbf{R}^2$. An image is regarded as an intensity function defined all over $\mathbf{R}^2$, with a value of zero outside some finite limits. The total energy *Snake* of the model is given by the sum of the energy for the individual snake elements.

$$E_{\text{snake}} = \int_0^{N-1} E_{\text{element}}(\ \mathbf{u}(s,t)\ )\quad ds$$

The energy for each element can be decomposed into three basic energy terms:

$$E_{\text{element}} = E_{\text{internal}}(\mathbf{u}) + E_{\text{external}}(\mathbf{u}) + E_{\text{image}}(\mathbf{u})$$

The parameters *s* and *t* are omitted where no ambiguity arises. The internal energy *Einternal* incorporates regularizing constraints that give the model tension and stiffness. The external energy *Eexternal* represents external constraints imposed by high-level sources such as human operators or automatic initialization procedures. The image (potential) energy *Eimage* drives the model towards salient features; it is usually generated by processing the image to enhance light and dark lines, edges, or terminations. Each of these energy terms produces a corresponding force which can be used to move the model and hence minimize its energy. To implement an active contour model in computer software the continuous representation must be approximated by *N* discrete snake element.

Each of this energy functional is mathematically derived, discussed and its effect on the snake is illustrated in next sub topic.

Advantages of the Snake method

Snakes have following multiple advantages over the other classical image segmentation techniques available in literatures. [3][4]

- They can be controlled interactively by using appropriately placed springs and volcanoes.
- They are easy to manipulate because the external image forces behave in an intuitive manner.
- They are autonomous and self-adapting in their search for a minimal energy state.
- They can be made sensitive to image scale by incorporating Gaussian smoothing in the image energy function.
- They are relatively insensitive to noise and other ambiguities in the images because the integral operator is an inherent noise filter.
- They can be used to track dynamic objects in temporal as well as the spatial dimensions.

Limitation and issues

Snakes are not without their drawbacks, Main among all is listed below. These drawbacks are only related to the Snake method developed by M Kass at al. [2]

- They can often get stuck in local minima states; this may be overcome by using simulated annealing techniques at the expense of longer computation times.
- They often overlook minute features in the process of minimizing the energy over the entire path of their contours.
- Their accuracy is governed by the convergence criteria used in the energy minimization technique; higher accuracies require tighter convergence criteria and hence, longer computation times
- Snakes cannot move toward objects that are too far away (because the snake falls into the closest local energy minimum)
- Snakes cannot move into boundary concavities.

To overcome limitation and issues related to seminal snake method various works has been done. Many advance methods like Dynamic snake, Balloon –pressurized snake, Gradient vector fields based on original work has been developed. These methods are not in the scope of this paper.

Normalized Cut

Introduction

A graph cut is the process of partitioning a directed or undirected graph into disjoint sets. The concept of optimality of such cuts is usually introduced by associating energy to each cut. Problems of this kind have been well studied within the field of graph theory but for graphs with large numbers of the nodes are very difficult. Nevertheless, ever since it became apparent that many low-level vision problems can be posed as finding energy minimizing cuts in graphs. These techniques have received a lot of attention in the computer vision community. Graph cut methods have been successfully applied to stereo, image restoration, and texture synthesis and image segmentation. It is taking a global image feature descriptor as a weighted graph and reduces segmentation to an optimal graph partitioning. For partitioning graph ,various graph cut techniques are proposed in literature like Minimum cut, Average cut, Normalized cut. In this chapter Normalized cut method is discussed.

Graph partitioning and Related approach

The set of points in an arbitrary feature space is presented as a weighted undirected graph ***G=(V,E).*** Nodes of the graph are the points in the feature space. An edge is formed between every pair of nodes and the weight on each edge ***W(i,j)*** is a function of the similarity between nodes **i** and **j**. A graph ***G=(V,E)*** is partitioned into two disjoint complementary sets A and B, $B=V-A$, by removing the edges connecting two parts. The degree of dissimilarity between two sets can be computed as a total weight of removed edges. That closely relates to a mathematical formulation of a cut .

A graph ***G=(V,E)*** can be partitioned into two disjoint sets, by simply removing edges connecting the two parts. The degree of dissimilarity between these two pieces can be computed as total weight of the edges that have been removed. In graph theoretic language, it is called the **cut.**

$$cut(A,B)=\sum_{u\in A,v\in B} w(u,v)$$

The optimal BI partitioning of a graph is the one that minimizes this cut value. Although there are an exponential number of such partitions, finding the minimum cut of a graph is a well-studied problem and there exist efficient algorithms for solving it.

Wu and Leahy proposed a clustering method based on this minimum cut criterion. In particular, they seek to partition a graph into k-sub graphs such that the maximum cut across the subgroups is minimized. This problem can be efficiently solved by recursively finding the minimum cuts that bisect the existing segments. This globally optimal criterion can be used to produce good segmentation on some of the images. [6]

However, the minimum cut criteria favors cutting small sets of isolated nodes in the graph. This is not surprising since the cut defined above increases with the number of edges going across the two partitioned parts. Fig. 1 illustrates one such case. Assuming the edge weights are inversely proportional to the distance between the two nodes, the cut that partitions out node n1 or n2 will have a very small value. In fact, any cut that partition out individual nodes on the right half will have smaller cut value than the cut that partitions the nodes into the left and right halves.

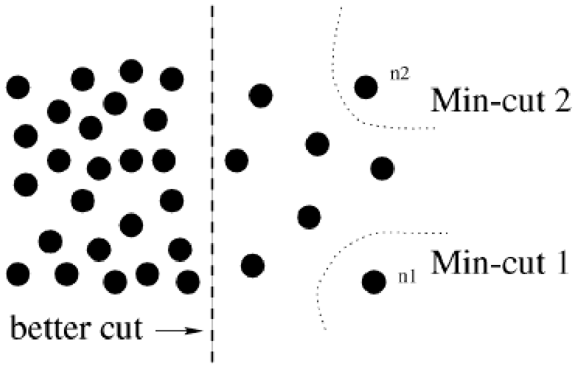

Figure 3.1 Case of minimum cut where isolated points are favored[6]

To avoid this unnatural bias for partitioning out small sets of points, Shi and Malik [6] proposed a new measure of disassociation between two groups. Instead of looking at the value of total edge weight connecting the two partitions, they compute the cut cost as a fraction of the total edge connections to all the nodes in the graph and called this disassociation measure the normalized cut (Ncut):

Normalized cut

Shi and Malik [6] propose a modified cost function, normalized cut, to overcome the problem involved in minimum cut. Instead of looking at the value of total edge weight connecting the two partitions, the proposed measure computes the cut cost as a fraction of the total edge connections to all nodes. Mathematically it is;

$$Ncut(A,B) = \frac{cut(A,B)}{asso(A,V)} + \frac{cut(A,B)}{asso(B,V)},$$

Where *asso(A,V)* is the total connection from nodes in A to all nodes in the graph and *asso(B,V)* is similarly defined.

$$asso(A,V) = \sum_{u \in A, t \in V} w(u,t)$$

With this definition of the disassociation between the groups, the cut that partitions out small isolated points will no longer have small Ncut value, since the cut value will almost certainly be a large percentage of the total connection from that small set to all other nodes.

In similar fashion, a measure for total normalized association within groups for a given partition can be defined.

$$Nassoc(A,B) = \frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B)}{assoc(B,V)},$$

Where *assoc(A,A)* and *assoc(B,B)* are total weights of edges connecting nodes within A and B, respectively. This is also an unbiased measure, which reflects how tightly on average nodes within the group are connected to each other. Another important property of this definition of association and disassociation of a partition is that they are naturally related [6]. Proof is given below.

Hence, the two partition criteria in this grouping algorithm, minimizing the disassociation between the groups and maximizing the association within the groups, are in fact identical and can be satisfied simultaneously. In this algorithm, this normalized cut used as the partition criterion. Unfortunately, minimizing normalized cut exactly is NP complete, even for the special case of graphs on grids.

However, when the normalized cut problem relaxed into the real value domain, an approximate discrete solution can be found efficiently [6].

$$
\begin{aligned}
Ncut(A,B) &= \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)} \\
&= \frac{assoc(A,V) - assoc(A,A)}{assoc(A,V)} \\
&\quad + \frac{assoc(B,V) - assoc(B,B)}{assoc(B,V)} \\
&= 2 - \left(\frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B)}{assoc(B,V)}\right) \\
&= 2 - Nassoc(A,B).
\end{aligned}
$$

Application

Normalized cut is a versatile technique, which is suitable for large class of problem related to image segmentation. Some of the applications are mentioned below.

- Simple and powerful methods to segment images.
- Flexible and easy to apply to other clustering problems
- Object tracking
- It direct segment into shapes space so therefore very useful to guide shape model.
- Motion segmentation

Figure 3.2 Image segmentation using normalized cut [1] samples

Issues and Limitation

These techniques also have some demerits. Most of them are listed below.

- High memory requirements (use sparse matrices).
- Very dependant on the scale factor for a specific problem.

- Solving a standard eigenvalue problem for all eigenvectors takes $O(n^3)$ time. This is impractical for some image segmentation applications where n is the number of pixels in the image.

Mean Shift

Introduction

Mean Shift is a powerful and versatile non parametric iterative algorithm that can be used for lot of purposes like finding modes, clustering, analysis of multi modal feature space and to delineate arbitrary shaped contour in it etc. Mean Shift was introduced in Fukunaga and Hostetler [10] and has been extended to be applicable in other fields like Computer Vision.

This section provides an intuitive idea of Mean shift and the later sections will expand the idea. **Mean shift considers feature space as an empirical probability density function [11]. Main intuitive idea of Mean shift is that, if we want to draw more samples from the same probability distrib**ution, then those points would probably be close to the points that we have already drawn. Build distribution by putting a little mass of probability around each data-point.

If the input is a set of points, then Mean shift considers them as sampled from the underlying probability density function. If dense regions (or clusters) are present in the feature space, then they correspond to the mode (or local maxima) of the probability density function. We can also identify clusters associated with the given mode using Mean Shift.

For each data point, Mean shift associates it with the nearby peak of the dataset's probability density function. For each data point, mean shift defines a window around it and computes the mean of the data point. Then it shifts the center of the window to the mean and repeats the algorithm till it converges. After each iteration, we can consider that the window shifts to a denser region of the dataset.

At the high level, we can specify Mean Shift as follows

1. Fix a window around each data point.
2. Compute the mean of data within the window.
3. Shift the window to the mean and repeat till convergence.

Kernel density estimation

Kernel density estimation is a non parametric way to estimate the density function of a random variable. This is usually called as the Parzen window technique. Given a kernel K, bandwidth parameter h ,input $x_1, x_2, x_3, \ldots x_n$ be the n data samples from d-dimensional Euclidean space, R^d.

Then estimated probability density function for data sample would be

$$p(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

K(x) is known as Kernel. Kernel is function that satisfies following properties.

K(x) is nonnegative i.e. $K(x) \geq 0$

$$\int_{R^d} K(x)dx = 1$$

Some example of the kernel function which are frequently used in practice are given below. Three kernel funtions with name in first column, its equation in second column and pictorial view in third column are displayed here.

The quality of a kernel density estimator is measured by the mean of the square error between the density and its estimate, integrated over the domain of defination. In practice, howerver, only an asymptotic approximation of this measure can be computed.

Mean shift estimation

The first step in the analysis of a feature space with the underlying density f(x) is to find the modes of this density. The modes are located among the zeroes of the gradient and the mean shift procedure is an elegant way to locate these zeros without estimating the density.

The density gradient estimator is obtained as the gradient of the density estimator which is shown below.

$$\hat{\nabla} f_{h,K}(\mathbf{x}) \equiv \nabla \hat{f}_{h,K}(\mathbf{x}) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^{n} (\mathbf{x} - \mathbf{x}_i) k' \left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right).$$

Assuming that deriative of the kernel profile k exists for all x. By defining the function

$$g(x) = -k'(x),$$

Now using *g(x)* for profile, the kernel *G(x)* is defined as following, where *Cg,d* is the corresponding normalization constant. The kernel *K(x)* is called shadow of the kernel *G(x)* . Now introducing g(x) into equation of the gradient that is already defined.

$$\begin{aligned} \hat{\nabla} f_{h,K}(\mathbf{x}) &= \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{x}) g \left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right) \\ &= \frac{2c_{k,d}}{nh^{d+2}} \left[\sum_{i=1}^{n} g \left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right) \right] \left[\frac{\sum_{i=1}^{n} \mathbf{x}_i g \left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right)}{\sum_{i=1}^{n} g \left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right)} - \mathbf{x} \right] \end{aligned}$$

Here in last equation, first term is constant, second term represents the density estimation at x computed with the kernel *G* . Third term shows the mean shift. So the difference between ther weighted mean, using kernel G for weight, and x, the centre of the kernel window is given by.

$$\mathbf{m}_{h,G}(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}_i g \left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right)}{\sum_{i=1}^{n} g \left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right)} - \mathbf{x}$$

The above expression shows that, at location x, the mean shift vector computed with kernel *G* is proportional to the normalized density gradient estimate obtained with kernel *K*. The normalization is by the density estimate in x computed with the kernel *G.*

Thus mean shift vector always points toward the direction of maximum increase in density. This relation intuitively shows local mean is shifted towards the region in which mejority of the point resides. So mean shift vector path leads to stationary point of the estimated density. The mean shift procedure obtained by successively computatioin of the mean shift vector *M* and translaton of the kernel window *G(x)* by *m*.

Applicatioin

Mean shift is very simple technique having imazing results. It has wide range of the application . Most of the application are listed below. Out of this we have explored three technique in this paper . following sub sections deals in detail with these three application of the Mean shift.

- Clustering
- Density estimation
- Mode seeking
- Discontinuity preserving smoothing
- Object contour detection
- Segmentation

- Object tracking

Summary

In this Paper, three very popular and benchmark image segmentation techniques have been thouroghly studied and presented as outcome of study. Overall summary of this paper are given below.
Image Segmentation is a technique that partitions the input image into meaningful regions. The goal is to make the image more simplified and more meaningful to analyze. For this purpose, number of low-level feature available in feature space like Gray level, Color, Texture, Depth, Motion are utilized. There are many image segmentation techniques have been in existence. Out of these three versatile techniques have been addressed.

This paper addressed three very popular, versatile and benchmark image segmentation techniques. These three techniques are (1) **Snakes,** the Snake is energy minimizing spline controlled by external constraint energy and influenced by image energy which pull it towards desired feature in image like lines, edges, contours etc. (2) **Normalized Cut,** this technique treats image segmentation problem as graph partitioning problem. To partition a graph Normalized Cut criteria are utilized. (3) **Mean Shift**, it's a non-parametric technique to segment image having complex multimodal background and arbitrary shaped clusters in it.
In this paper, for each technique underlying concepts, mathematical framework, specific applicability's, results, limitations and various issues have been thoroughly studied and discussed.

References

[1] Shital Raut at al, "Image segmentation-State of art survey for prediction ", ICACC, IEEE -2008
[2] Michael Kass et al, "Snakes: Active contour models.", International Journal of Computer Vision, pages 321-331, 1987.
[3] http://en.wikipedia.org/wiki/Active_contour_model
[4] http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/RAMANI1/node31.html
[5] R Szeliski, "Computer vision, algorithm and application", Springer 2010
[6] J Shi and J Malik, "Normalized cut and image segmentation", vol.22, NO. 8, TPAMI, IEEE, 2000
[7] Tutorial on image segmentation by Charless Fowlkes at al at University of California at Berkeley
[8] www.wisdom.weizmann.ac.il/~deniss/vision_spring04/files/mean_shift/mean_shift.ppt
[9] Dorin Comaniciu and Peter Meer, "Mean Shift : A Robust approach towards feature space analysis," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol 24, No 5, May 2002.
[10] Fukunaga and Hostetler, "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition", IEEE Transactions on Information Theory vol 21 , pp 32-40 ,1975
[11] www.saravananthirumuruganathan.wordpress.com/introduction-to-mean-shift