IMAGE SEGMENTATION

Abstract - Image segmentation is one of the fundamental approaches of digital image processing. It is the process of dividing an image into multiple segments which can be used to locate objects and boundaries in images. There are many applications where Image segmentation is transforming industries: Detecting cancerous cells, Traffic Control Systems, Self Driving Cars, Locating objects in satellite images etc. This report reviews main image segmentation algorithms.

Introduction

Image segmentation is a method that creates multiple layers and fragments of images from an image or picture. Segmentation in concept is very simple. By looking at an image, we can tell what a picture contains. Visually it is very easy to locate a region of interest; doing so with a computer algorithm is tricky. Image segmentation algorithms help computers and machines in telling one object apart from another when scanning an image.

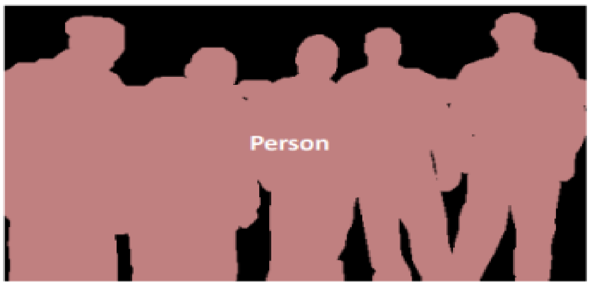
We can divide or partition the image into various parts called segments. By doing so, we can make use of the important segments for processing the image. An image is a collection or set of different pixels. Image segmentation assigns a value to each pixel which is then grouped together according to their similarity in areas like colour, saturation, and distance between them. This way, the image is fragmented into different parts which we can work on without altering the whole image, just the selected fragment. There are three levels of image analysis:

- *Classification* – classifying the whole image into a class such as "people", "animals"
- *Object detection* – detecting objects in an image and drawing a box around them, for example, a person or a vehicle.
- *Segmentation* – identifying parts of the image and deciding what object they belong to. Segmentation is the basis for performing object detection and classification.

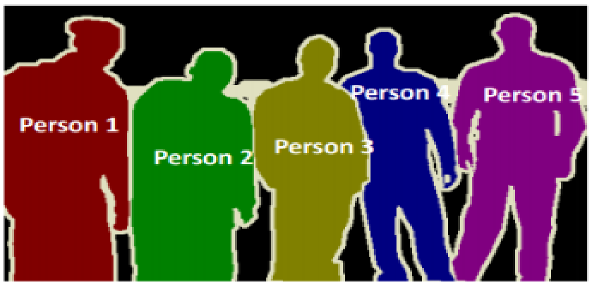
Semantic Segmentation vs. Instance Segmentation

The segmentation process itself, has two methods:

- ***Semantic segmentation***—classifies all the pixels of an image into meaningful classes of objects. For example, we can highlight all the pixels associated with a cat. This is also known as dense prediction because it predicts the meaning of each pixel. (***Classification*** *assigns a single class to the whole image whereas* ***semantic*** *segmentation classifies every pixel of the image to one of the classes.*)
- ***Instance segmentation***—identifies each instance of each object in an image. Semantic segmentation differs in that it doesn't categorize every pixel. If there are three vehicles in an image, semantic segmentation classifies all of them as one class, whereas instance segmentation identifies each individual vehicle.



Semantic Segmentation



Instance Segmentation

Discontinuity and homogeneity of the pixels with respect to their local neighbourhood are used in many segmentation methods. Segmentation methods based on discontinuity property of pixels are known as boundary or edges based techniques while the ones based on similarity or homogeneity are region based techniques.

Region Based Segmentation

The main idea here is to classify a particular image into a number of regions or classes. For each pixel in the image we need to decide or estimate which class it belongs to. There are a variety of approaches to do region based segmentation.

Threshold Segmentation

Thresholding divides an image into a foreground and background. A specific threshold value separates pixels into one of two levels to differentiate objects. By selecting a threshold value T, the gray level image is converted to binary image. The pixel value falling below or above that threshold is classified accordingly (background or object). By creating a binary image first, complexity of the data reduces and the process of recognition and classification gets simpler.

Threshold techniques can be categorized into two classes: **global threshold** and **local (adaptive) threshold**. In global threshold, a single threshold value is used for the whole image. If we want to divide the image into two regions (object and background), we define a single threshold value. If the image contains multiple objects along with the background, multiple thresholds (local threshold) are created using local information around the pixel.

A common method to select T is by analyzing the histograms of the images that are to be segmented. Otsu's method looks at the histogram and tries to minimize the within-class variance. Simple images consist of an object and a background. The background is usually one gray level and covers a major part of the image. Therefore, a large peak represents the background gray level in the histogram. A smaller peak represents the object, which is another gray level. But it only considers the gray information of the image, hence is sensitive to noise and grayscale unevenness. Some of the advantages of this method are that the calculations are simple so fast operation speed. When object and background have high contrast, this method performs really well. But when there's not a significant difference in the grayscale values, or there is an overlap of the grayscale pixel values, it becomes very difficult to get accurate segments.

Regional Growth Segmentation

Region growing segmentation technique works by using pre-defined criteria to group pixels or sub-regions. The aggregation starts from some pixels (seeds) which represent distinct image regions and grows these regions using the criteria for growth mechanism and checks the homogeneity of the regions after each growth step by features like similarity between grayscale, texture, colour, shape.

1) Begin by selecting an arbitrary seed pixel and compare it with neighbouring pixels.
2) Region is grown by adding in neighbouring pixels that are similar thereby increasing the size of the region.
3) When the growth of one region stops, choose another seed pixel which does not belong to any region and start again.
4) This process is continued until all pixels belong to some region.

Edge Detection Segmentation

There is always an edge between two adjacent regions with different grayscale values (pixel values). The edges can be considered as the discontinuous local features of an image. This discontinuity comes in handy in detecting the shapes of multiple objects present in an image. It can often be detected using derivative operations which can be calculated using differential operators. The operators are basically weight matrix, which is an approximation to a derivative of an image; it is applied (convolved) over the image. The output depends on the values of the weight matrix which helps to extract information from the input. Some specific values for these weight matrices helps us detect horizontal or vertical edges (or even the combination of horizontal and vertical edges).

One such weight matrix is the ***Sobel operator***. It is typically used to detect edges. The sobel operator has two weight matrices – one for horizontal and vertical edges each.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

The horizontal and vertical gradient approximations of each pixel of the image can be combined to calculate the size of the gradient using the following formula:

$$G = \sqrt[2]{G_x^2 + G_y^2}$$

Unlike the Sobel, the ***Laplacian edge detector*** uses only one matrix. It calculates second order derivatives in a single pass. Here's the kernel used for it:

0	-1	0
-1	4	-1
0	-1	0

The laplacian operator

-1	-1	-1
-1	8	-1
-1	-1	-1

The laplacian operator (include diagonals)

It is more relevant when only the position of the edge is concerned irrespective of the pixel gray scale difference around it. One drawback when we're working with the laplacian edge detector is that it is extremely sensitive to noise as it uses second order derivatives.

Laplacians are computationally faster to calculate and are usually applied to an image that has first been smoothed using some Gaussian Smoothing filter so as to reduce its sensitivity to noise.

Segmentation based on clustering

Clustering is the process of grouping data points on the basis of some similarity factor such that data points in the same groups have more common features than those in other groups. These groups are called clusters. One of the most commonly used clustering algorithms is k-means. K-means clustering algorithm identifies clusters in the data, where variable K is the number of groups by computing the means of the distance between the data points. The algorithm assigns each data point (or pixel) to one of the groups based on the distance between the points. Clustering works iteratively to form groups, rather than analysing predefined groups.

Steps in K-Means algorithm:

1. Choose K number of clusters.
2. Select random K points.
3. Assign each data point to the closest centre.
4. Compute the new centre for each cluster and set it as new centre.
5. Reassign each data point to the new closest centre. If change in centres is significant, go to step 4, otherwise, the model is ready.

The objective of K-Means clustering is to minimize the sum of squared distances between all points and the cluster centre. But k-means being a distance-based algorithm is not suitable for clustering non-convex clusters.

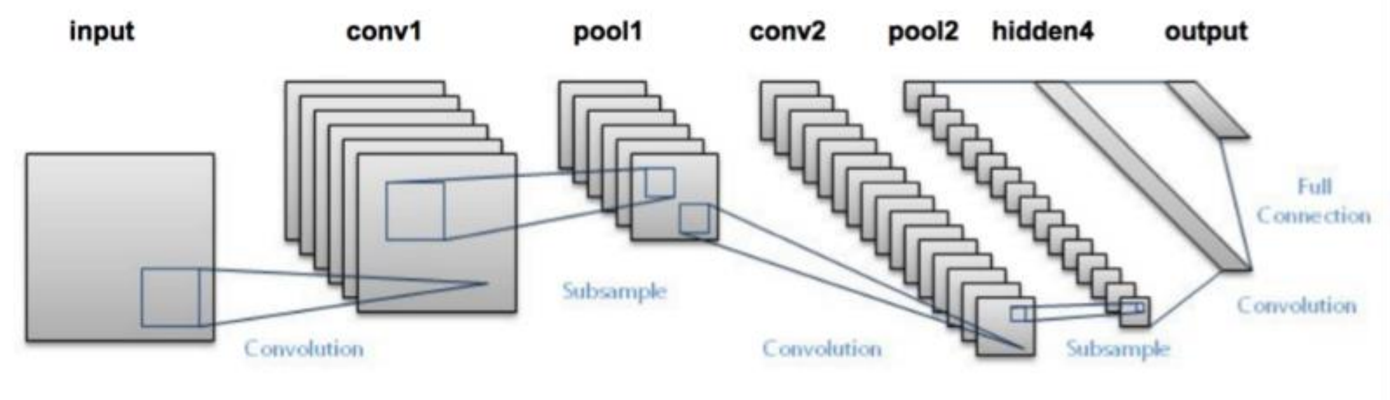
Segmentation based on CNN

Modern image segmentation techniques are powered by deep learning technology. Image segmentation with Convolutional Neural Networks (CNNs) involves feeding multiple segments of an image as input to the network, which classifies the pixels as a feature. The CNN does not process the whole image at once rather scans the image, looks at small filtered subsample of several pixels until it has mapped the entire image.

Every network layer in the network acts as a detection filter for the identification of specific features or patterns present in the original data. The first layers in a CNN detect large features that can be recognized and interpreted easily. Later layers detect features that are more refined and are present in larger features detected by previous layers. The last layer can make very precise classification by combining all the specific features in the input data detected by previous layers.

CNN consists of a number of Convolution and Pooling layer pairs before reaching the output layer which gives the output in the form of a class.

1) An image is converted into an array where each cell represents a value.
2) Then filters are applied to find features of the image (convolution).
3) Then the image size is reduced by pooling layer.
4) After pooling layer it is passed through the activation function like RELU (Rectified Linear Units) that decides the final value of a neuron.
5) Then we apply a dropout layer to remove some neurons to prevent over fitting.
6) Then a dense (fully-connected) layer is applied to connect all neurons where each neuron gives a probability of image accuracy.



Conclusion

Image segmentation helps determine relations between objects and the context of objects in an image. As the basic technique of image processing and applications in satellite image analysis, autonomous vehicles and especially in medical field, image segmentation has a promising future. Although there are a number of segmentation algorithms, none of them can apply to all types of images and segmentation techniques usually aim at certain application. Since the result of image segmentation is affected by lots of factors, such as: homogeneity of images, spatial structure character of the image, continuity, texture, image content, physical visual character and so on, universal segmentation algorithm has become the focus of contemporary research.