# Active Contour Algorithm

An active contour model, also called a snake is technique for detecting and segmenting objects using deformable curves which will match the objects. It relies on a deformable model controlled by an energy minimization function. This energy depends on the image and also on the parameter we want to give to our deformable contour.

## 1.1 Model

By creating an active contour, we want to create a curve whose behavior is going to be constrained by two aspects. The first and most important is due to the objective which is to perform a segmentation based on object and shape detection. So, we need our contour to converge to the edges of the object we are interested in. The second constraint lies on the model of the contour we want to have. Indeed, we want to obtain a deformable model which means that we would like to have a continuous curve where the curvature will match the shape curvature. So this implies to have a theoretical model we can implement to describe the contour’s own behavior.

### External Energy:

The external energy is the component of our behavior function that describe how the deformable curve will match with objects of the image. Indeed, the external energy is the function that will constrain the contours displacement. To have such a function in an image we are going to use its gradient. Indeed, around edges the gradient is presenting these two characteristics of presenting a local extremum and having a monotonic behavior.

### Internal Energy:

The internal energy is the component of the behavior function that describe the physical properties of our contour like smoothness or continuity and curvature. The internal energy is the component of the behavior function that describe the physical properties of our contour like smoothness or continuity and curvature.

## 2. Active Contour Model:

### 2.1 Gaussian Smoothing

An important part of the set up to perform a segmentation using active contour is to preprocess the image using a Gaussian filtering. If our input image has a good quality, its edges are going to be extremely well defined, which means that there are going to be abrupt. This is going to be bad for our optimization algorithm and the snake energy evolution because the external energy relies on the use of the image gradient. If our input image has a good quality, its edges are going to be extremely well defined, which means that there are going to be abrupt. This is going to be bad for our optimization algorithm and the snake energy evolution because the external energy relies on the use of the image gradient.

### 2.2 Contour Construction

We draw the big circle around object in the image. Then, the size of the circle decreases until get the edges of the object and the shape of the circle changes around to the object.

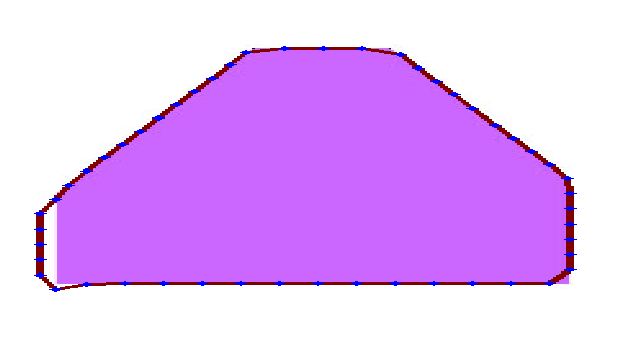
### 2.3 Greedy Method:

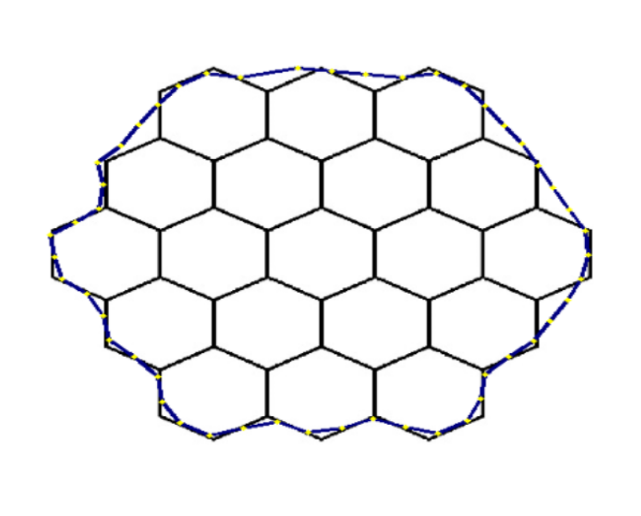
The greedy method is an implementation technique used to simplify the implementation of the minimization of energy without having to perform an optimization algorithm technique such as the gradient descent. It works under the assumption that finding for each point of the contour the closest local energy minimizing neighbor will converge to the overall global minimum of the contour.

## How the function work in GUI:

when you open the tab of active contour, you found two button "Start" & "Clear". when you press start button, we get the image and circle around the image and this circle decreases until getting the edges of object. To stop this operation, in any time you should press "clear" button which make the circle to stop decreasing. before closing the app,you should press "clear" button to stop the operation and remove the image.

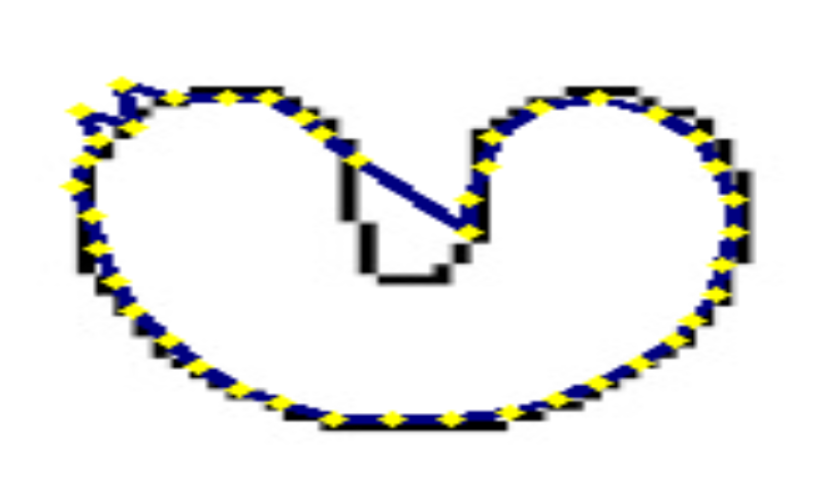
# Results of Active Contour:

The first test to see if our implementation was working was to analyze the evolution of the contour. In this implementation, we had to see each point being displaced over time in the way we selected them because we are iterating over points to make them move. the set-up of the α, β and coefficient does play an important role in the shape of the contour.



After testing in previous image examples, we have to test the implementation on the images like in the lecture. When we test the algorithm, we get the good results like:





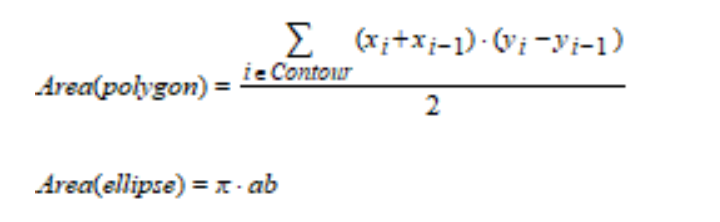
## Area And perimeter:

A perimeter is either a path that encompasses/surrounds/outlines a shape in two dimensions or its length one-dimensional. The perimeter of a circle or an ellipse is called its circumference.

Calculating the perimeter has several practical applications. A calculated perimeter is the length of fence required to surround a yard or garden. The perimeter of a wheel/circle (its circumference) describes how far it will roll in one revolution. Similarly, the amount of string wound around a spool is related to the spool's perimeter; if the length of the string was exact, it would equal the perimeter.

### Area:

The Area is calculated from the shapes contour (periphery), i.e. the closed polygon that surrounds the feature. The area is calculated using:



In the GUI, we calculate the perimeter and area as follow:

