Intelligent Scissors

# TA: Mohamed Raafat

## T093

|  |  |
| --- | --- |
| منة الله إبراهيم محمد عفيفي  أحمد محمد توفيق محمد  أيمن مصطفى حسن حسين | 20191700665  20191700062  20191700162 |

Abstract

Selection tools can be used to select objects in an image to resize/delete/copy/move the objects. There are many types of selection tools such as rectangles or free-form selection tool, sometimes free-form selection tools are called Lasso’s. You can imagine a lasso as a rope surrounding your selection. Unfortunately, selection using ordinary lasso’s can be tedious and boring. In Photoshop, there is a more advanced version of ordinary lasso’s called Magnetic Lasso Tool. Magnetic Lasso is a tool that automatically snaps to an Objects’ Boundaries.

The technical term for the Magnetic Lasso Tool is Livewire or Intelligent Scissors. In this project we want to implement a simple magnetic lasso to learn more about image processing, graphs, and greedy algorithms.



Contents

* [Graph Construction](#_Graph_Construction)
* Calculating Shortest Path
* Backtracking Shortest Path
* Drawing Path on Image
* Generating a sequence of connected paths using multiple anchor points
* Frequency
* Scaling small Images
* Cropping Final Image

# Graph Construction

• Initializing *adj* List<> -> ->

## private static void constructGraph(RGBPixel[,] ImageMatrix)

• Loops on pixels to add the weights of neighboring pixels

• Main Complexity:

• Two nested loops looping on the pixels of the Image -> ->

• Helpers:

• addEnergyFor(int row, int col, Vector2D pixelEnergies) ->

• addEnergyToNeighbour(..) ->

Note: List<>.Add() is considered O(1) since the initial capacity of the List after adding the first element is **4** and the maximum edges per vertex is **4** so the List<> won’t need the O(N) expansion

• Final Complexity:

• Notes: