7MR10070: Software and Robotic Integration

Semester 2

Assignment 1

Written by Alexandros Megalemos

# Task 1

(a)

Point p = (x, y, z)

List of points

List of output points

For each p in :

Retrieve pixel intensity c, where

If c = 1 then

(b)

Point

Point ep is a point p representing an entry point

Point tp is a point p representing a target point

List of entry points

List of target points

List of ventricle points

List of output points

Line l is a line between two points

For each ep in :

For each tp in :

For each p in

If does not then

(c)

Point

Point ep is a point p representing an entry point

Point tp is a point p representing a target point

List of entry points

List of target points

List of blood vessel points

List of output points

Line l is a line between two points

For each ep in :

For each tp in :

For each p in

If does not then

(d)

Point

Point ep is a point p representing an entry point

Point tp is a point p representing a target point

List of entry points

List of target points

List of blood vessel points

List of output points

Line l is a line between two points

For each ep in :

For each tp in :

If angle <= 55 then

# Task 2

For each point in input:

Get the coordinates (x, y, z) of the point

Retrieve the pixel value of (x, y, z) in the image

If pixel value == 1:

Add the point to the list

(b)

For each entry point in entry points:

For each target point in target points:

Get the line between the two points

Get the points of the line

validLine = true

For each point on the line:

If point passes through ventricle:

validLine = false

break

if validLine:

add (entry, target) point to valid points list

(c)

For each entry point in entry points:

For each target point in target points:

Get the line between the two points

Get the points of the line

validLine = true

For each point on the line:

If point passes through blood vessel:

validLine = false

break

if validLine:

add (entry, target) point to valid points list

(d)

For each entry point in entry points:

For each target point in target points:

Get the line, lineET, between the two points

Get perpendicular line where lineET passes through the cortex

Calculate the angle between the two lines

If angle <= 55:

Add (entry, target) point to valid points list

# Task 3

TODO

# Task 4

TODO