MINI Project #2 Report

3D Shape Search Engine

Shuaiyu Liang (sl5352)

1. Introduction to submitted bag

Matlab code:

File name	Description
Acquire_EucDis.m	This code will read mesh from settled directory and compute pair wise Euclidean distance between different points. A file named Edescriptor_*.dat will be created in current directory contains Euclidean distance descriptor for the entire dataset.
Acquire_GeoDis.m	This code will read mesh from settled directory and compute pair wise Geodesic distance between different points. A file named same as mesh name will be created in directory of C:\Users\carly\Documents\Matlab\3DComputerVision\project2\**different_path_name**\. Each file contains current read in mesh's pair wise geodesic distance.
Geodescriptor.m	This code read in pair wise Geodesic Distance Matrix and compute Geodesic distance descriptor, create a file named Geodescriptor_*.dat in current directory.
Dijkstra_sl5352_V3.m	This is a function I created for compute pair wise geodesic distance using Dijkstra algorithm.
P-Rcurve.m	This code compute and plot two different descriptor performance based on different read in descriptor data. The consequence will be talked more in detail in second part of this report.
ShapeSeacrhGUI.m	Provided GUI search engine

Dataset, file folder name:

Folder name	Description
smallTOSCA	Original dataset
smallTOSCASMALLmissing	Small part cut in meshlab
smallTOSCAMEDIUMmissing	Medium part cut in meshlab
smallTOSCALARGEmissing	Large part cut in meshlab
smallTOSCAsmallNoise	Add small noise on original dataset
smallTOSCAmediumNoise	Add medium noise on original dataset
smallTOSCAhighNoise	Add large noise on original dataset

Dat file (basically just descriptor under different situation):

ED file name	GD file name	Description
Edescriptor_original	Gdescriptor_original	Original
Edescriptor_smallnoise	Gdescriptor_smallnoise	Small noise
Edescriptor_mediumnoise	Gdescriptor_mediumnoise	Medium noise
Edescriptor_highnoise	Gdescriptor_largenoise	Large noise
Edescriptor_smallmissing	Gdescriptor_smallmissing	Small parts missing
Edescriptor_mediummissing	Gdescriptor_mediummissing	Medium parts missing
Edescriptor_largemissing	Gdescriptor_largemissing	Large parts missing

2. My Dijkstra implementing

2.1 Time consume

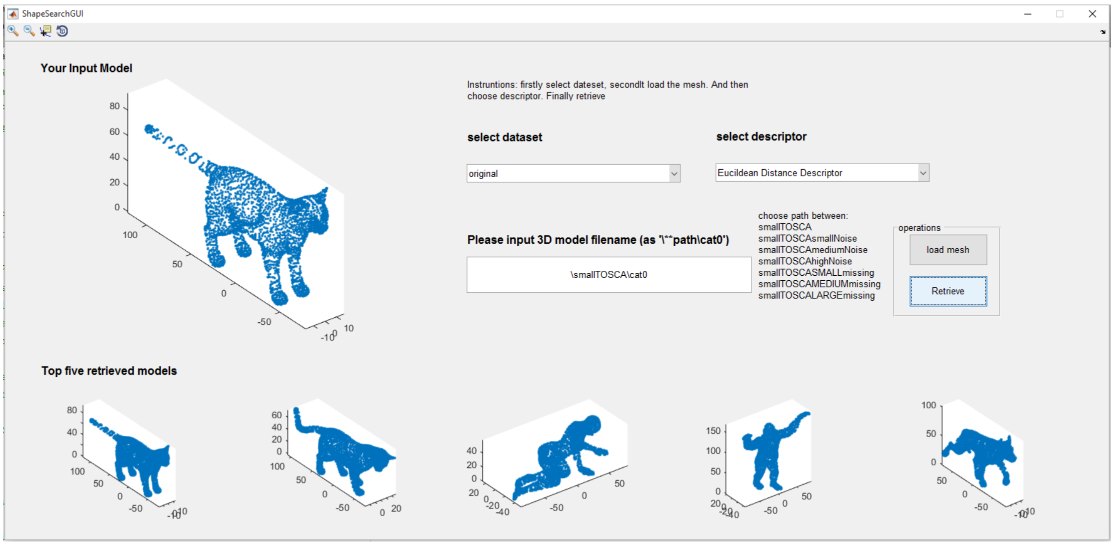
Uploaded Dijkstra_sl5352_V3 usually takes about 2 to 3 minutes to finish the point wise-entire mesh computing. The longest time it got was 5 minutes, and shortest time it got was less 1 minute depends on numbers of points each model owns.

2.2 Flaws

In this function, in order to shorten the entire computing time, I only computing upper triangle point wise geodesic distance. This idea should be right, however the way I implement this idea leave out some point directly. For instance, if point number 3 has to go to point number 50 via point number 1 if taking the shortest path. Originally I just do not want to compute the distance from point3 to point1 so I leave out point1 from entire searching loop. Then point 50 cannot be reached because of this. And thus leads to my Geodesic distance matrix to have some inf. length filled.

Also step#2 and step#4 of my version of code is still computational heavy, according to my code.

3. My GUI



Instruntions: firstly select dateset, secondlt load the mesh. And then choose descriptor. Finally retrieve

select dataset: original

select descriptor: Eucildean Distance Descriptor

Please input 3D model filename (as '**path\cat0'): \smallTOSCA\cat0

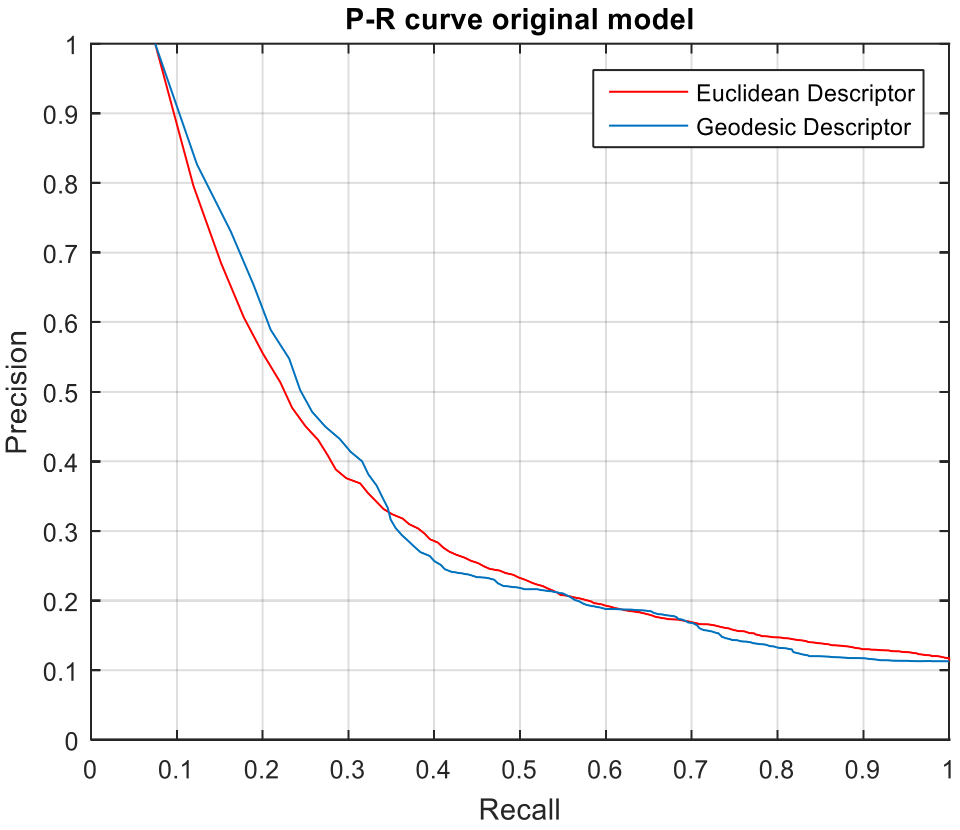
choose path between:
smallTOSCA
smallTOSCAsmallNoise
smallTOSCAmediumNoise
smallTOSCAhighNoise
smallTOSCASMALLmissing
smallTOSCAMEDIUMmissing
smallTOSCALARGEmissing

Overall layout is shown above. When using, typing in file name and some part of directory (like static text next to the edit text), load the mesh, select dataset and descriptor, then click retrieve. You MUST select one of the datasets and descriptors.

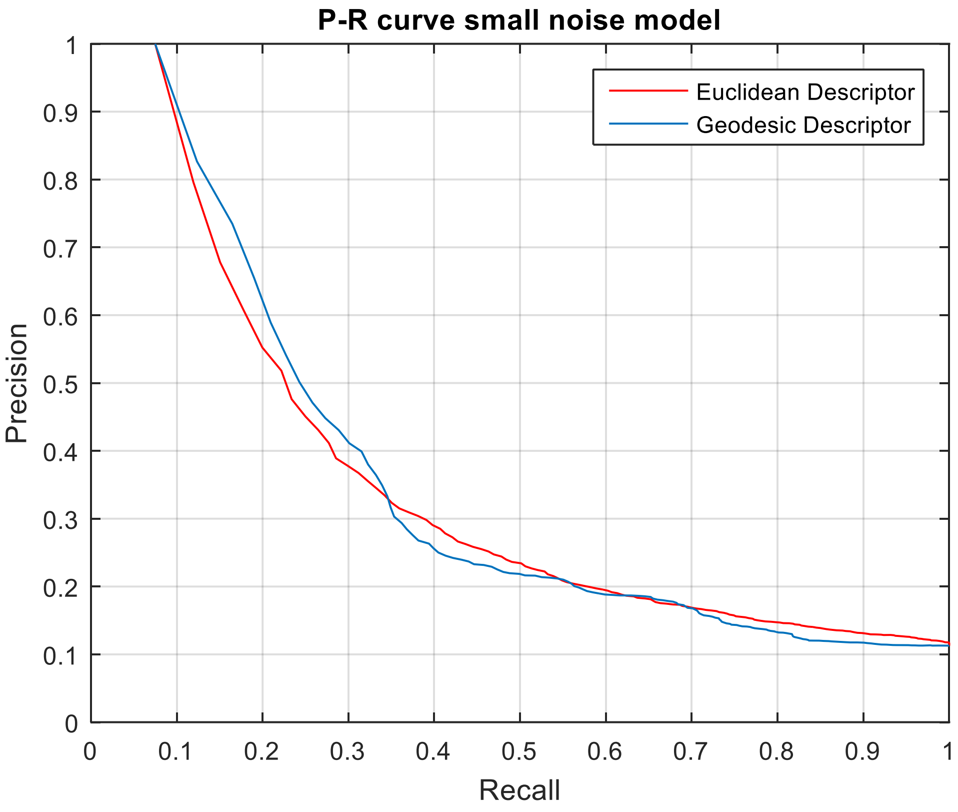
According to GUI, when I input original cat0 and select Geodesic Descriptor, it somehow get some lion from dataset, which is fun but not right. So P-R curve talked in the following section could to some degree show descriptor preforms however not totally map that out.

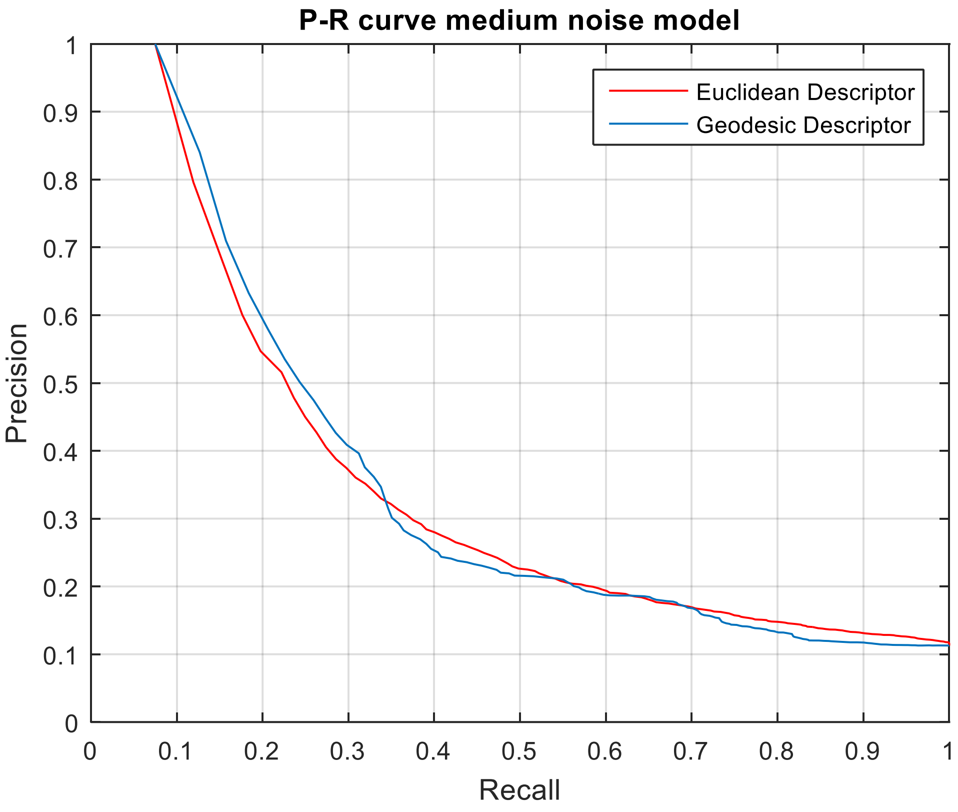
4. Precision-Recall curve evaluation

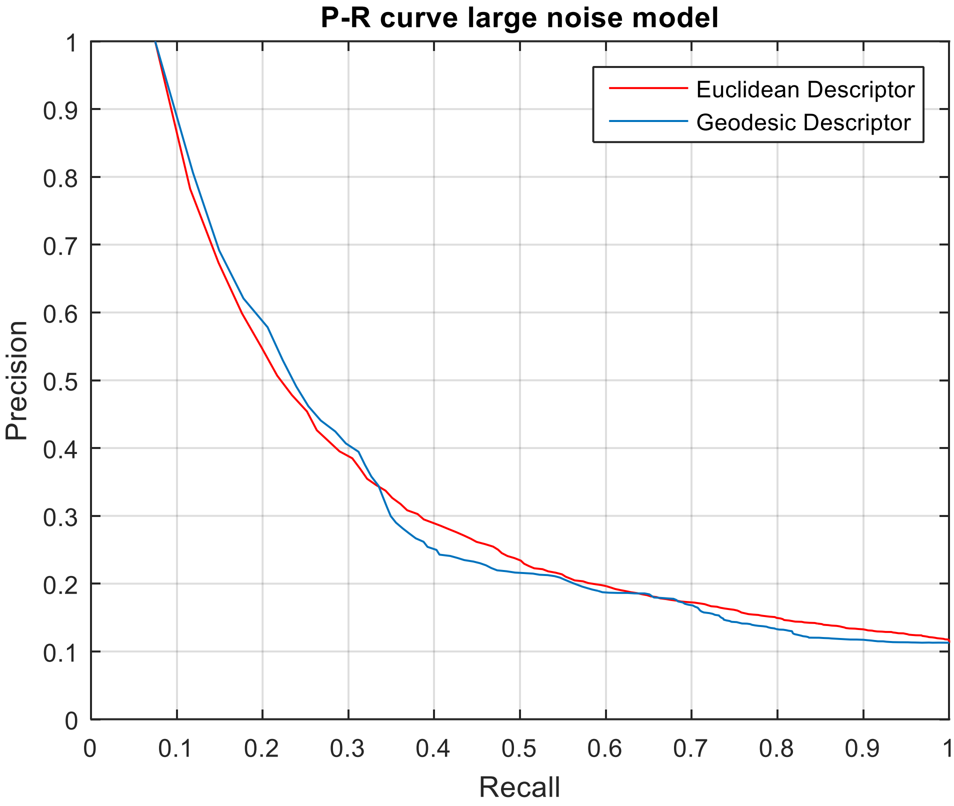
Here are seven P-R curve under different situation. The overall performance of these two descriptor kind of descriptors are very similar. They both drop down fast, and yield not so ideal output. Except for the original dataset, and small noise dataset, which Geodesic Descriptor performed slightly better than Euclidean Descriptor.



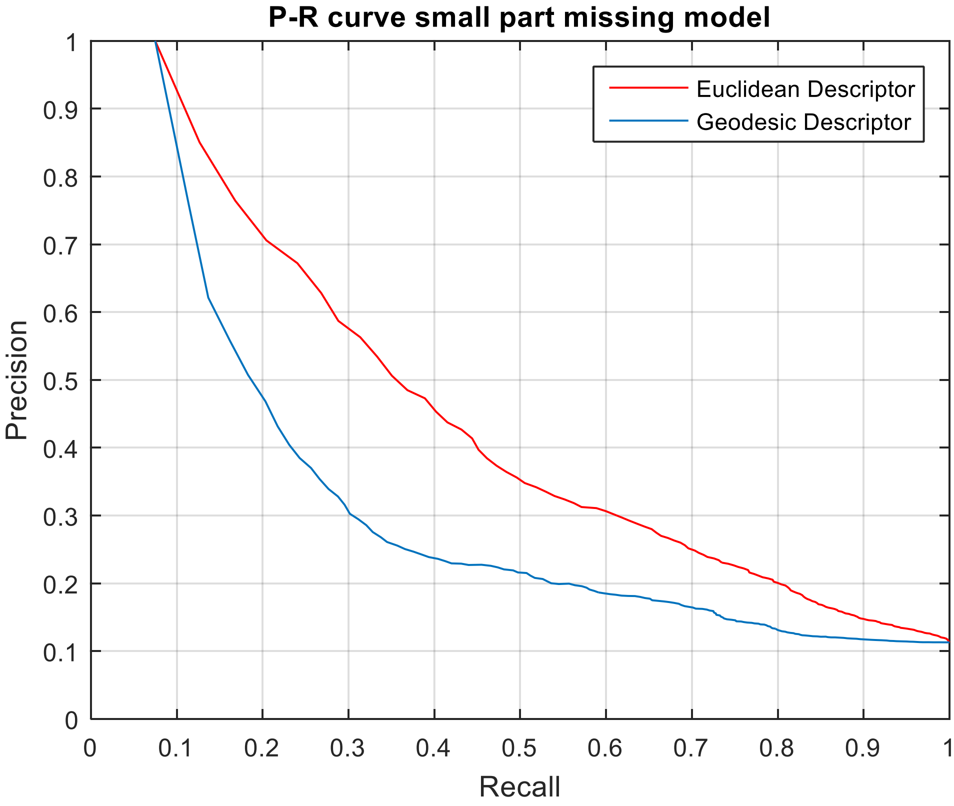
When noise going up, Geodesic Descriptor performance are getting worse. As you could see Euclidean's performance did not get much change according to grid. Geodesic Descriptor's curve moving towards Euclidean's. That is probably caused by Geodesic's non-robust to noise property.


P-R curve small noise model
Precision
Recall
Euclidean Descriptor
Geodesic Descriptor
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

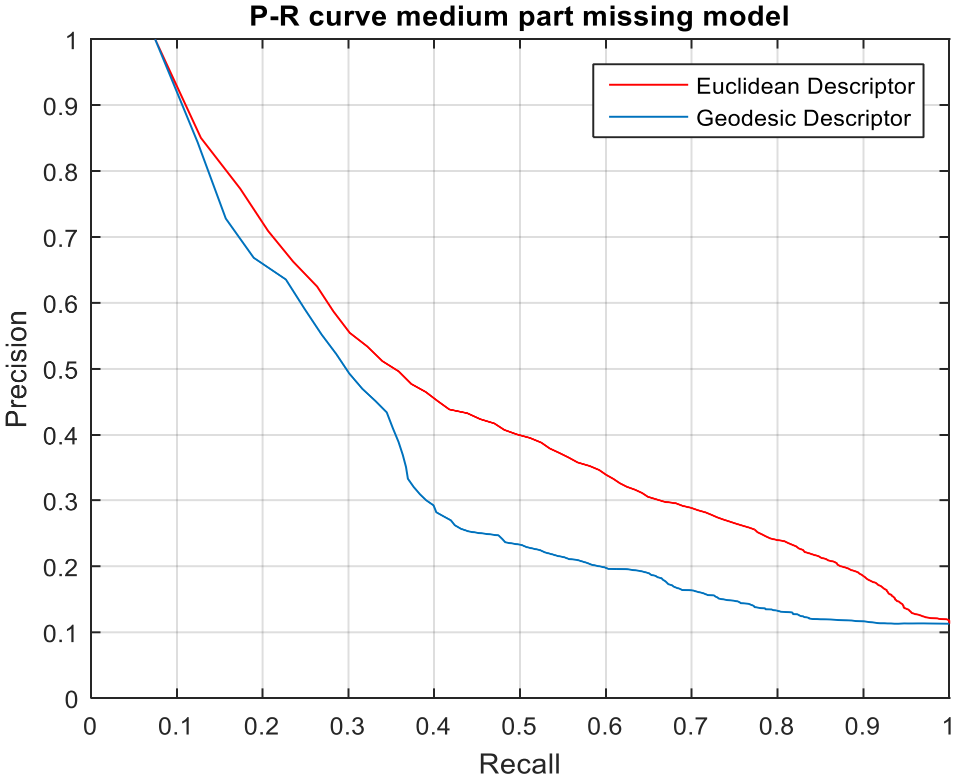

P-R curve medium noise model
Precision
Recall
Euclidean Descriptor
Geodesic Descriptor
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

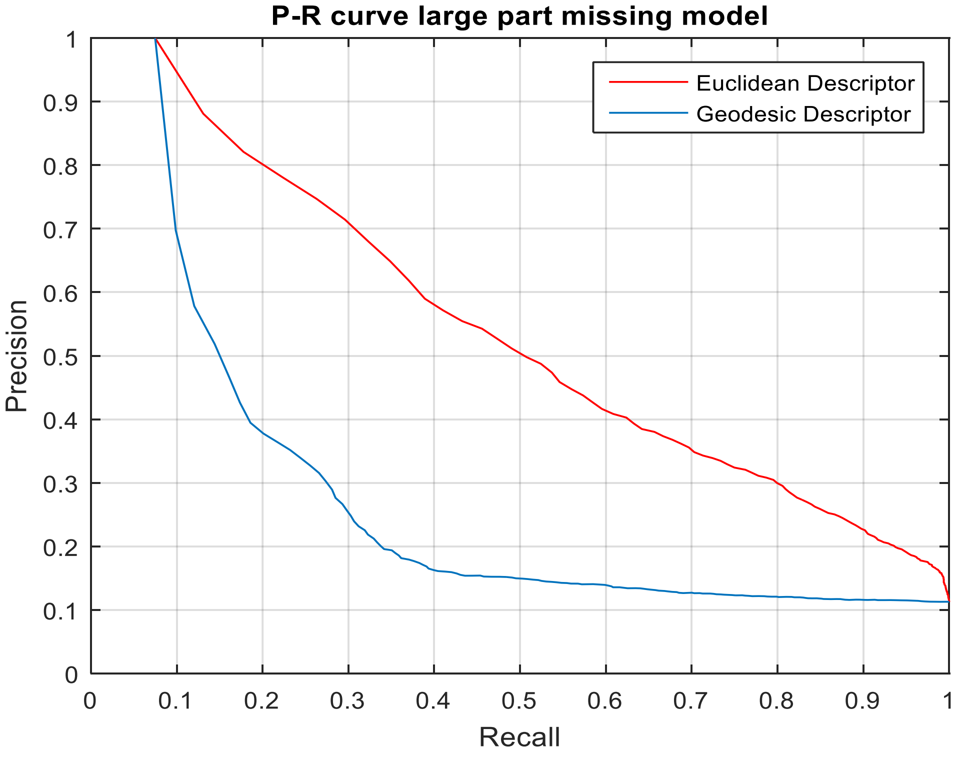


When it comes to partial missing dataset, the Geodesic Descriptor got a dramatic down fall rather than Euclidean Descriptor. ED, however, raised its performance also in a very large scale. One reason could cause this is when cutting mesh in meshlab, some part of geometric complex



part are cut off, which leads to a more stable recognition performance of Euclidean Descriptor. For reason of geodesic curve drop down could also be cutting leads to many break of certain point, which leads to the difficulties of finding the neighbors if some certain path is cut. In this case may lead to many infinity value in Geodesic distance matrix and thus leads to non-preferable output.





5. Overall performance evaluation

MAP	ED	GD
Original	0.4141	0.4159
Small noise	0.4146	0.4151
Medium noise	0.4125	0.4124
Large noise	0.4146	0.4050
Small missing	0.5062	0.3744
Medium missing	0.5298	0.4344
Large missing	0.5986	0.3229

My GD that implemented my version of Dijkstra algorithm got some fatal flaws that leads to dramatic drop in MAP of partial missing.

6. Reference

[1] Dijkstra algorithm https://www.youtube.com/watch?v=gdmfOwyQlcI

[2] Matlab GUI callback http://www.mathworks.com/help/matlab/creating_plots/callback-definition.html