Overview

Friday, November 5, 2021 9:20 PM

Name 1	Hongyi Fan
Email	hfan15@jhu.edu
Other contact information (optional)	
Name 2	Yuxin Chen
Email	ychen506@jhu.edu
Other contact information (optional)	
Signature (required)	I (we) have followed the rules in completing this assignment Hongyi Fan. Yuxin Chen

Overview

Contents of this document:

- Program Structure
- Algorithm of Distortion Correction Function
  Algorithm Of Pivot Calibration With Distortion Correction
  Algorithm For Computing F_reg and Tooltip Position In CT Frame
- Validation and Results
  A tabular summary of the results obtained for unknown data

A short statement of who did what:

Yuxin Chen:

We went through the whole assignment together and discussed the algorithm and mathematical approach we decided to use. Then I mainly wrote the function and report for computing the C_expected value and distortion correction function. We wrote the report together.

HongYi Fan:

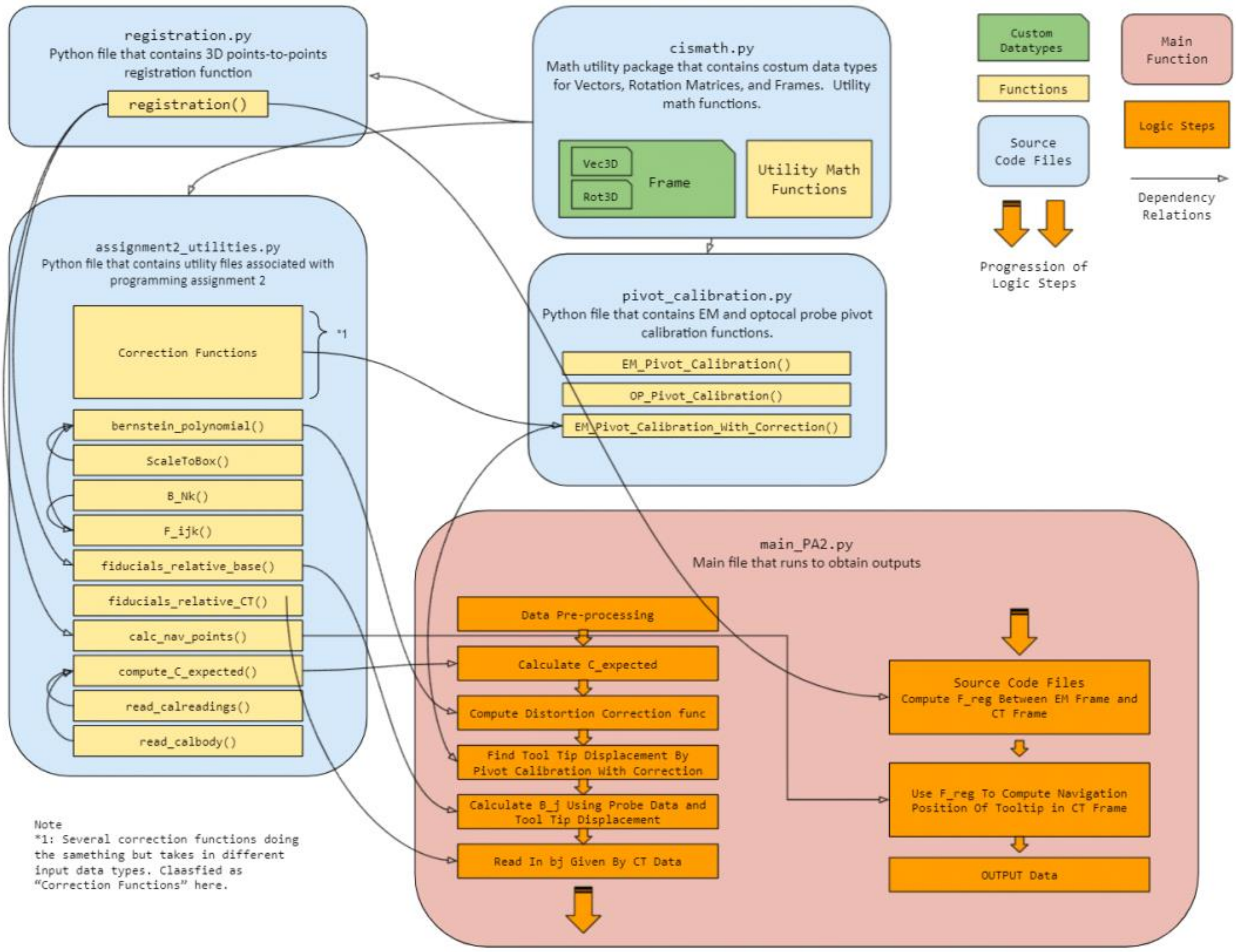
Went through the assignment and algorithms with teammate. Wrote the majority of the code for pivot calibration and cismath package.
Contributed in programming of pa2_utilities and main_PA2 scripts.
Contributed in report writing and validating

External Resources:

- Numpy (https://numpy.org/)
- Matplotlib

Program Structure

Friday, November 5, 2021 9:20 PM



Algorithm of Distortion Correction Function

Tuesday, November 2, 2021 21:16

Algorithm of computation of the expected values $\overrightarrow{\mathrm{C}_{\mathrm{i(expected)}}}$ for the $\vec{\mathrm{C}}$ has shown in the previouse PA1 report.

Distortion Correction Function

Section 1: build the distortion correction function

Known:

- Input: measured data set(C value from the calreading.txt file), known ground true data set(C_expected we calculated), other measured data we will correct distortion in this assignment

Goal:

- Output: coefficient used in the distortion correction function

Steps :

1. We build a "tensor form" interpolation polynomial using 5th degree Bernstein polynomials:

$$F_{ijk}\left(u_x, u_y, u_z\right) = B_{5,i}(u_x)B_{5,j}\left(u_y\right)B_{5,k}(u_z)$$

The mathematical approach taken:

Bernstein Polynomials:

$B_{N,k}(v) = \binom{N}{k}(1-v)^{N-k}v^k$ where $\binom{\mathrm{N}}{\mathrm{k}}$ is a binomial coefficient, $\binom{\mathrm{N}}{\mathrm{k}} = \frac{\mathrm{N}!}{\mathrm{k}!(\mathrm{N}-\mathrm{k})!}$

$\mathrm{P}(c_0, \ldots, C_N; v) = \sum_{k=0}^{N} c_k B_{N,k}(v)$

2. Since Bernstein polynomial does great job in the range between 0 and 1, we need to scale the input value, the measure value, to the range between 0 and 1. To scale our input value, we use the following equation ScaleToBox(q, $q^{min}, q^{max}) = \frac{q - q^{min}}{q^{max} - q^{min}}$

   We will save this scale box(q^{min}, q^{max}), since we will use this same scale box to scale our measured data to the range between 0 and 1 when we apply the correction function later. To make sure this scale box include all the measure value we will correct in this assignment, we find the q^{min}, q^{max} for all the measure value will correct in this assignment,include C and G measure values.
3. Then we can setup a least square problem to solve for the coefficient of Bernstein Polynomial for the correction function:

$$\begin{bmatrix} & \vdots & \\ F_{000}(\overrightarrow{u_s}) & \cdots & F_{555}(\overrightarrow{u_s}) \\ & \vdots & \end{bmatrix} \begin{bmatrix} c_{000}^x & c_{000}^y & c_{000}^z \\ \vdots & \vdots & \vdots \\ c_{555}^x & c_{555}^y & c_{555}^z \end{bmatrix} = \begin{bmatrix} & \vdots & \\ p_s^x & p_s^y & p_s^z \\ & \vdots & \end{bmatrix}$$

Section 2: apply the distortion correction function

Known:

- Input: coefficient we get from last section, measured data set

Goal:

- Output: Corrected the distortion in the measured data set, return a corrected data set.

Steps :

1. Then the correction function will be:

$\vec{\mathrm{p}} = \mathrm{CorrectDistortion}(\vec{q})$

$\{ \ \vec{u} = ScaleToBox(\vec{q}, \overrightarrow{q_{min}}, \overrightarrow{q_{max}})$

Return $\sum_{i=0}^{5}\sum_{j=0}^{5}\sum_{k=0}^{5} \overrightarrow{c_{i,j,k}} B_{5,i}(u_x)B_{5,j}\left(u_y\right)B_{5,k}(u_z)$

$\}$

2. To apply the distortion correction function, we will first scale the input data, the measured value, to range between 0 and 1 by using the same scale box(q^{min}, q^{max}) as we used when we calculate the coefficient.
3. Then we will use the coefficient we get from last section to get the corrected value by using the following equation :

$$corrected\ value = \sum_{i=0}^{5}\sum_{j=0}^{5}\sum_{k=0}^{5} \overrightarrow{c_{i,j,k}} B_{5,i}(u_x)B_{5,j}\left(u_y\right)B_{5,k}(u_z)$$

Algorithm Of Pivot Calibration With Distortion Correction

2021年11月7日 0:10

Known:

- Multiple data frames of "Positions of markers fixed on a tool, $distorted\_G_i$, in respective to some fixed coordinate frame", during a pivot calibration procedure.
- The *coefficient* for distortion correction, the degree , N, of Bernstein polynomial, and the scale box (a min value and a max value, usually float type) of Bernstein polynomial.
- Tool tip is at the some position P_{dimple} during the calibration process

Goal:

- Calculate $\vec{p}_t$ that represents the displacement from tool tip to the tool trackers (centroid of all tracker markers)

STEPS:

1. Apply distortion correction to all points $distorted\_G_i$ to get corrected G_i

   For every point:

   $$G_i = correction\_function(G_i(distorted),\ coefficient,\ N, scalebox\ )$$

   Where correction function is the distortion correction function implementing the algorithm introduced in previous section "*Algorithm of Distortion Correction Function*".

2. Compute a reference coordinate frame using the first frame of corrected calibration data

   We denote the i'th data of k'th data frame here as G_i^k.

   First we calculate a position vector, G_0, as a reference position

   $$G_0 = \frac{1}{\#of\ markers} \cdot \sum_{i}^{\#of\ markers} G_i^1$$

   $$\vec{g}_i = G_i^1 - G_0$$

   The set of vectors g_i*[i = 1 ... $\#of\ markers$]* describes the displacement between each marker and the centroid of the markers. These vectors are used to calculate the pose of the tool. Note that g_i should be the same for all data frames because the markers on a probe is always fixed relative to the probe's local frame. We can denote the probe's local frame as F_{local}. This local frame should be used for all future probe pose calculations.

3. For each data frame, compute the pose of the tool relative to the tracker base.

   For the k_{th} data frame, the transformation F_k satisfies:

   $F_k \cdot \vec{g}_i = \vec{G}_i^k$

   Where $\vec{g}_i$ is the set of vector computed in step 1, G_i^k {i = 0... #of markers} is the set of locations of markers in the kth data frame.

   F_k can be computed using the registration algorithm developed in part 2.

   Now we have F_k that is the transformation between the initial data frame and the kth data frame

4. Compute p_t using a set of F_k

It is known that in the kth data frame:
$F_k \cdot \vec{p}_t = \vec{p}_{pivot}$
$R_k\vec{p}_t + \vec{p}_k = \vec{p}_{pivot}$
$R_k\vec{p}_t - \vec{p}_{pivot} = -\vec{p}_k$

We can write above equation in the form below:

$$\begin{bmatrix} \vdots & \vdots \\ R_k & -I \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vec{p}_t \\ \vec{p}_{pivot} \end{bmatrix} = \begin{bmatrix} \vdots \\ -\vec{p}_k \\ \vdots \end{bmatrix}$$

Since R_k, I, p_k are known values, we can use a least square root method to calculate $\begin{bmatrix} \vec{p}_t \\ \vec{p}_{pivot} \end{bmatrix}$

A = $\begin{bmatrix} \vdots & \vdots \\ R_k & -I \\ \vdots & \vdots \end{bmatrix}$, B = $\begin{bmatrix} \vdots \\ -\vec{p}_k \\ \vdots \end{bmatrix}$

AX = B

Code:
```
X = numpy.linalg.lstsq(A,B)
```

Now we have a 6-d vector that consists of $\vec{p}_t$ *and* $\vec{p}_{pivot}$

Algorithm For Computing F_reg and Tooltip Position In CT Frame

2021年11月7日 1:37

Known:

- The displacement of tool tip relative to the probe frame $\vec{p}_{tip}$
- Multiple frames of data, each contains the positions of EM markers $\vec{G}$ on the probe relative to the EM tracker when the tool tip is in contact with the fiducials on the bone
- The *coefficient* for distortion correction, the degree , N, of Bernstein polynomial, and the scale box (a min value and a max value, usually float type) of Bernstein polynomial.
- The displacement vectors, $\vec{b}_j$ of fiducials relative to the CT frame
- Multiple frames of data, each contains the positions of EM markers $\vec{L}$ on the probe relative to the EM tracker when the tool tip is moving during the operation
- The local frame F_{local} used during the pivot calibration procedure. That is, the Frame associated with the set of $\vec{g}$ vectors obtained using the first frame of data in pivot calibration

Goal:

- Calculate F_{reg}, the transformation from the EM tracker frame to the CT frame
- Calculate $\vec{p}_{nav}$, the position vector of the tool tip relative to the CT frame for several probe poses

STEPS:

1. Apply distortion correction to each value of $\vec{G}$ using the method introduced in "*Algorithm Of Pivot Calibration With Distortion Correction*".

2. Find the probe pose F_k relative to the EM tracker for each frame of data using the method introduced in "*Algorithm Of Pivot Calibration With Distortion Correction*". Note that F_{local} here is used to calculate F_k instead of finding a new local probe frame using a new frame of data.

3. Find the position vectors of each fiducials, $\vec{B}_j$, relative to the EM tracker

   F_k is the pose of the probe when the tool tip is in contact with the k_{th} fiducial. Therefore, we can calculate the position of the k_{th} fiducial, $\vec{B}_k$ by:

   $$\vec{B}_k = F_k \cdot \vec{p}_{tip}$$

4. Calculate F_{reg} by previously developed point cloud to point cloud registration algorithm *(See appendix)*
   It satisfies:
   $\vec{b}_k = F_{reg} \cdot \vec{B}_k$
   F_{reg} is the transformation from EM tracker frame to CT frame.

5. Apply distortion corrections to all values of $\vec{L}$ using the method introduced in "*Algorithm Of Pivot Calibration With Distortion Correction*".

6. Calculate the pose of the probe for each data frame F_k using the method introduced in "*Algorithm Of Pivot Calibration With Distortion Correction*".

7. Calculate the tool tip position relative to the EM frame, then calculate the tool tip position relative to the CT frame.
   For each F_k, we obtain the tool tip position relative to the EM tracker as follow:
   $\vec{t}_k^{EM} = F_k \cdot \vec{p}_{tip}$

   Since F_{reg} is the transformation from EM tracker frame to CT frame,
   we obtain the tool tip position relative to the CT frame for each frame:

   $\vec{t}_k^{CT} = F_{reg} \cdot \vec{t}_k^{EM}$

Validation and Results

Friday, November 5, 2021 9:21 PM

To verify the result, we have the following unit testing:

1) Verify that the distortion correction function is correct
2) Verify that the pivot calibration is correct with distortion correction
3) Verify that the tip location with respect to the CT image are correct

The verification of point to point registration and computation of $\overrightarrow{C_{i(expected)}}$ has shown in the PA1 report. In this PA2, we used the same algorithm and code as in PA1.

1) Verify that the distortion correction function is correct by using the debug dataset "pa2-debug-a"

To verify that our distortion correction function is correct we used the debug dataset "pa2-debug-a". In "pa2-debug-a", there is no EM distortion, EM Noise and OT jiggle. So the distortion correction function should return the same value as the given measured data.
Thus, we first test our distortion correction function on the "pa2-debug-a". After we used the C value from "pa2-debug-a-calreadings" file and the C_expected we calculated in the first step to calculate the coefficient of Bernstein Polynomials we used for distortion correction function, we first tested it on the C value from "pa2-debug-a-calreading.txt" in "pa2-debug-a" dataset. We expected the C value after corrected should be very close to C_expected we calculated(ground truth) in "pa2-debug-a" dataset.

Table of the error in C value after corrected(C_corrected) and C_expected(ground truth value) we calculated:

Dataset	Average error of X In C_corrected value	Average error of Y In C_corrected value	Average error of Z In C_corrected value
Pa2-debug-a	0.00	0.00	0.00

As we can see from the above, the C value after corrected has very small error with the C_expected value we calculated(ground truth value). This is consistent with our expectation. After using the correction function, the corrected measurement value should be same as the ground truth value. To further verify our correctio function is correct, we continue compare the G values in the next table.

Then, we tested it on the G value we read from the "pa2-debug-a-empivot.txt", "pa2-debug-a-optpivot.txt", "pa2-debug-a-em-fiducialss.txt" file and "pa2-debug-a-EM-nav.txt" in "pa2-debug-a" dataset. Since there is no EM distortion, noise or jiggle in "pa2-debug-a" dataset, the measured value should be the ground truth value. Thus, we expected G value after corrected should be very close to the measure value in "pa2-debug-a" dataset.

Table of the error in G values read from different files before and after correction in "pa2-debug-a" dataset:

Data file	Average error of X in G value	Average error of Y in G value	Average error of Z in G value
"pa2-debug-a-empivot.txt"	0.00	0.00	0.00
"pa2-debug-a-em-fiducialss.txt"	0.00	0.00	0.00
"pa2-debug-a-EM-nav.txt"	0.00	0.00	0.00

Since there is no EM distortion in the "pa2-debug-a" dataset, the measured value has not distortion. Thus, after we corrected the measured value, we should still have the same value. As we can see from the above table, the G value after we used distortion correction function is the same as the measured value in the dataset a, , which is consistent with our expectation and proved that our distortion correction function is correct in this step. Then we will further prove that correction function in the following validation.

Validation and Results

Monday, November 8, 2021 1:13 PM

2) Verify that the pivot calibration is correct with distortion correction

To verify that our pivot calibration with distortion correction function is correct, we compared the estimated post position with EM probe pivot calibration and estimated post position with optical probe pivot calibration we get from our code with the value in the output1 file from debug file a-f.

Table of the error in estimated post position with EM probe pivot calibration and estimated post position with optical probe pivot calibration from debug dataset a-f

Calibration for EM probe:

Dataset	$\vec{p}_{pivot}$ w/out correction	$\vec{p}_{pivot}$ after correction	Expected $\vec{p}_{pivot}$ from output1 file	Error Before Correction	Error After Correction
A	[207.23, 201.94, 208.5]	[207.23, 201.94, 208.5]	[207.23, 201.94, 208.50]	[0, 0, 0]	[0, 0, 0]
B	[201.45, 195.35, 200.44]	[201.12, 195.17, 200.18]	[201.43, 195.19, 200.36]	[0.02, 0.16, 0.08]	[0.31,0.02,0.18]
C	[206.33, 201.63, 207.67]	[207.37, 199.31, 207.69]	[207.38, 199.33, 207.69]	[1.05, 2.3, 0.02]	[0.01, 0.02, 0]
D	[192.41, 195.62, 199.46]	[192.41, 195.62, 199.46]	[192.41, 195.62, 199.46]	[0, 0, 0]	[0, 0, 0]
E	[199.08, 211.92, 200.62]	[190.2, 204.24, 196.63]	[190.81, 204.53, 196.66]	[8.27, 7.39, 3.96]	[0.61, 0.29, 0.03]
F	[193.21, 202.91, 205.08]	[195.16, 201.24, 205.78]	[195.07, 201.08, 205.81]	[1.86, 1.83, 0.07]	[0.09, 0.16, 0.03]

Discussion of the result:

As we can see from the above table, the error after we used the distortion correction function became very small. The error is small enough to prove that our correction function is correct and the pivot calibration with distortion correction is correct. The result from the table is consistent with our expectation. In dataset A, there is no EM distortion, noise and OT jiggle. Thus, the error are both zero before and after we applied the distortion correction function. In other dataset that has no EM distortion, such as Dataset B and D, the error remain almost the same because the error came from either EM noise or OT jiggle, which distortion correction function cannot reduce. Meanwhile in those dataset with EM distortion, such as dataset C, E and F, the error after applying the distortion correction function reduced and became much smaller.

Validation and Results

2021年11月8日 18:04

Calibration for Optical probe:

Dataset	$\vec{p}_{pivot}$ Returned by Program	Expected $\vec{p}_{pivot}$ from output1 file	Error
A	[401.24, 406.78, 198.28]	[401.24, 406.78, 198.28]	[0, 0, 0]
B	[405.78, 390.98, 209.44]	[405.78, 390.98, 209.43]	[0, 0, 0.01]
C	[395.46, 395.98, 198.17]	[395.46, 395.98, 198.16]	[0, 0, 0.01]
D	[401.94, 393.63, 198.89]	[401.94, 393.63, 198.88]	[0, 0, 0.01]
E	[405.62, 408.44, 208.42]	[405.63, 408.44, 208.42]	[0.01, 0, 0]
F	[407.49, 406.64, 196.43]	[407.49, 406.64, 196.43]	[0, 0.01, 0]

Discussion of the result:
As we can see from the above table, the error of $\vec{p}_{pivot}$ from optical proce pivot calibration between our result and the value from output1 file are very small. The error is small enough to prove that our optical calibration is correct.

3) **Verify that the tip location with respect to the CT image are correct**

To verify that the tip location with respect to the CT image are correct, we compared our result the tip location with respect to the CT image with the result in the output2 file from debug dataset a-f.

Table of the error in the tip location with respect to the CT image from debug dataset a-f compared with the tip location value from output2 file

Dataset	Error of tip location with respect to the CT image **before calibration**	Error of tip location with respect to the CT image ***after calibration***
A	[0.00, 0.00, 0.00]	[0.00, 0.00, 0.00]
B	[0.01, 0.02, 0.01]	[0.07, 0.01, 0.1]
C	[0.38, 0.08, 0.06]	[0.01, 0.01, 0.00]
D	[0.00, 0.00, 0.00]	[0.00, 0.00, 0.00]
E	[0.52, 2.6, 0.21]	[0.19, 0.13, 0.05]
F	[0.64, 0.89, 0.02]	[0.12, 0.00, 0. 00]

Discussion of the result:
As we can see from the above table, the error of the tip location with respect to the CT image after we used the distortion correction function became very small. The error is small enough to prove that the tip location with respect to the CT image we get are correct. The result from the table is consistent with our expectation. Since in dataset A, there is no EM distortion, noise and OT jiggle, the error are both zero before and after we applied the distortion correction function. In other dataset that has no EM distortion, such as Dataset B and D, the error remain almost the same because the error came from either EM noise or OT jiggle, which distortion correction function cannot reduce. In those dataset with EM distortion, such as dataset C, E and F, the error after applying the distortion correction function reduced and became much smaller.

A tabular summary of the results obtained for unknown data

Friday, November 5, 2021 9:21 PM

Calibration for EM probe:

Dataset	$\vec{p}_{pivot}$
g	[204.86, 205.44, 190.77]
h	[193.54, 209.42, 200.59]
i	[195.79, 209.91, 192.35]
j	[192.11, 196.0 , 192.89]

Calibration for OP probe:

Dataset	$\vec{p}_{pivot}$
g	[403.42, 407.11, 205.19]
h	[395.38, 408.77, 206.8]
i	[391.36, 406.36, 206.96]
j	[403.99, 403.74, 199.26]

Discussion of Result:
According to our validation result with Debug data sets and the verification shown in both PA1 report and this PA2 report, we expect the $\vec{p}_{pivot}$ outcome of the program using unknown data sets to be correct. And even if the unknown data has EM distortion, as we applied distortion correction function, our outcome should successfully reduce the error from EM distortion.

Tip location with respect to the CT image:

Dataset	$\vec{t}_{tip}, CT\ frame$
g	[111.82,75.78,148.56] [85.57,143.12,75.26] [76.6 ,47.65, 92.45] [65.84, 80.8 ,26.38]
h	[116.61, 63.62, 146.89] [42.1 , 164.79, 72.84] [66.4 , 107.55, 61.56] [41.34, 140.77, 26.81]
i	[63.72, 91.56, 109.05] [106.41, 165.7, 42.28] [73.82, 145.8 ,122.51] [50.08, 87.74,88.82]
j	[143.09, 113.91, 59.88] [96.99, 34.02,114.61] [133.89, 41.2,109.1] [126.28, 46.91,117.82]

Discussion of Result:
According to our validation result of tip location with respect to the CT image with Debug datasets and the verification shown in previous sections, we expect the tip location with respect to the CT image outcome we get from the program using unknown data sets to be correct. And even if the unknown data has EM distortion, as we applied distortion correction function, our outcome should successfully reduced the error from EM distortion.
Convergence:
All numerical calculations involved in Programming Assignment 2 with given dataset were able to converge within reasonable time(< 10 seconds).