

# Motion Analyst 2D Version 1.5 Help Documentation

Rev 1.5

## Table of Contents

<a href="#">1 Revision History.....</a>	<a href="#">1</a>
<a href="#">2 Purpose.....</a>	<a href="#">1</a>
<a href="#">3 Using the Program.....</a>	<a href="#">2</a>
<a href="#">3.1 Getting Started.....</a>	<a href="#">2</a>
<a href="#">3.2 Image Control.....</a>	<a href="#">3</a>
<a href="#">3.3 Marker Locations.....</a>	<a href="#">4</a>
<a href="#">3.4 Tracking Configuration.....</a>	<a href="#">6</a>
<a href="#">3.5 Program Control.....</a>	<a href="#">7</a>
<a href="#">4 Files.....</a>	<a href="#">10</a>
<a href="#">4.1 Control File.....</a>	<a href="#">10</a>
<a href="#">4.2 Results File.....</a>	<a href="#">10</a>
<a href="#">4.3 Results 2D File.....</a>	<a href="#">11</a>
<a href="#">5 Future Work.....</a>	<a href="#">12</a>
<a href="#">5.1 Known Bugs.....</a>	<a href="#">12</a>
<a href="#">5.2 Limitations.....</a>	<a href="#">13</a>

## 1 Revision History

Revision	Date	Name	Comment
01	2-9-10	R. Colbrunn	Created
02	2-15-10	R. Colbrunn	Created Version 1.2. Updated section 5
03	3-24-11	R. Colbrunn	Created Version 1.3. Added 3 features; Select Assist, Pan and Zoom, and Auto-increment selecting. Section 3.3 and 3.4. Updated section 5.
04	11-17-11	N. Mealey R. Colbrunn	Created Version 1.4. Improved robustness of tc_multi_points.m software. See revision history in file for details.
1.5	5-13-12	R. Colbrunn	Created Version 1.5. Modified button layout and expanded the toolbar. Added image enhancement features of brightness, contrast, and edge detection. Improved robustness of tc_multi_points.m software with sub-pixel motion detection algorithm to prevent erroneous creep. See revision history in file for details. Made cross platform compatible (mac, unix). Fixed new bug with Matlab 2011 when opening files with 'versn' parameter. Updated all figures in document with significant changes to section 3.4.

## 2 Purpose

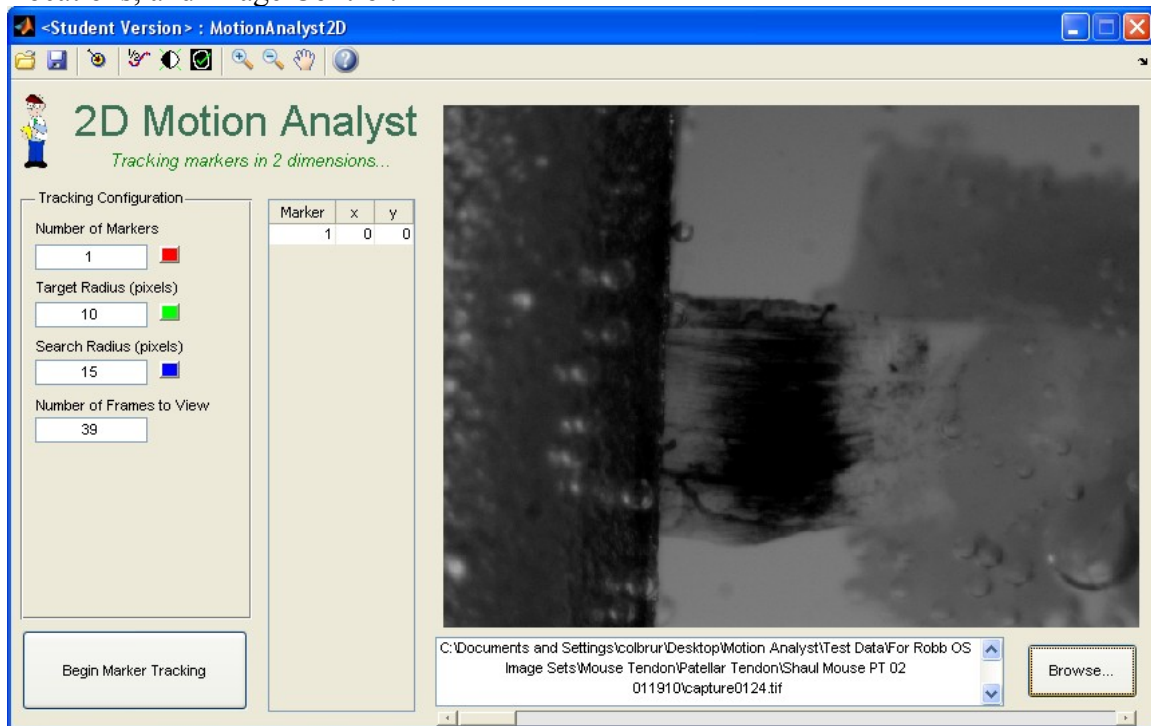
This document outlines how to use the Motion Analyst 2D software package. Motion Analyst 2D is a software package to provide a more user friendly GUI and automated functionality to employ the 2D texture correlation capabilities of the tc\_multi\_points.m software written by Michael Bey which can track the movement of markers in 2D images. The tc\_multi\_points.m software has more functionality than Motion Analyst 2D is able to capture, such as batch processing, but it is a useful visual interface for 2D tracking.

### 3 Using the Program

#### 3.1 Getting Started

This software is used to track motion of markers in video images. The images must be saved as a series of .tif or .jpg files. The filenames must be the same length with the last characters being numerically increasing values that correspond to frame numbers of the video. The tracking is done in grayscale so all images should be converted to grayscale prior to tracking. All files should be placed in a folder unique to those images only.

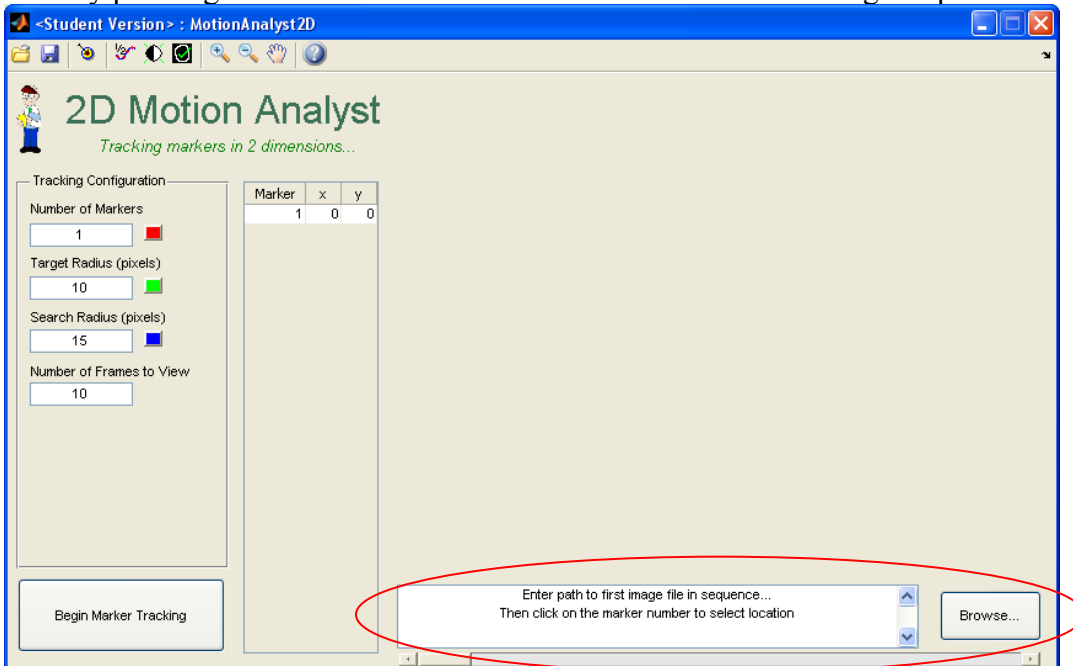
The program has 4 main areas: Tracking Configuration, Program Control, Marker Locations, and Image Control.



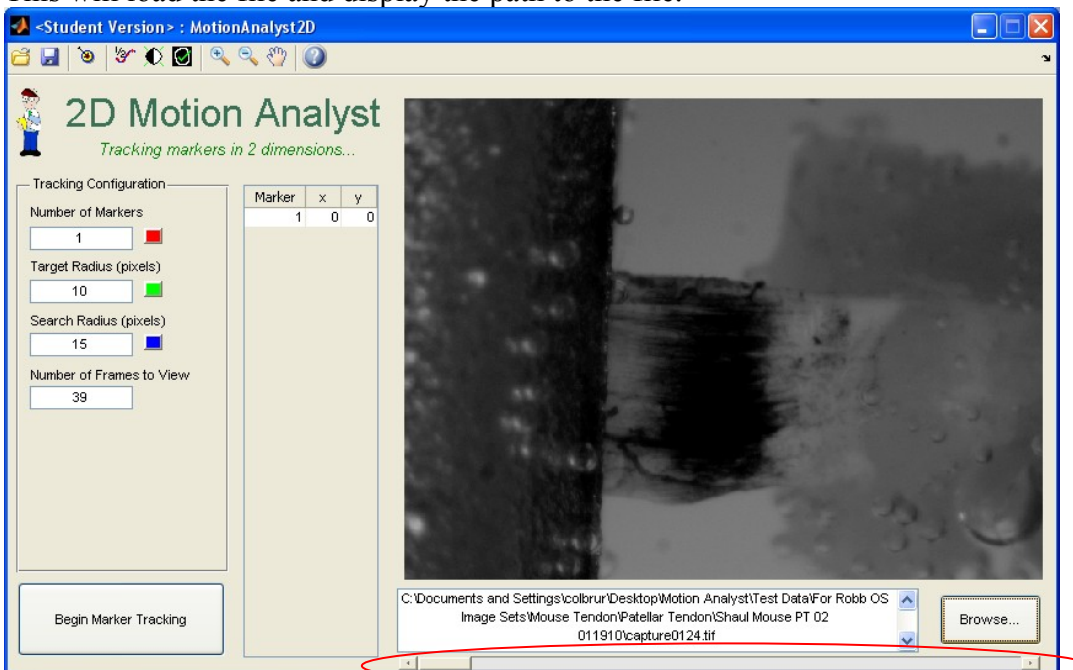
The program can be used to select the first image in a series and define the marker locations for that image. All the tracking parameters can be defined and then the Begin Marker Tracking button will initiate the tc\_multi\_points.m software. This software does not actually track markers, but more correctly it tracks pixel value patterns from one image to the next. It looks for a set of pixels of Target Radius size and tries to find where within the Search Radius size it shows up in the subsequent image. Markers can be any pattern of pixels that provides as high of a contrast as possible.

### 3.2 Image Control

Start by pressing the browse button to select the first file in the image sequence.



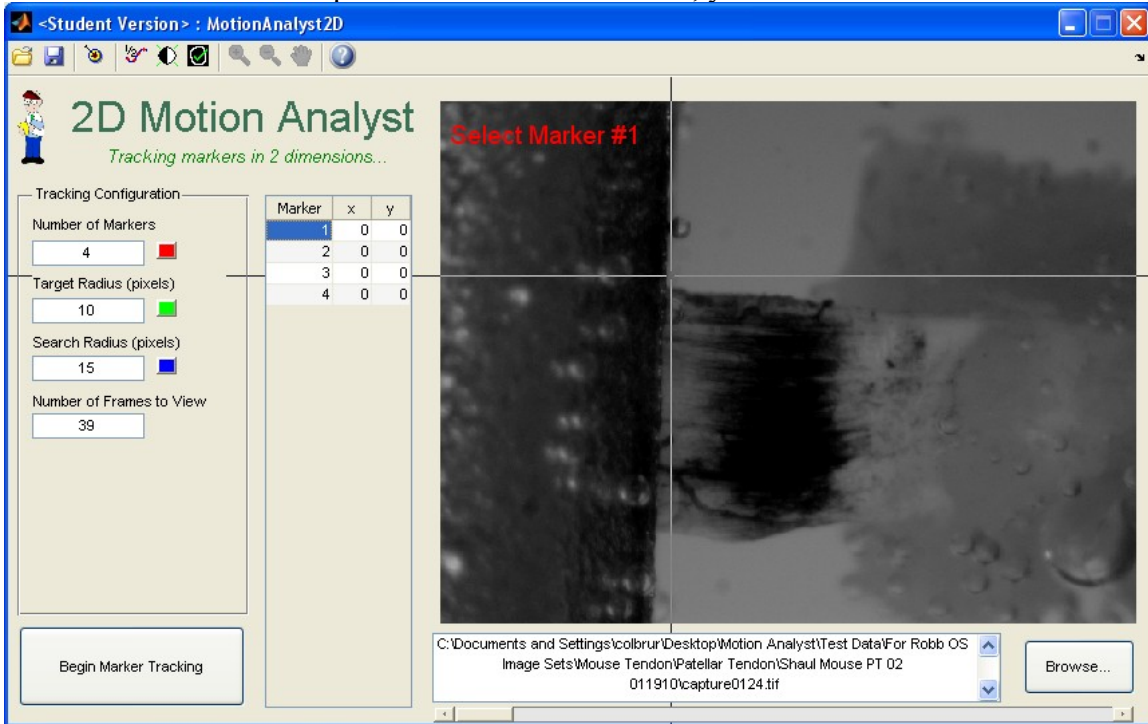
This will load the file and display the path to the file.



The slider bar can then be used to scroll through all the images in the sequence to verify that all points of interest are included in all frames. As you change the slider bar the path will update to show which frame is being viewed at the time.

### 3.3 Marker Locations

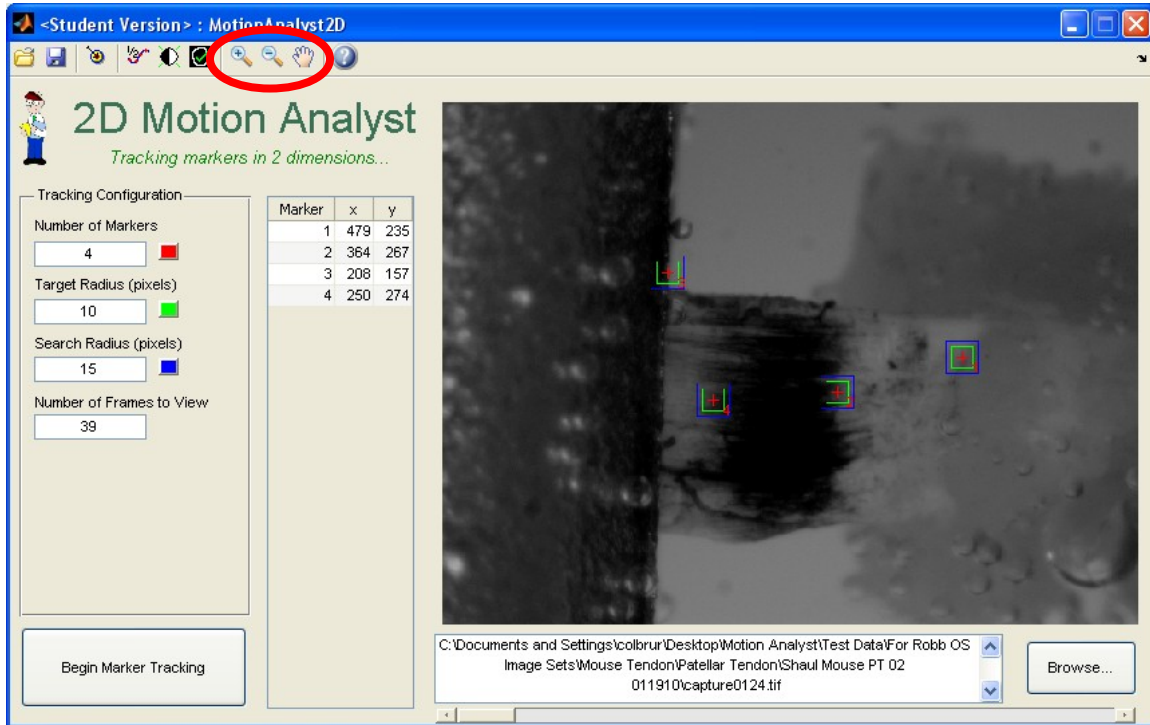
Once the image is loaded, type in the number of markers to track in the Tracking Configuration area. This will expand the table to show the x, y coordinates for all the markers.



Click on the Marker number in the table and a prompt, along with cross-hairs will appear to direct you to click on the image where that particular marker will be located. The x, y pixel values for this marker will show up in the table. Repeat for all markers. The software will auto-increment through each marker on the list above the marker selected as the starting point. To interrupt this if only one, or a few markers are desired to be reselected, the user can 1) click off the image, 2) right click, or 3) press the ESC key.

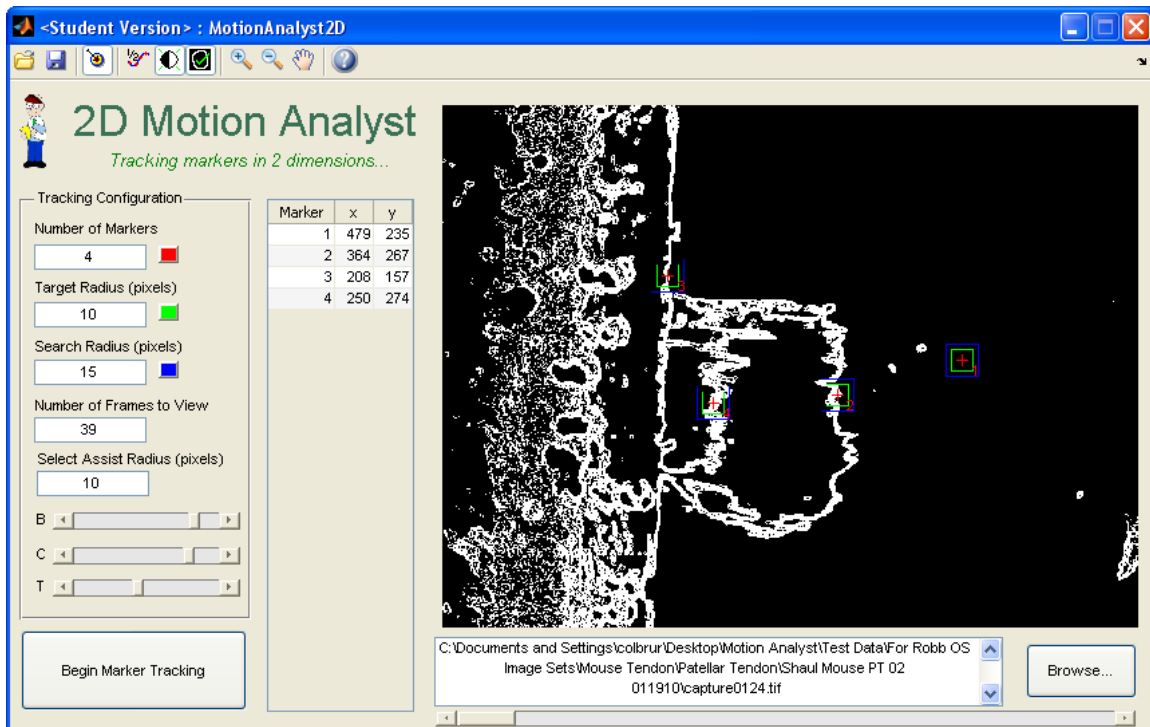
Zoom and Pan controls in the upper left hand corner can be used to zoom into the image to verify marker selection was as desired.

Make sure to pick markers where not only will they not leave the image area, but also they need to be far enough away from the edge that the target and search area for each marker does not leave the image area. This will cause an error in the texture correlation software.



The zoom and pan features on the toolbar can be enabled to investigate marker placement with more precision. Trying to replace a marker will cause the view to reset to the first frame and fully zoomed out. Fine adjustment of the marker placement can be done by manually entering new values in the marker table.

### 3.4 Tracking Configuration



The Tracking Configuration section of the screen has several parameters. There are color selectors to allow modification of the colors so that they can be as visible as possible.

*Number of Markers* – Number of markers the program is expecting to track.

*Target Radius* – This is the size of the region that the tracked marker (or pattern) is located. Note that this software does not actually track markers, but more correctly it tracks pixel value patterns from one image to the next. It looks for a set of pixels of Target Radius size and tries to find where within the Search Radius size it shows up in the subsequent image. Markers can be any pattern of pixels that provides as high of a contrast as possible. The term radius may be a bit of a misnomer as the region is actually a square where the length of a side =  $(2 \times \text{Radius} + 1)$  pixels.

*Search Radius* – Similar in function to the Target Radius except it defines the image area where the pixel pattern from the target area from the previous image will be searched for. It is also a square where the length of a side =  $(2 \times \text{Radius} + 1)$  pixels.

*Number of Frames to View* – When running the tracking software the images from the sequence will be displayed in a new plot and they will display a white dot where the marker center has been calculated to be. The number of frames to view will close this new plot after the specified number of frames. Keep in mind that the tracking will occur on all frames in the folder regardless of this parameter. It is only a display parameter. It will auto-populate to match the number of images in the sequence folder. If there is a long se-

quence and there is no desire to view all the images then reducing this value will close the plot early.

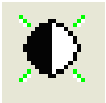
The toolbar also allows for other features of the tracking to enabled and disabled. Parameters of these features will only be visible when they are enabled.



*Select Assist* – The Select Assist option can be used to improve detection of marker centroids when markers can be encompassed by target area. Select Assist will use the *Radius* value to define how big of an area to look for the marker in. Select assist is designed to detect the centroid of dark markers against the contrast of a lighter background. Select assist will return undesirable results if trying to detect lighter markers against a dark background. Do not use when using edges as markers. It is off by default.



*Interpolate* – Interpolate is used to improve the tracking capabilities of the texture correlation software. It doubles the resolution of the image and also effectively doubles the target and search radii. This can be used when the standard tracking does not seem to be working. Note that this will also quadruple processing time.



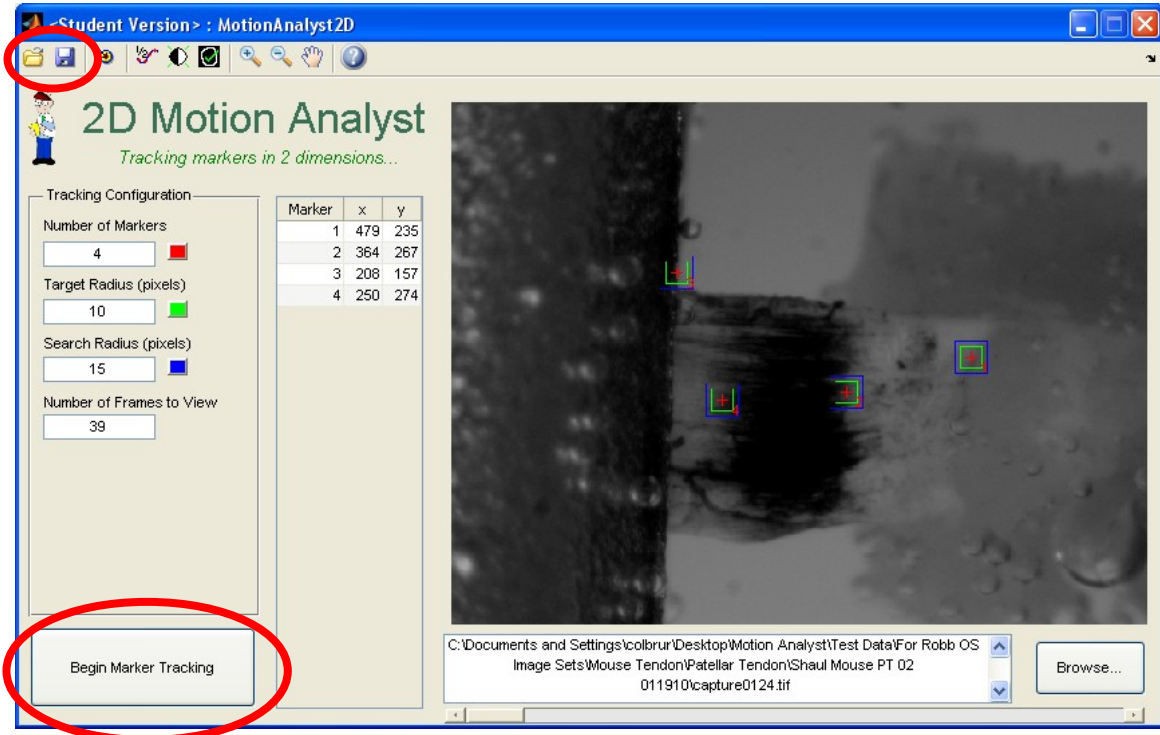
*Brightness/Contrast Adjustment* – By adjusting the brightness and contrast of the image, the tracking algorithm may better differentiate the marker movement. The ‘B’ and ‘C’ slider bars can be adjusted to modify these parameters. These settings will be stored in the control file and applied to each image during processing. Note that this will slightly increase processing time.



*Edge Detection* – By adding edge detection a high degree of contrast can be created in the image. In some circumstances this can produce results that improve motion tracking. The ‘T’ slider bar can be adjusted to modify the threshold parameter. This setting will be stored in the control file and applied to each image during processing. Note that this will slightly increase processing time. The image processing chain is brightness/contrast and then edge detection. In this way, edge detection can also be refined with modifications to the brightness and contrast parameters.

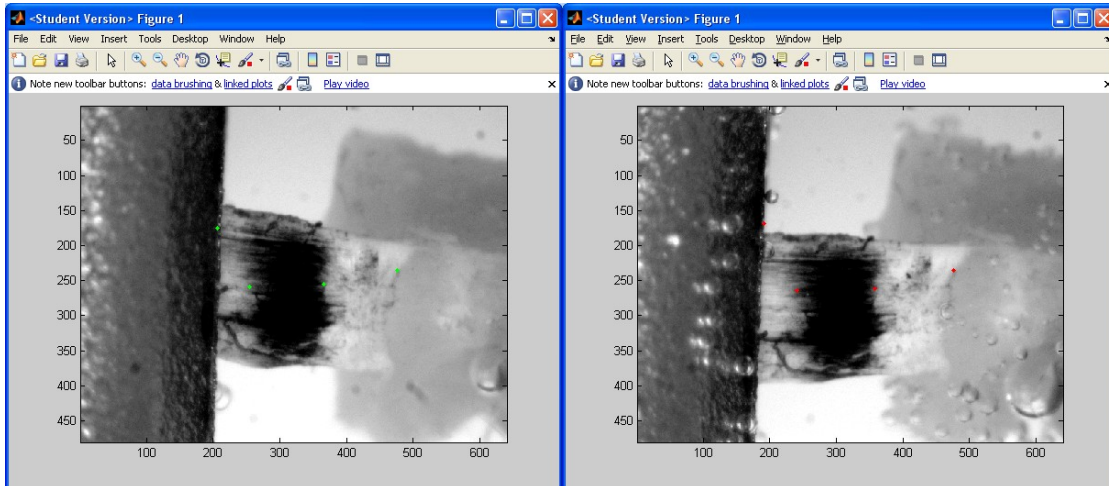
### 3.5 Program Control

The Program Control section consists of 3 buttons; Load, Save, and Begin Marker Tracking.



The parameters for a given test are stored in what is referred to as a control file. This control file is a .csv file of a specific format that the tc\_multi\_points.m software reads in and uses to track the markers. By clicking on the Save button a control file is created in this format using whatever filename the user chooses. By clicking on the Load button, a control file can be selected and loaded. By clicking on Begin Marker tracking a control file (named 'control file.csv') is automatically created with the existing parameters and it is placed in the folder where all the images are located. The user can choose to either use this file, or select a previously created file with different parameters. The tracking will then begin and a window will pop up and display the tracking. The markers are shown as green or red dots in the image and this allows the user to visually confirm how well it is tracking. The green dots correspond to all markers having moved from their previous position. The red dots correspond to a frame when at least one of the markers did not move and the sub-pixel motion algorithm is enabled.





Progress of the tracking (along with any errors) will be displayed in the Matlab workspace so be sure to check there for feedback. Once the tracking is complete, a results file will be created in both .csv and .mat format. If the same sequence is tracked multiple times the results.csv will be appended with the new data. The .mat file will be overwritten each time. The .mat file will be named Results2D.mat and it is a Matlab proprietary binary format with all the data organized in a way that future manipulation in the Motion Analyst 3D can be easily completed.

## 4 Files

### 4.1 Control File

The control file is a comma separated value (.csv) file that, when imported into excel, looks like the following.

	A	B	C	D	E	F	G	H	I	J
1	Directory	C:\Data\Patellar Tendon\Shaul Mouse PT 02 011910								
2	ImageType	tif								
3	InitialXYPoints		479	235	364	267	208	157	250	274
4	TargetWidth	10								
5	SearchWidth	15								
6	Interpolate	0								
7	NumFramesToDisplay	39								
8	HistogramAdjust	1								
9	Brightness	0.75859								
10	Contrast	1.0606								
11	EdgeDetect	1								
12	EdgeThreshold	0.05074								

*Advanced Feature Note:* If the batch processing functionality of the tc\_multi\_points.m software is to be used, individual control files can be created using Motion Analyst 2D and saved. Then the control files can be manually merged in any spreadsheet program and then saved as a combined .csv file with a unique name.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Directory	R:\BME\WRMTC\Projects\RB-0086-01 Shoulder Repair Strain - 3D Camera\Motion Analyst\Test Data\Mouse Tendon\Patellar Tendon\Mouse PT 02 011910													
2	ImageType	tif													
3	InitialXYPoints		498	213	364	278									
4	TargetWidth	10													
5	SearchWidth	15													
6	Interpolate	0													
7	NumFramesToDisplay	39													
8	Directory	R:\BME\WRMTC\Projects\RB-0086-01 Shoulder Repair Strain - 3D Camera\Motion Analyst\Calibration\3D Calibration\Verification Data\cam1													
9	ImageType	TIF													
10	InitialXYPoints		246	182	544	176	547	275	447	372	126	340	109	480	
11	TargetWidth	20													
12	SearchWidth	40													
13	Interpolate	0													
14	NumFramesToDisplay	5													

To run the batch program, instead of selecting the default control file after pressing the Begin Marker Tracking button, select the new uniquely named combined file and the program will proceed as defined.

### 4.2 Results File

The results file is a comma separated value (.csv) file that, when imported into excel, looks like the following.

	A	B	C	D	E	F	G	H	
1	1/27/2010 14:43								
2	x1	y1	x2	y2	x3	y3	x4	y4	
3		246	182	544	176	547	275	447	372
4		246	162	543	156	547	251	448	351
5		245	136	541	130	546	217	448	318
6		244	117	539	111	544	190	447	289
7		244	129	534	124	539	174	445	261
8									

### 4.3 Results 2D File

The results 2D file (always named Results2D.mat and placed in the same directory as the image sequence) is in a Matlab proprietary binary format with all the data organized in a way that future manipulation in the Motion Analyst 3D can be easily completed. The two values contained in this file are ‘Res2DFileHeader’ and ‘xypoints’.

Here is an example of the Res2DFileHeader cell array.

Res2DFileHeader <5x2 cell>		
	1	2
1	'Date/Time'	'09-Feb-2010 15:51:39'
2	'Number of Markers'	4
3	'Number of Frames'	39
4	'Data format'	'x1 y1 x2 y2...'
5	'Data units'	'pixels'

Here is an example of the xypoints 2D array.

xypoints <39x8 double>									
	1	2	3	4	5	6	7	8	9
1	498	213	364	278	210	152	246	281	
2	498	213	363	278	207	156	244	280	
3	498	213	362	278	203	161	242	278	
4	498	213	361	277	200	164	240	276	
5	498	213	359	275	195	166	237	273	
6	498	212	358	274	193	166	236	271	
7	498	212	357	273	192	166	235	270	
8	498	212	357	273	192	166	235	270	
9	498	212	357	273	192	166	235	270	
10	497	212	357	273	192	166	235	270	
11	497	212	357	273	192	166	235	270	
12	497	212	357	273	192	166	235	270	
13	497	212	357	273	192	166	235	270	
14	497	212	357	273	192	166	235	270	
15	497	212	367	258	207	175	250	264	
16	497	212	367	258	206	178	250	266	

## 5 Future Work

### 5.1 Known Bugs

Error handling is limited and can be enhanced to improve functionality. For example, the control files selected and loaded must be in the expected format to work.

Sometimes when the number of markers is being modified the marker list will not update (change size) as expected. The work around is to set the number of markers to 0 or 1 and hit enter until it reduces to a single marker. Then the number can be increased to the desired value.

All efforts were made to be able to handle images of many sizes and formats. However, not all possibilities were accounted for and sometimes if the image does not match an expected way to handle it the plots will display as a white graph with labeled axes. The work around is to modify the image format or size to achieve the correct behavior.

Motion Analyst 2D can define marker locations for multiple image formats and sizes, but it may have trouble tracking color images. The work around is to convert the images to grayscale in a 3<sup>rd</sup> party program (e.g. Irfanview or Photoshop).

Marker locations cannot be tracked and the program errors out if the search or target region extends beyond the image area. There is no work around other than to make certain the markers are sufficiently away from the edge of the image. In the future, the program should be modified to dynamically change the search and target region to stop at the edge rather than try to look beyond it.

Select assist is designed to detect the centroid of dark markers against the contrast of a lighter background. Select assist will return undesirable results if trying to detect lighter markers against a dark background. Future versions should handle this distinction automatically.

The user can break out of auto-increment of marker selection by right clicking, clicking off the image, or pressing ESC key. However, a Matlab bug prevents the ESC key from working before the first marker is selected. The work around is to use one of the other two options.

This is not necessarily a bug, but a reality when working with a Matlab GUI is that the current directory of the Matlab session needs to be directed to where the Motion Analyst files are located. If they are not, then when the user tries to click on any button the program will not respond until the current directory is returned to the correct location.

## **5.2 Limitations**

Motion Analyst 2D can define marker locations for multiple image formats and sizes, but is limited to tracking only .jpg or.tif images.

Motion Analyst has not been tested on all versions of Matlab Software. It must be run on Matlab revision 2008b or higher. It also requires the Image Processing toolbox.