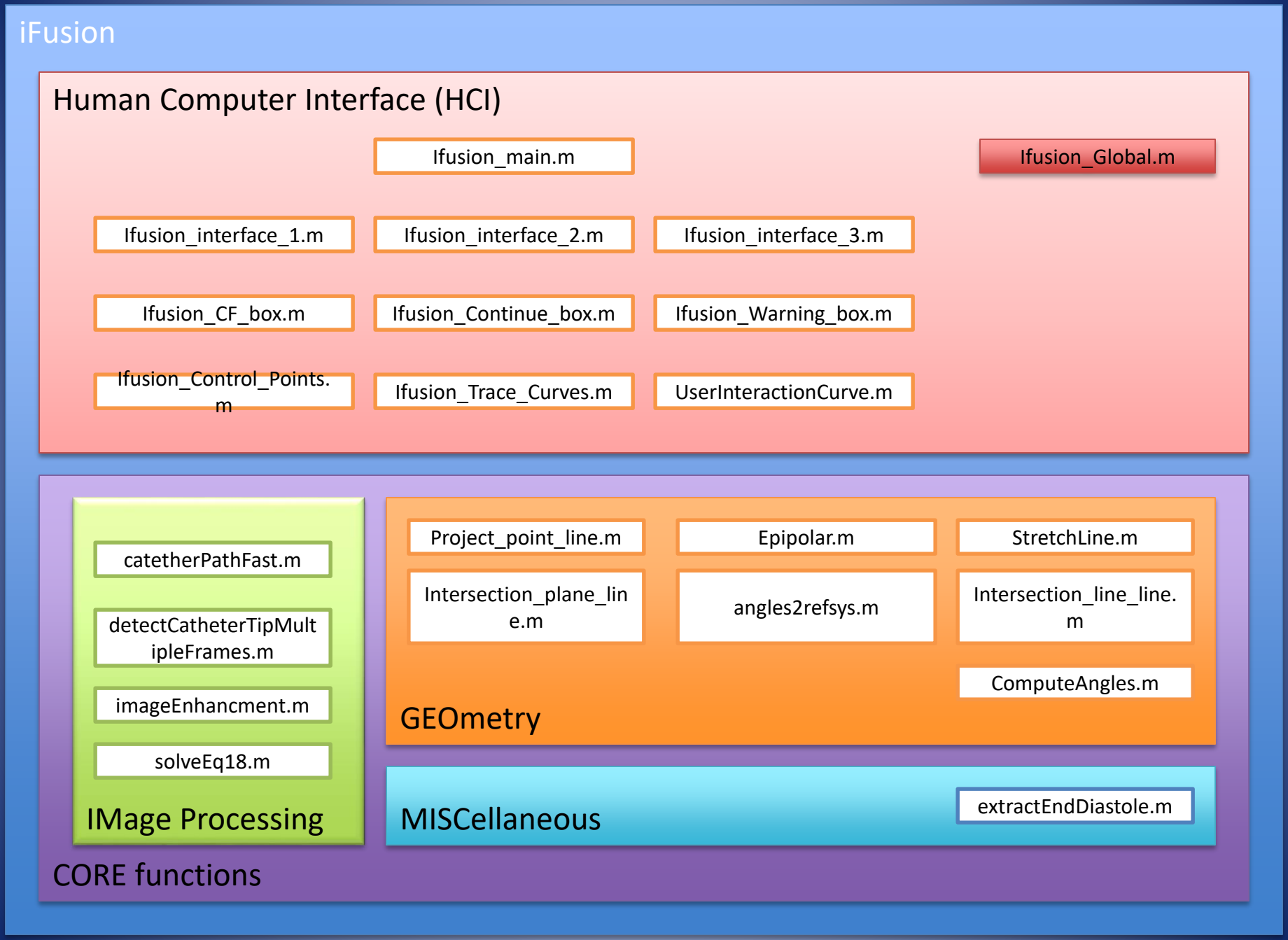
iFusion

iFusion

- MATLAB prototype.
- Two layers:
  - INTERFACE: Dedicated to Human Computer Interaction routines (11 functions)
  - CORE: Dedicated to the REAL part of the iFusion project (12 functions)
    - IMage Processing
    - GEOmetry
    - MISCellaneous

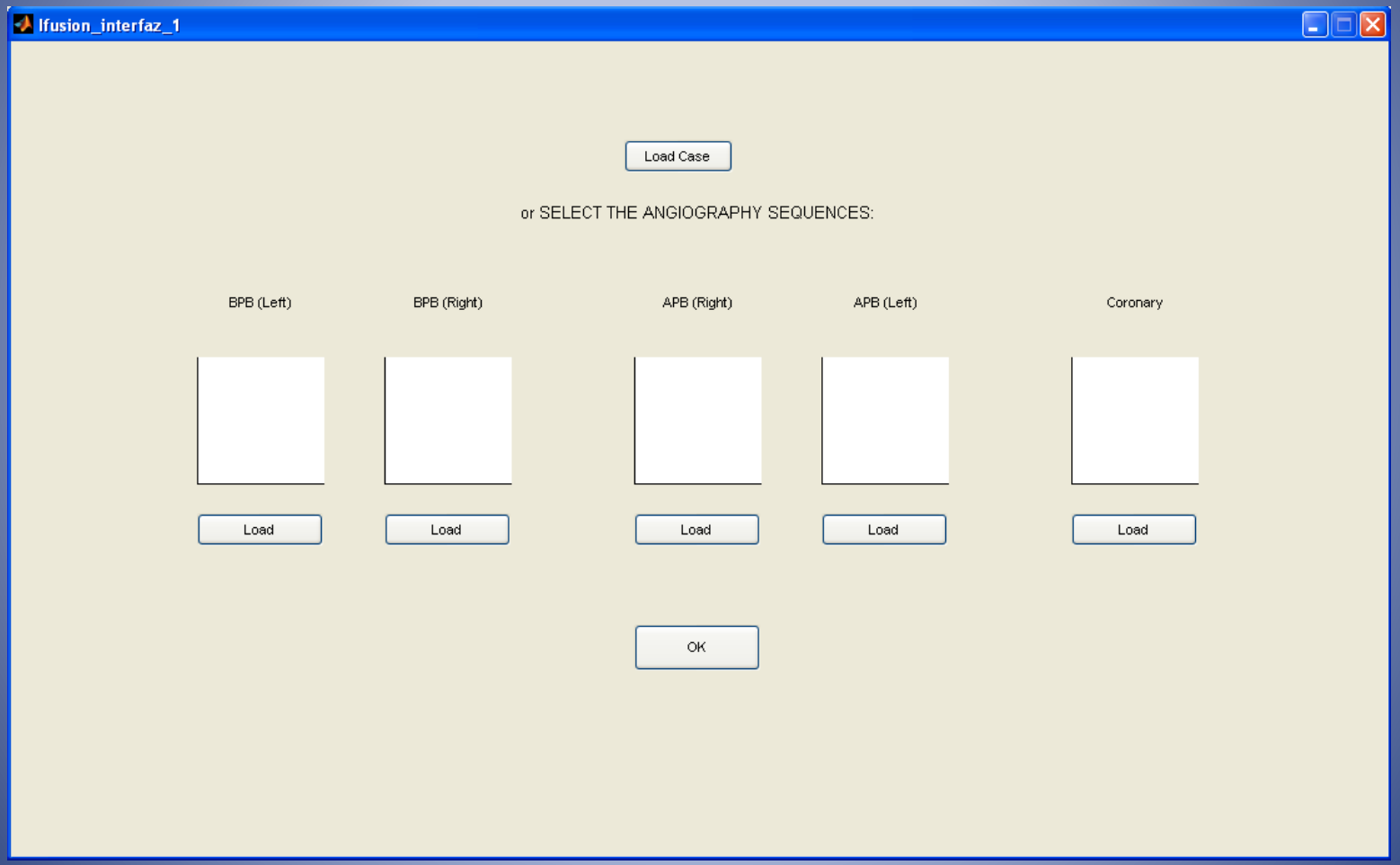

iFusion
Human Computer Interface (HCI)
Ifusion_main.m
Ifusion_Global.m
Ifusion_interface_1.m
Ifusion_interface_2.m
Ifusion_interface_3.m
Ifusion_CF_box.m
Ifusion_Continue_box.m
Ifusion_Warning_box.m
Ifusion_Control_Points.m
Ifusion_Trace_Curves.m
UserInteractionCurve.m
catetherPathFast.m
detectCatheterTipMultipleFrames.m
imageEnhancment.m
solveEq18.m
IMage Processing
Project_point_line.m
Epipolar.m
StretchLine.m
Intersection_plane_line.m
angles2refsys.m
Intersection_line_line.m
ComputeAngles.m
GEOmetry
MISCellaneous
extractEndDiastole.m
CORE functions

iFusion: INTERFACE

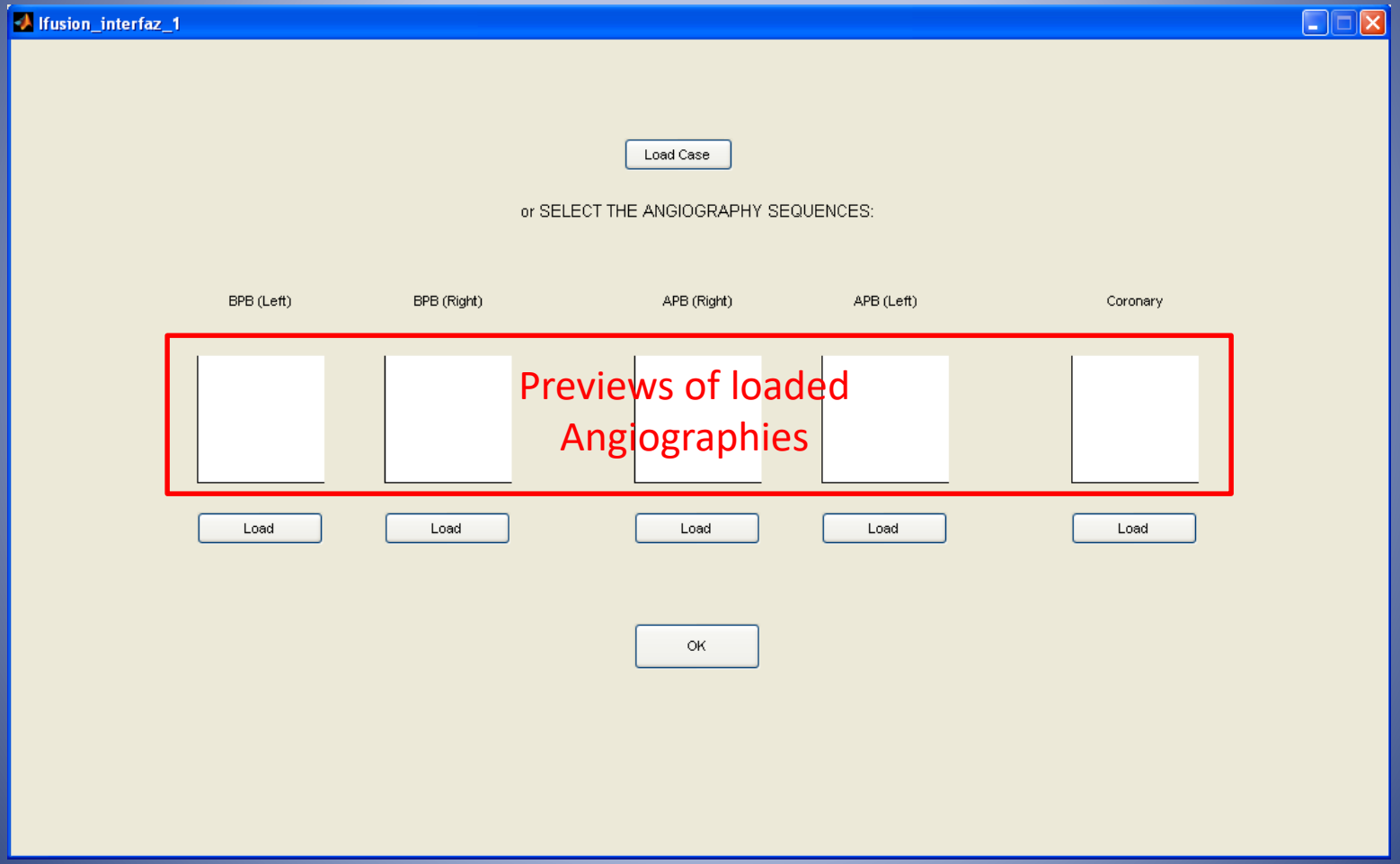
iFusion: INTERFACE

- Similar to the one of the first C++ prototype.
- Actual visualization (buttons, menus, etc.) quite differs from the previous version.
- NO functions or methods regarding the iFusion project.

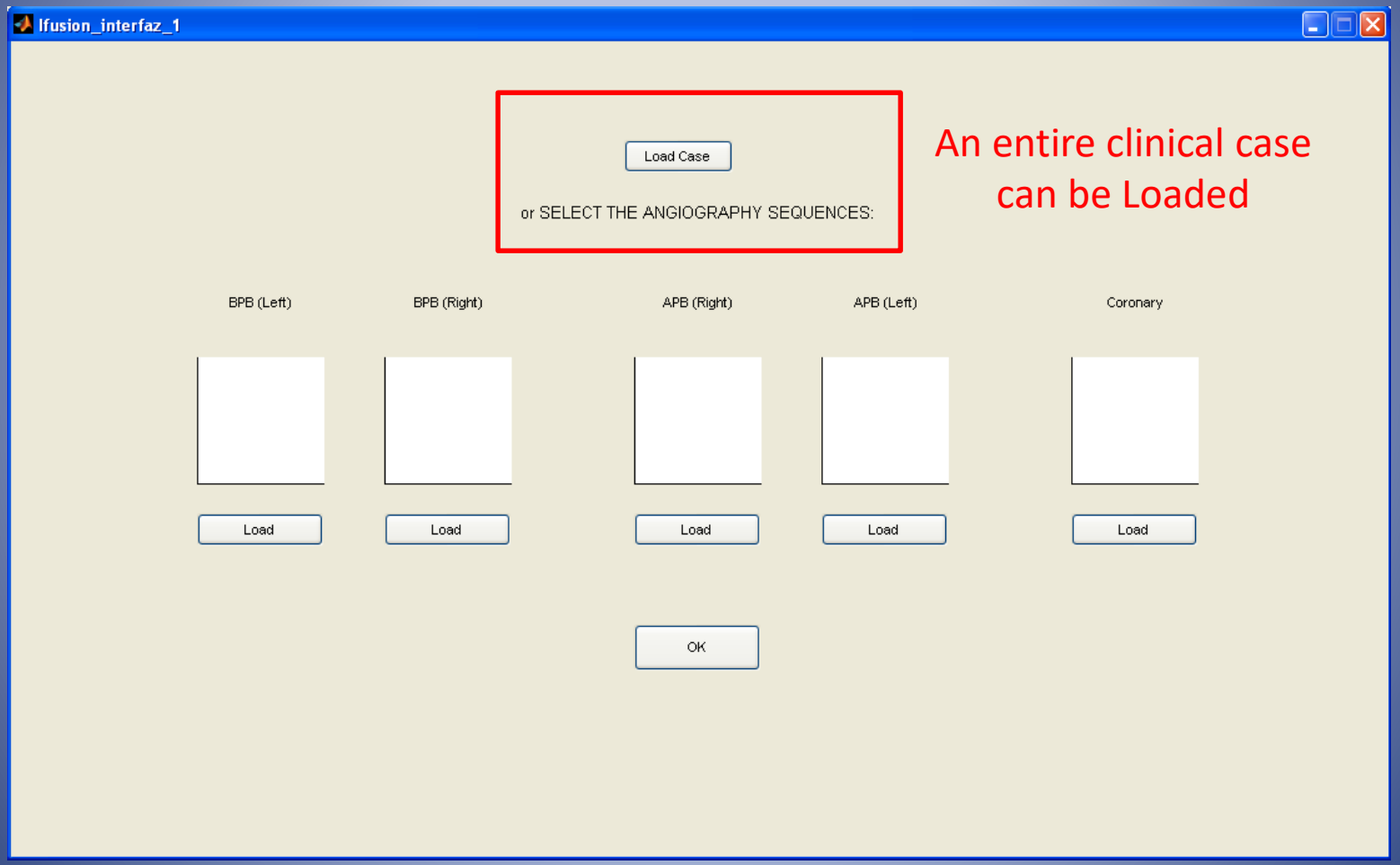
iFusion: INTERFACE_1
Loading Angiographies



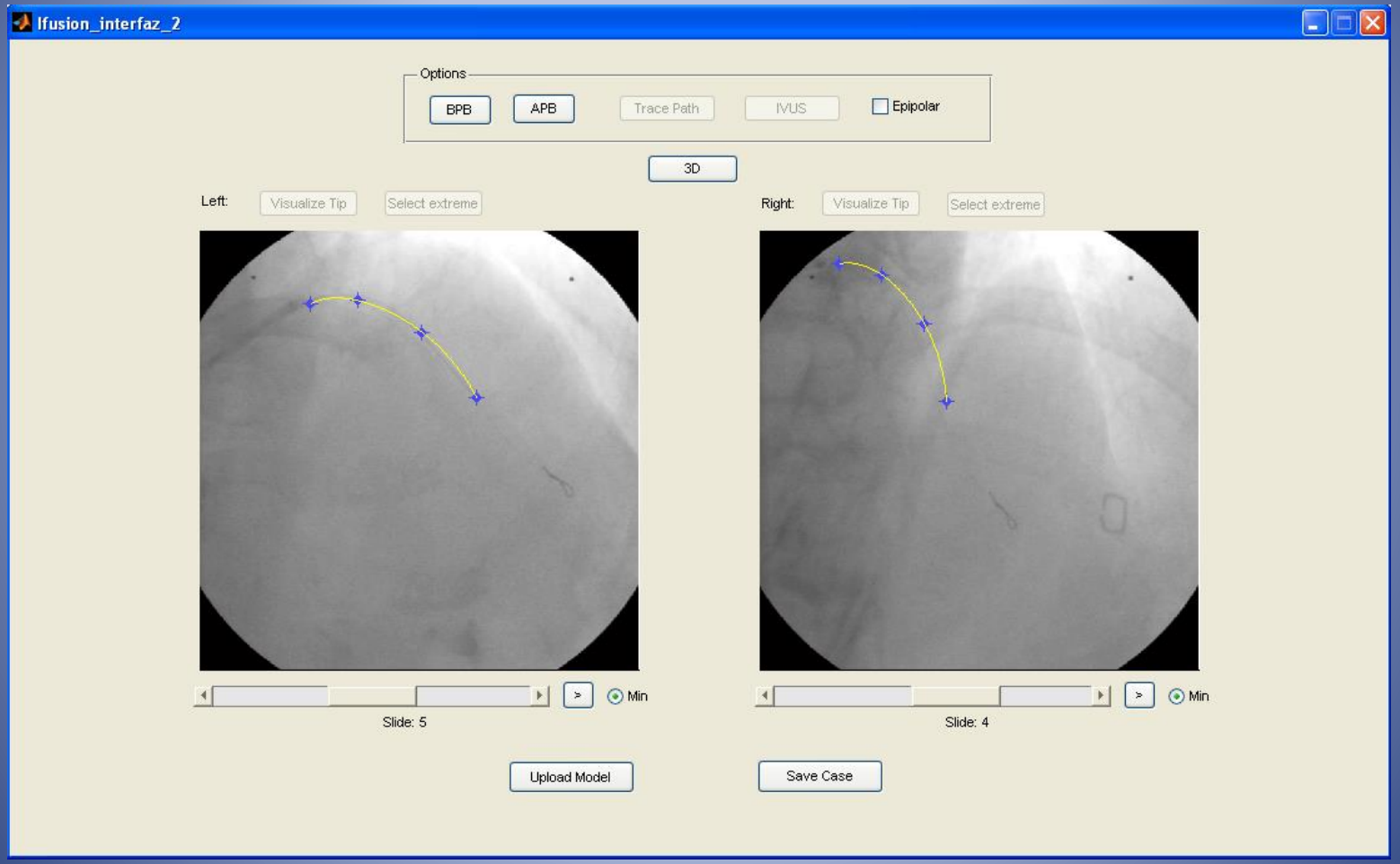
iFusion: INTERFACE_1
Loading Angiographies



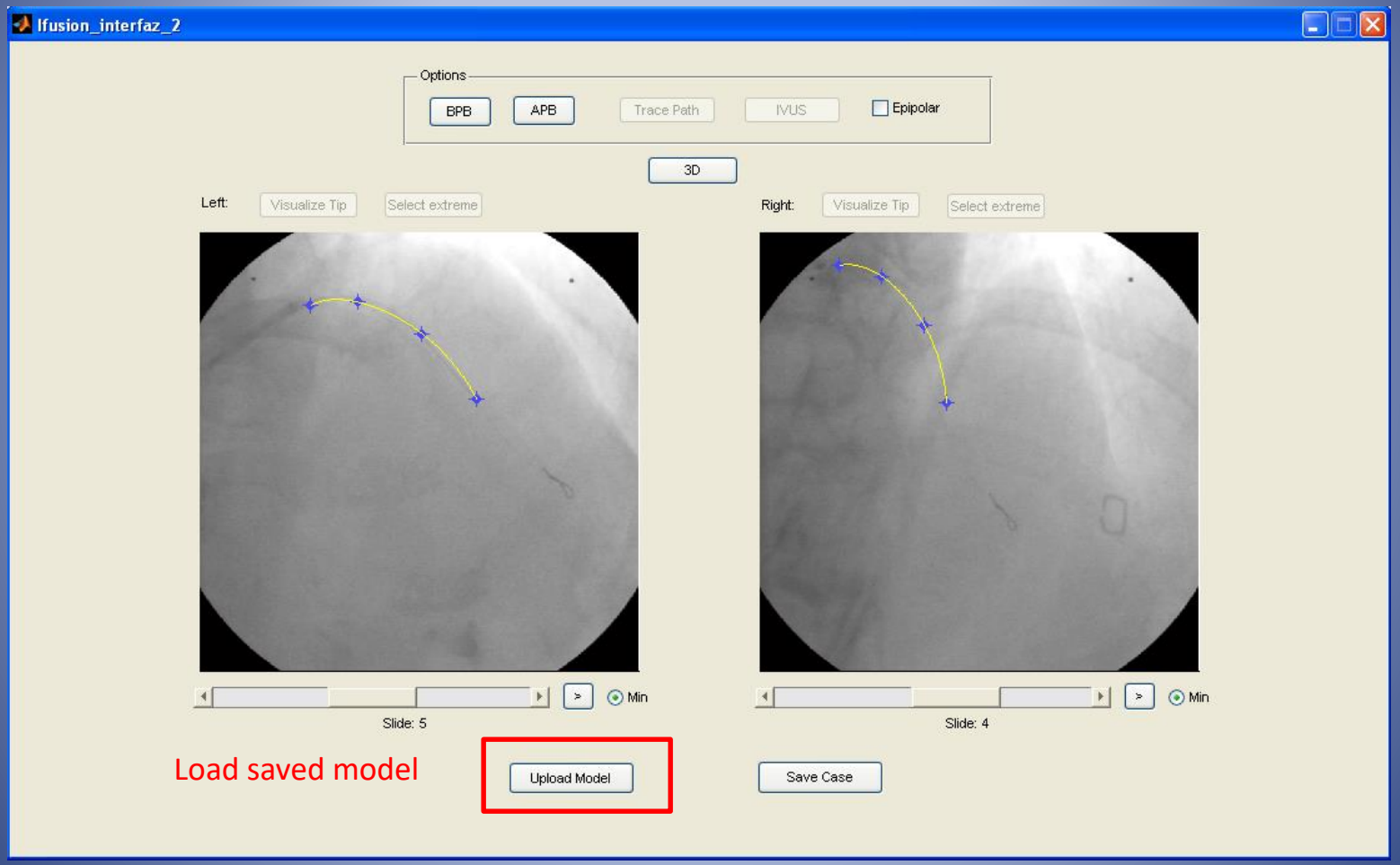
iFusion: INTERFACE_1
Loading Angiographies



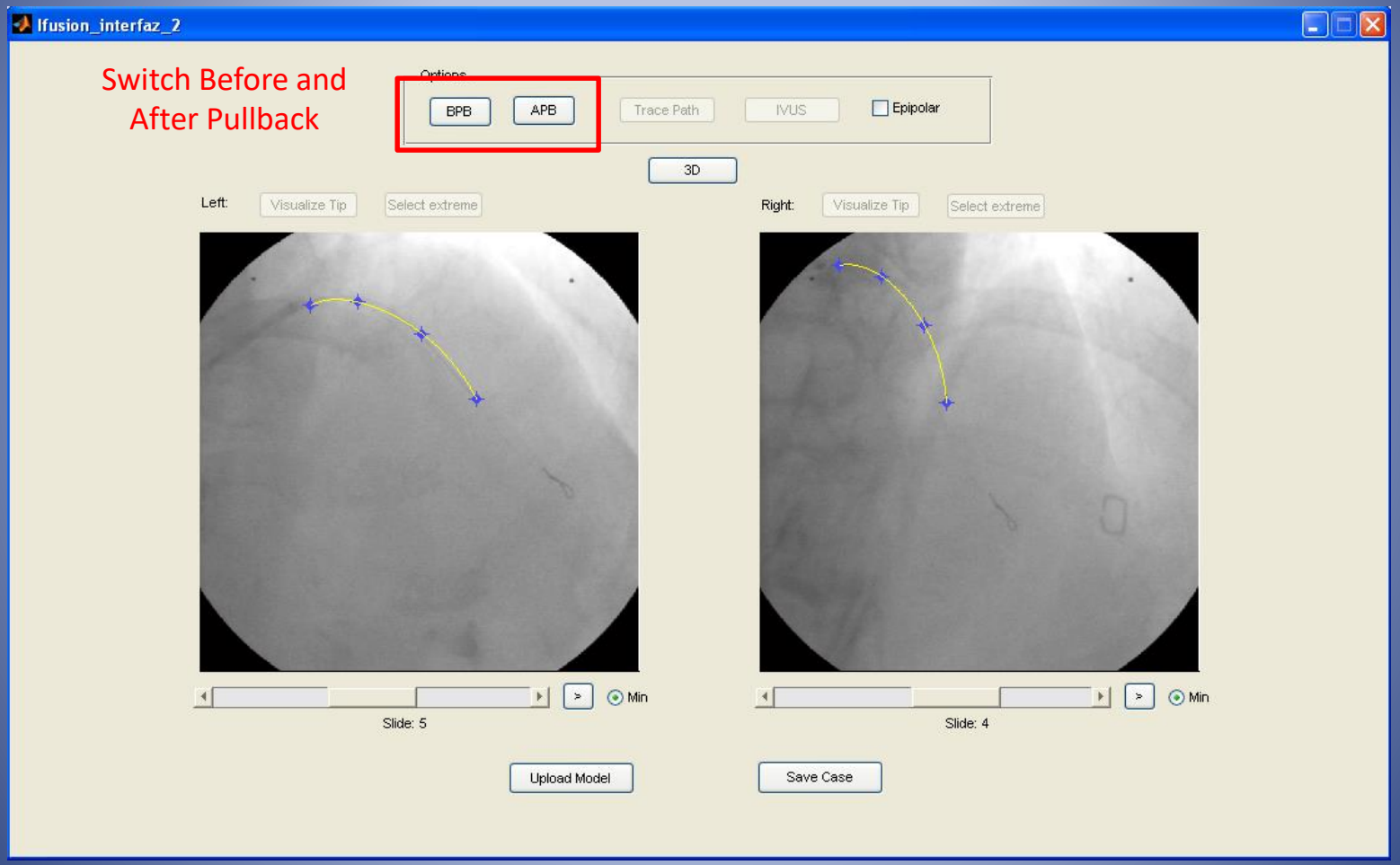
iFusion: INTERFACE_2
Curve Model Generation



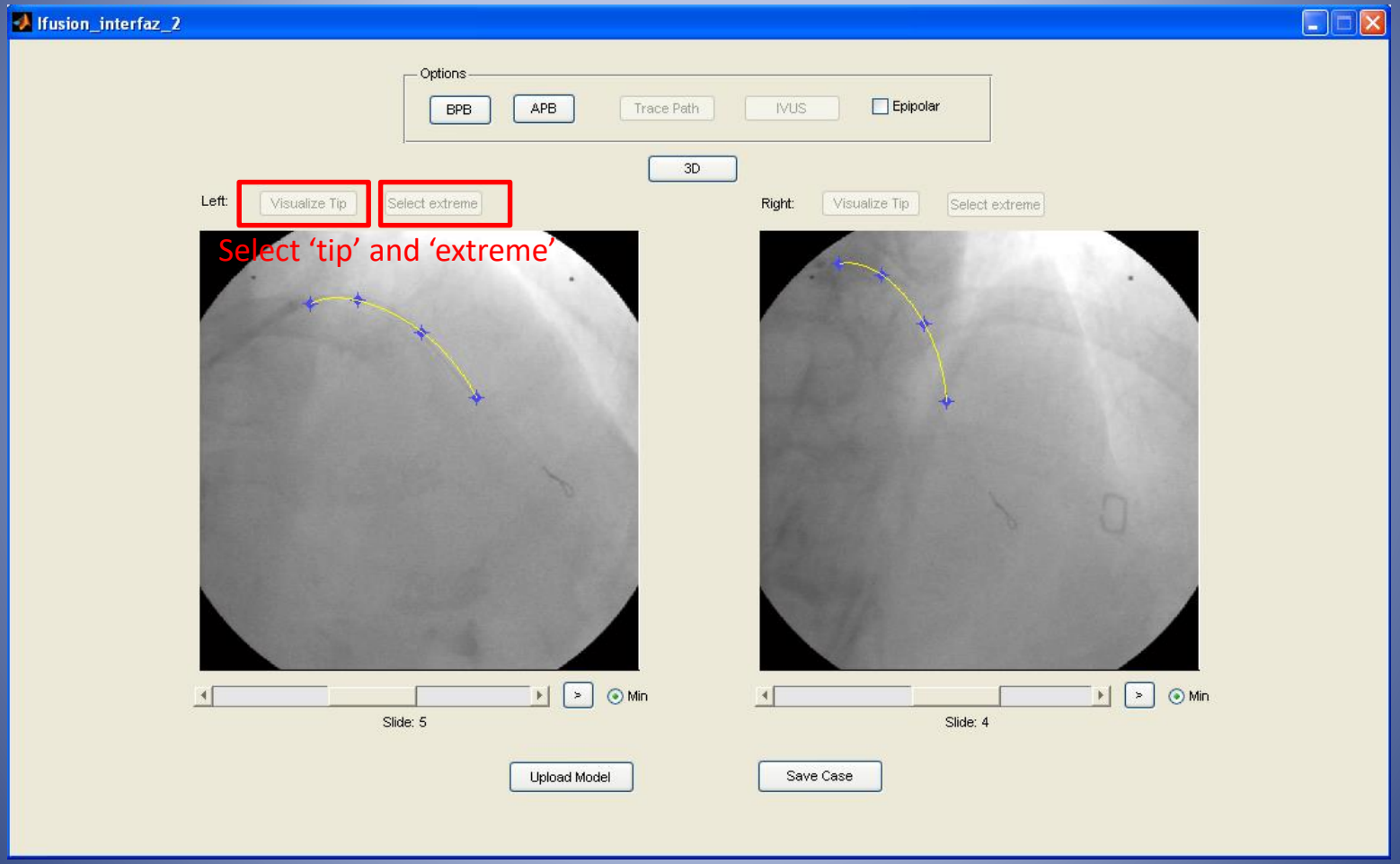
iFusion: INTERFACE_2
Curve Model Generation



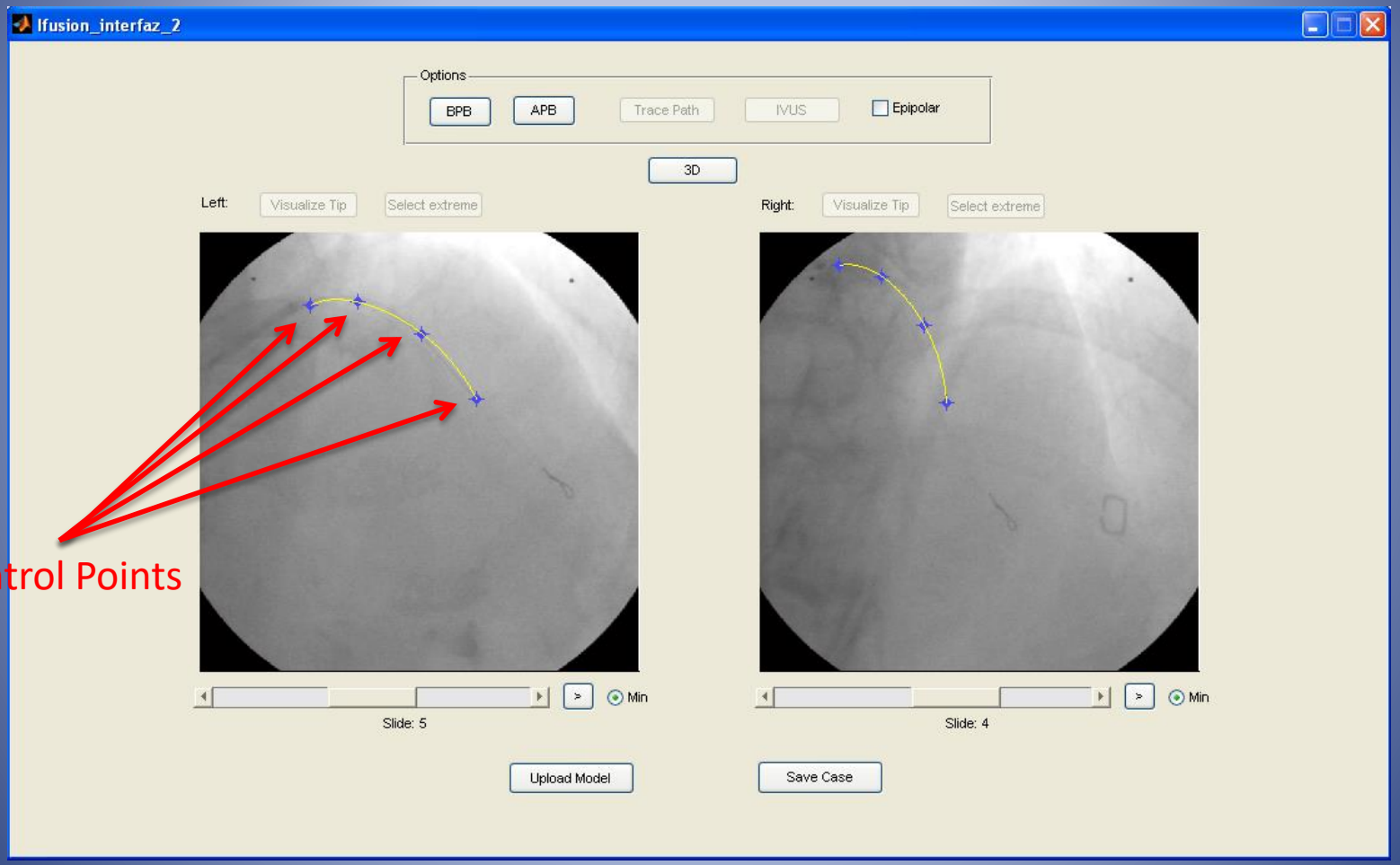
iFusion: INTERFACE_2
Curve Model Generation



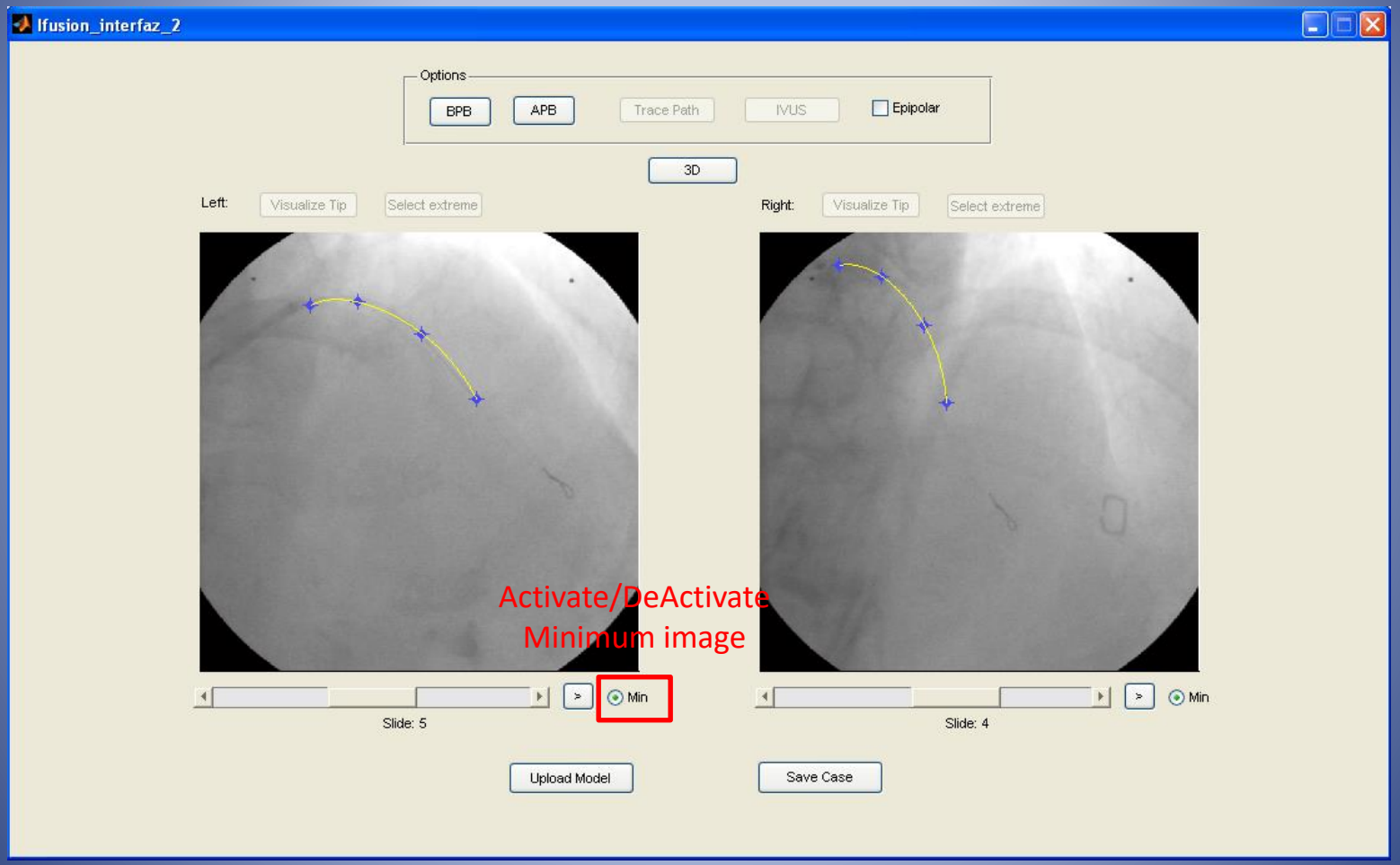
iFusion: INTERFACE_2
Curve Model Generation



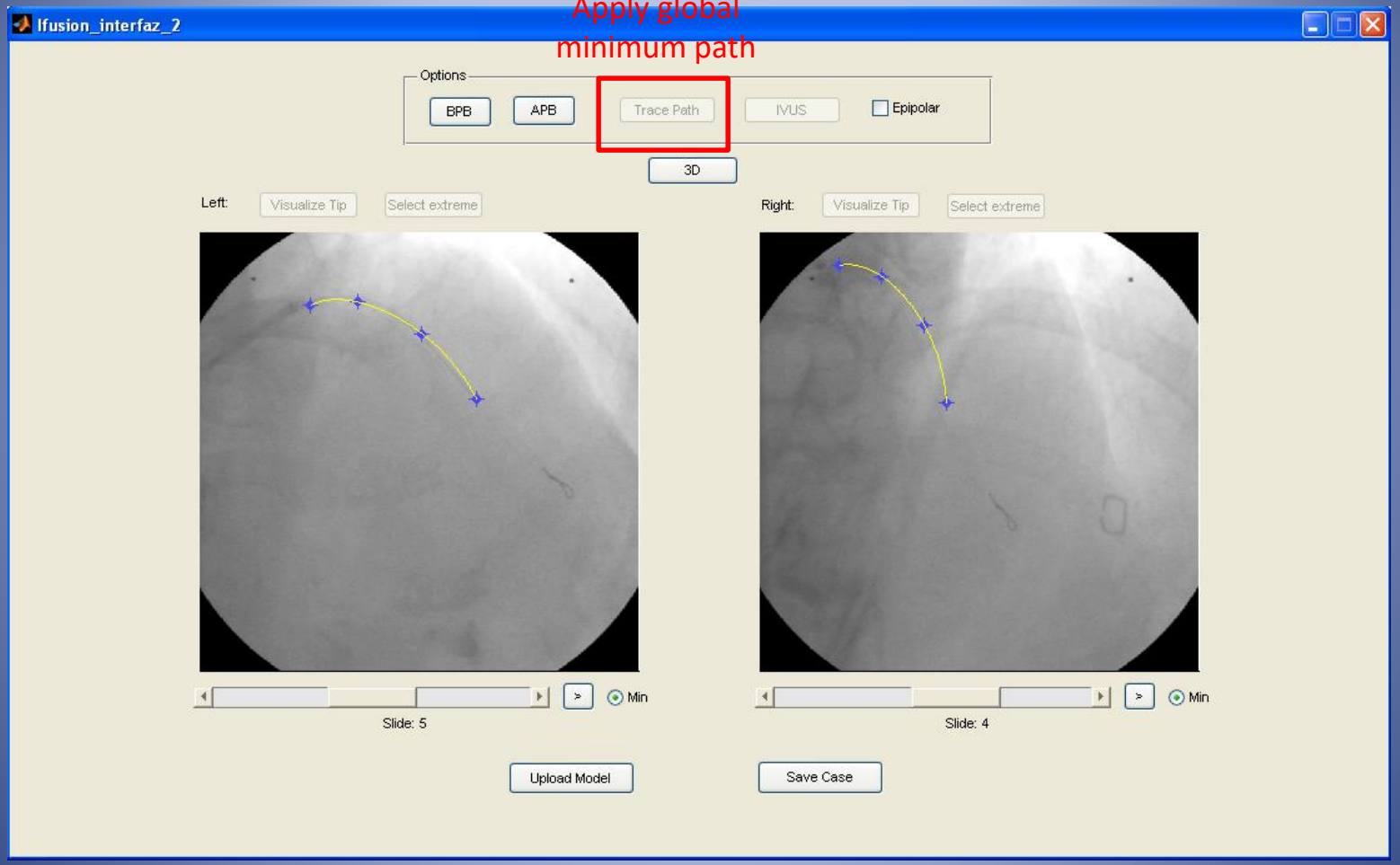
iFusion: INTERFACE_2
Curve Model Generation



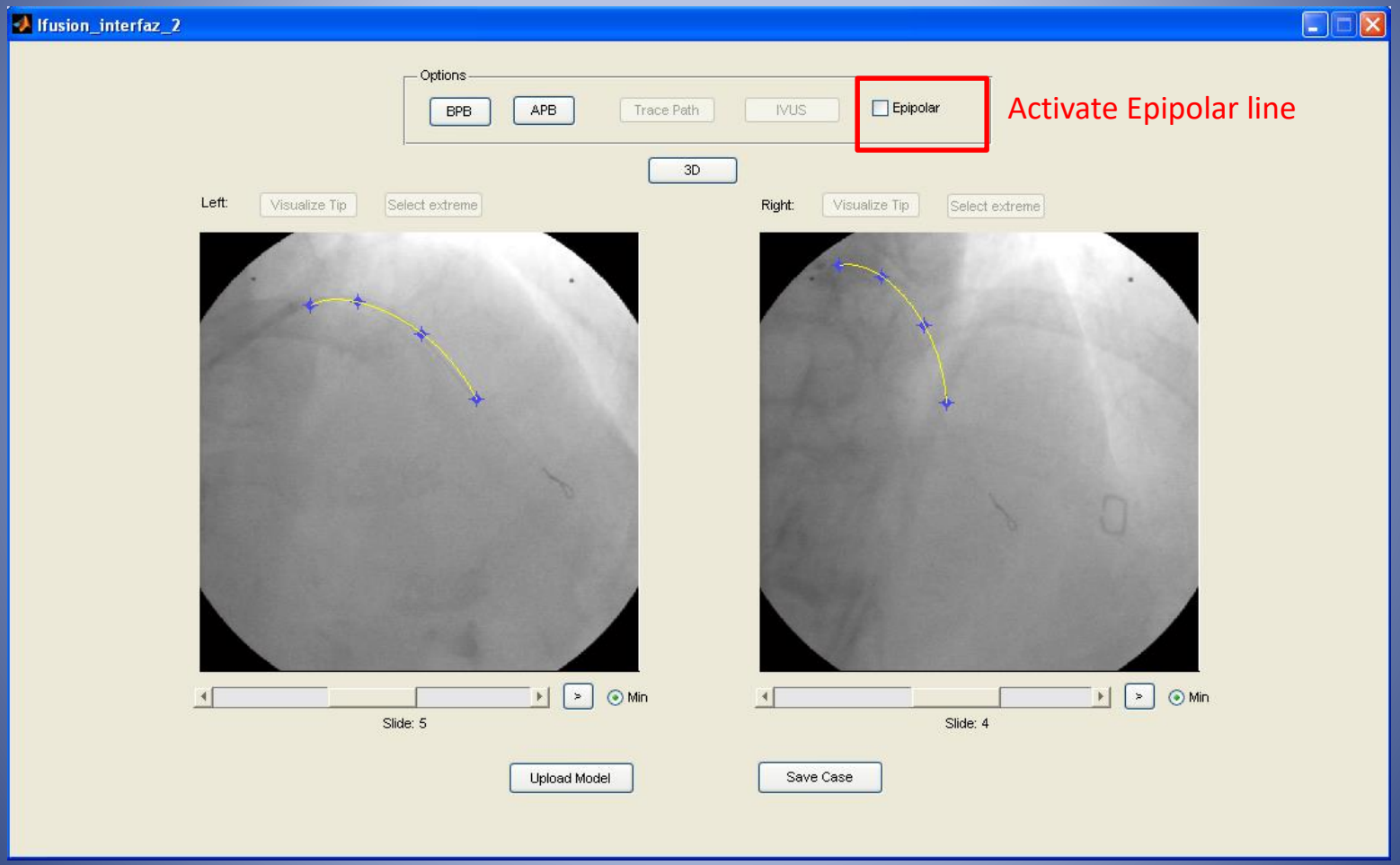
iFusion: INTERFACE_2
Curve Model Generation



iFusion: INTERFACE_2
Curve Model Generation

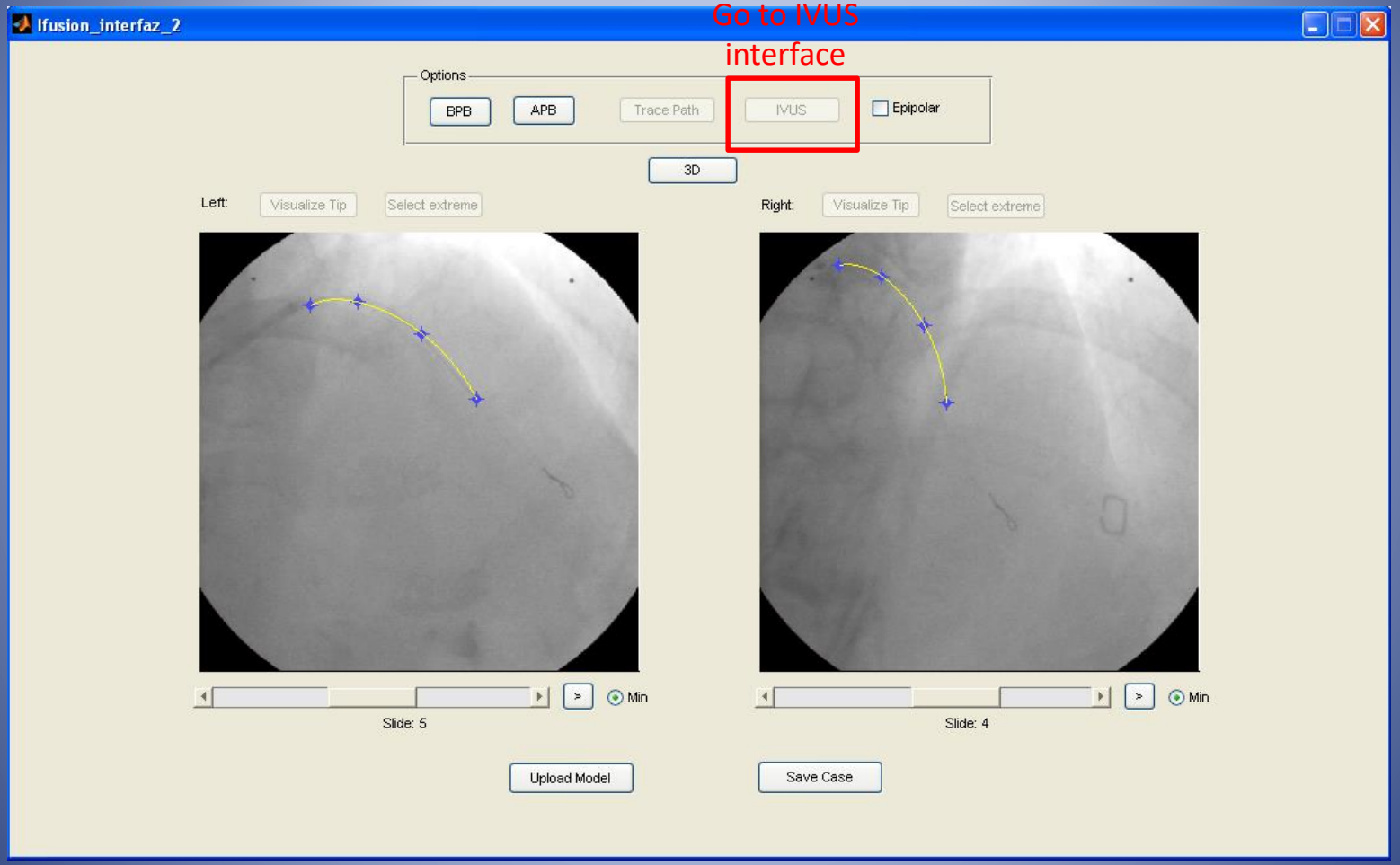


iFusion: INTERFACE_2
Curve Model Generation

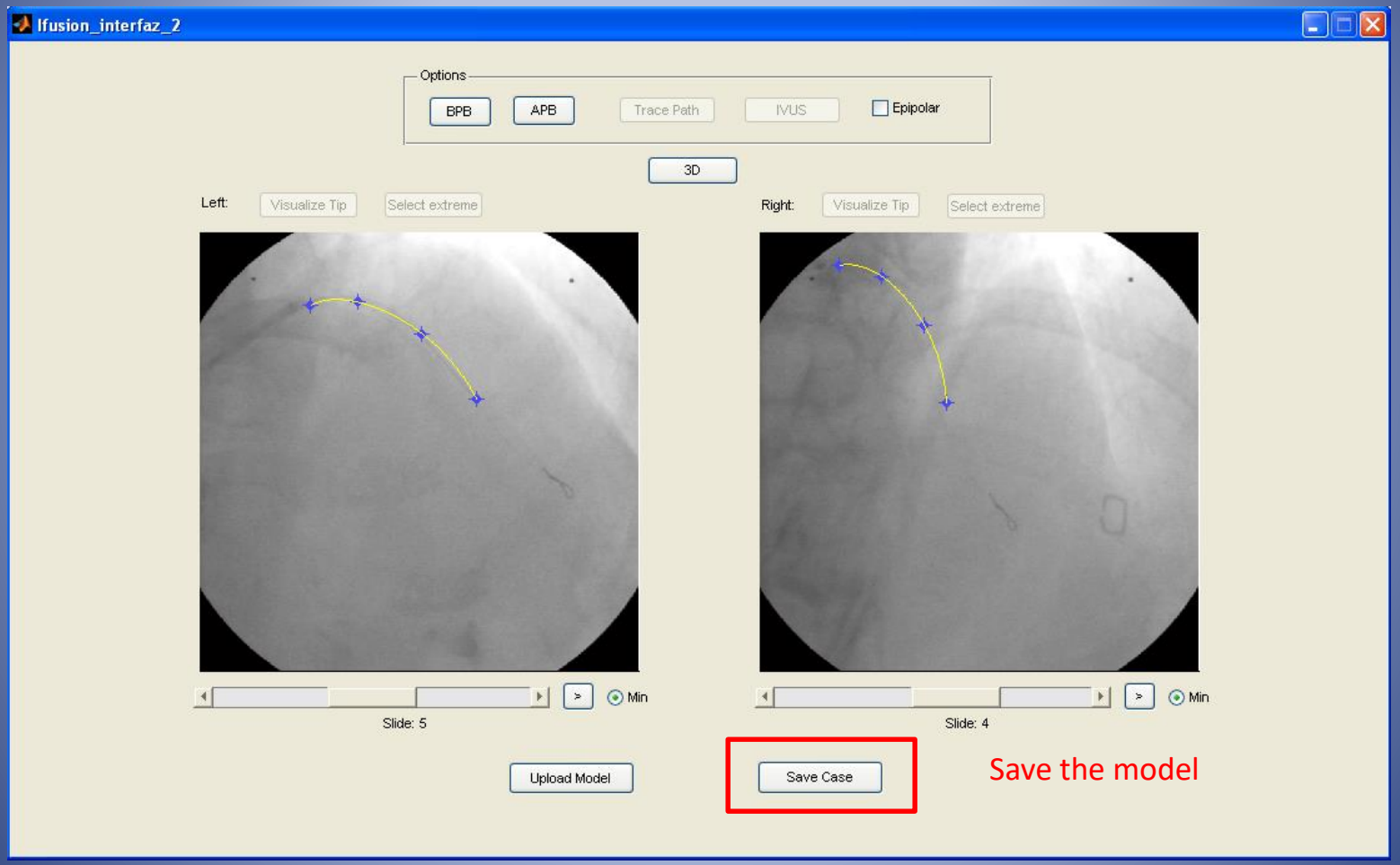


iFusion: INTERFACE_2
Curve Model Generation

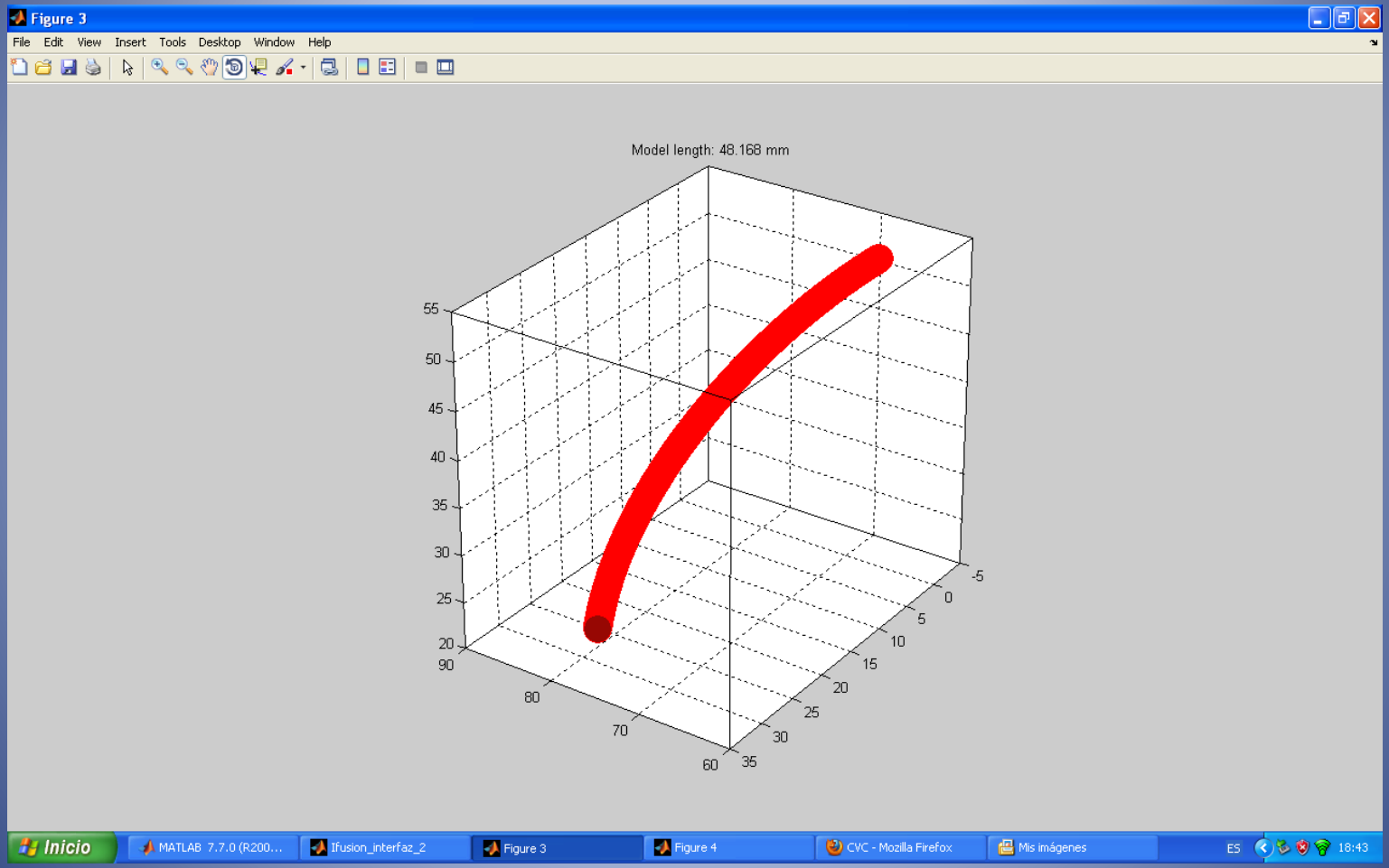
iFusion: INTERFACE_2
Curve Model Generation



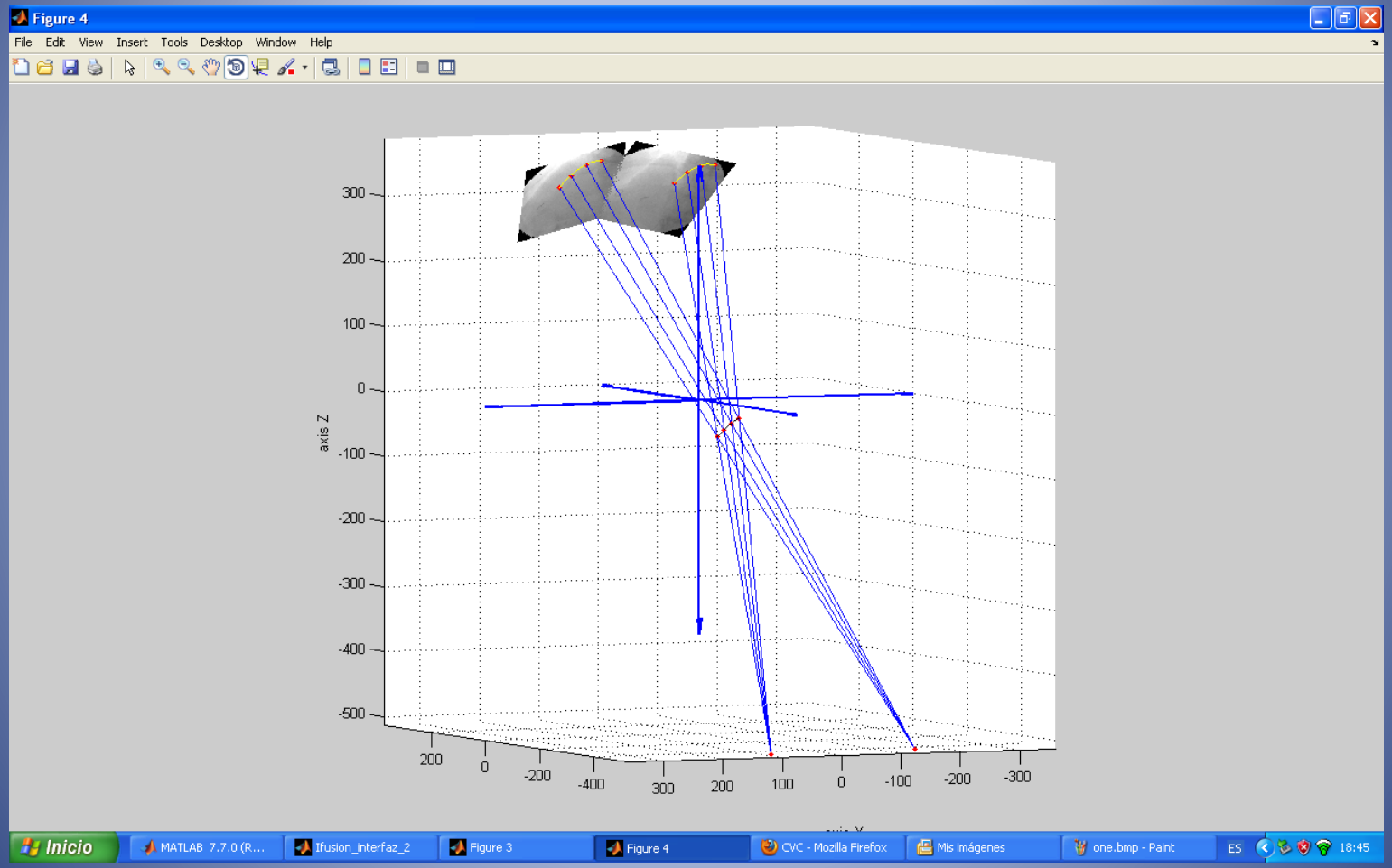
iFusion: INTERFACE_2
Curve Model Generation



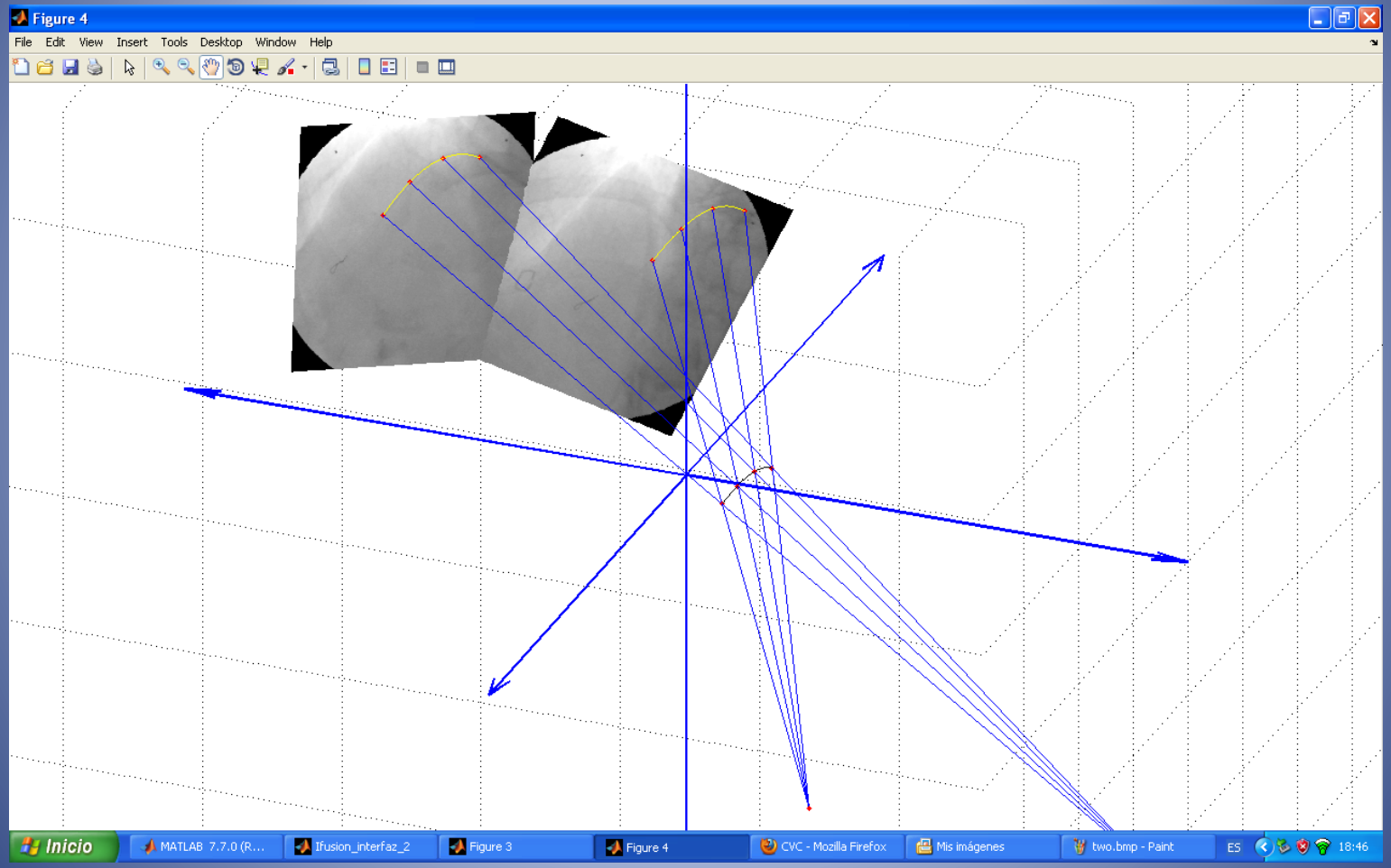
iFusion: 3D vessel reconstruction



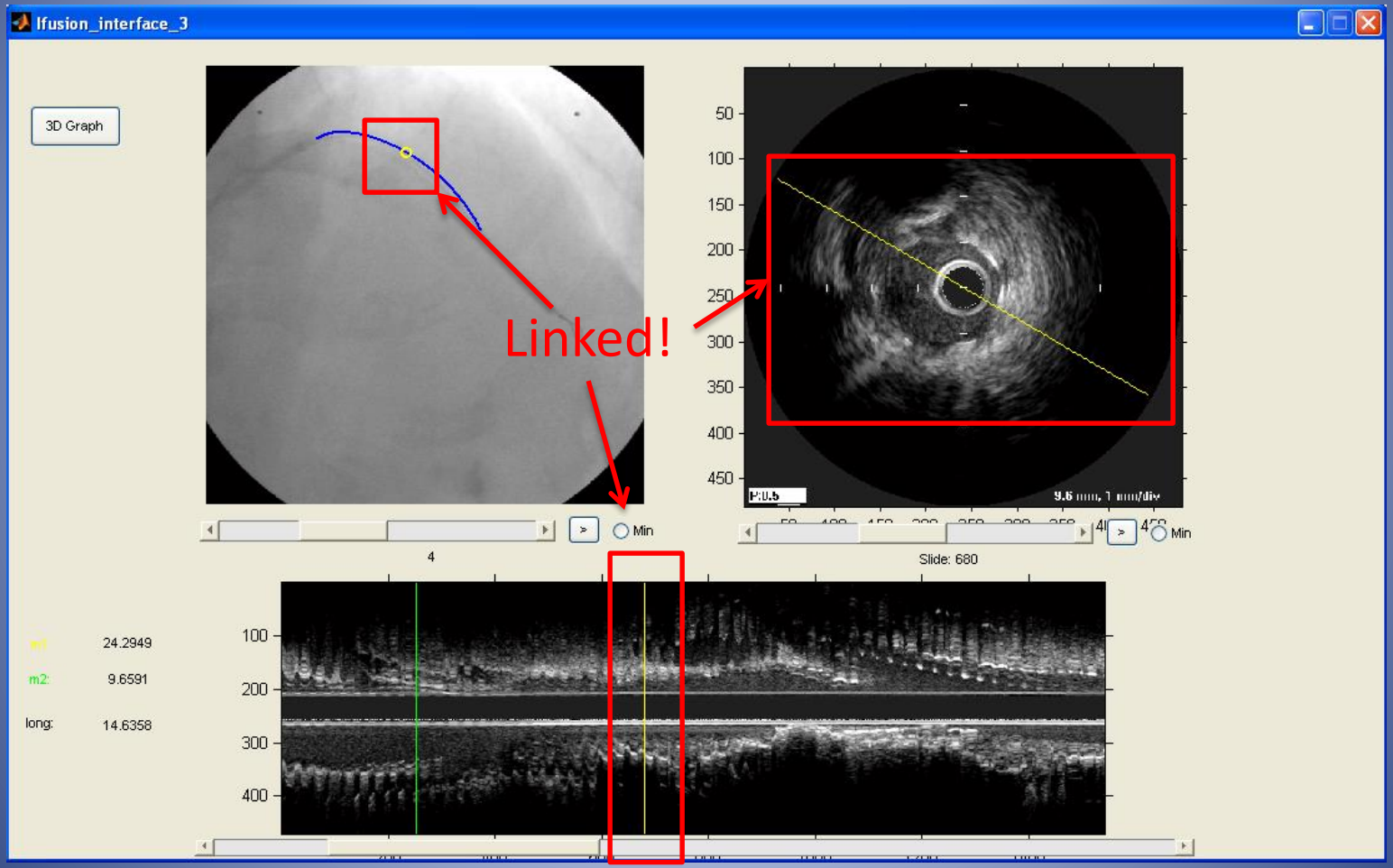
iFusion: complete 3D geometry of the global and local reference systems



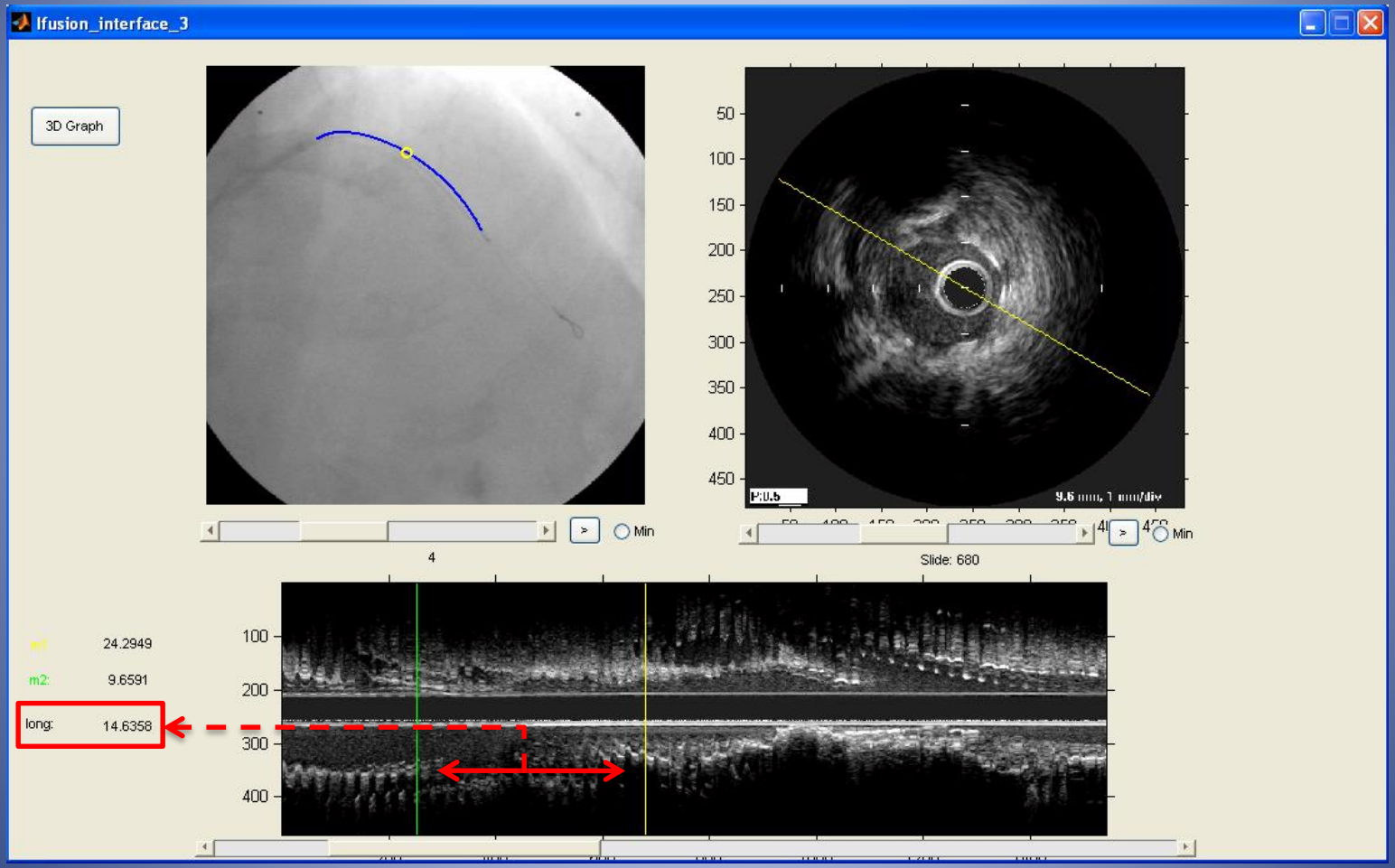
iFusion: complete 3D geometry of the global and local reference systems



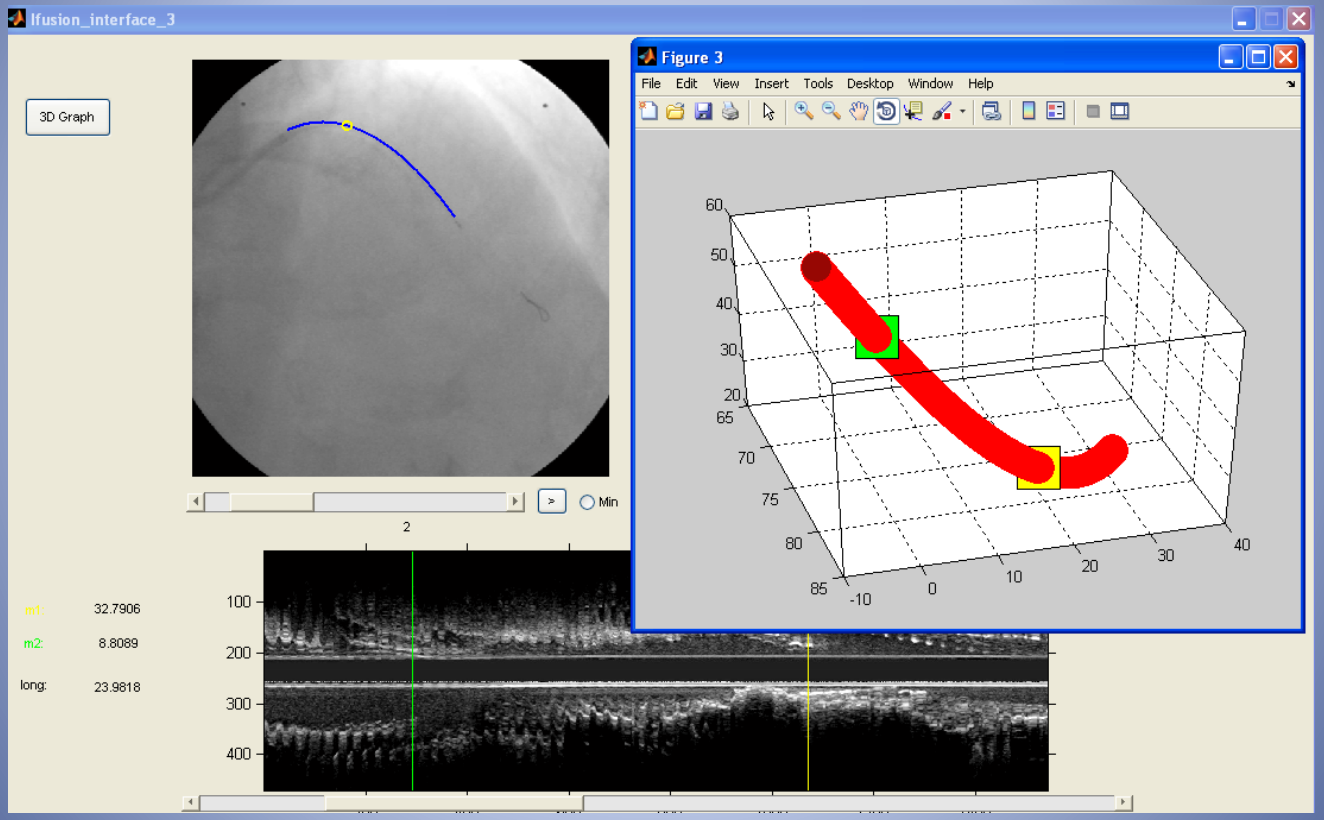
iFusion: INTERFACE_3
IVUS to Angio correspondence



iFusion: INTERFACE_3
IVUS to Angio correspondence

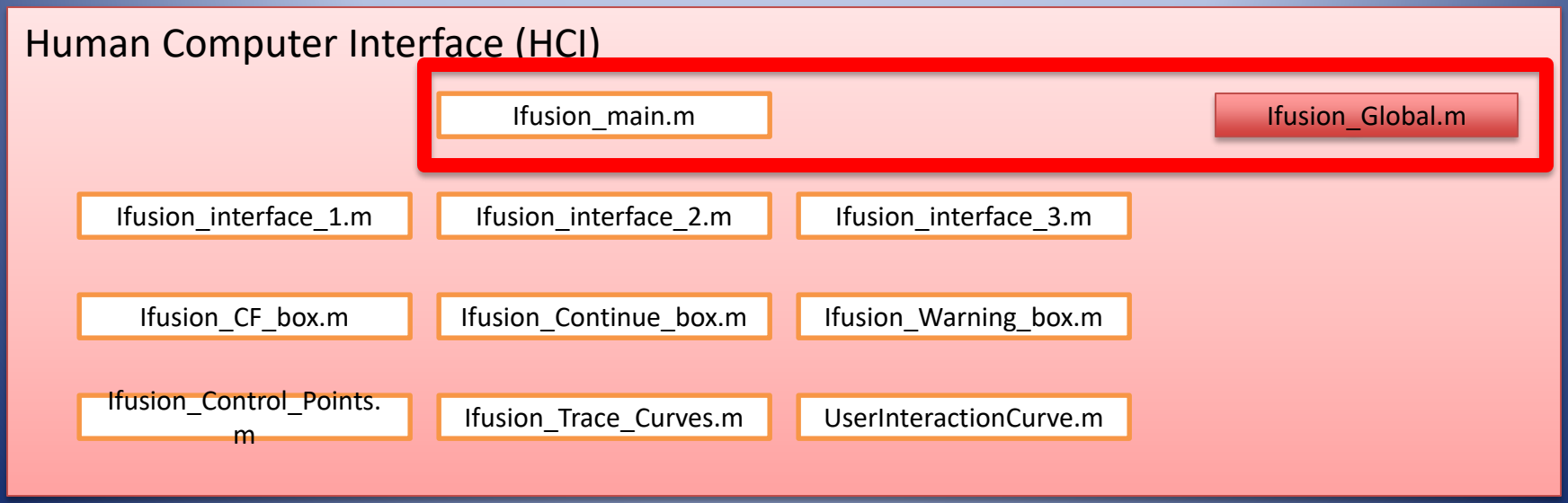


iFusion: 3D visualization of vessel and selected positions



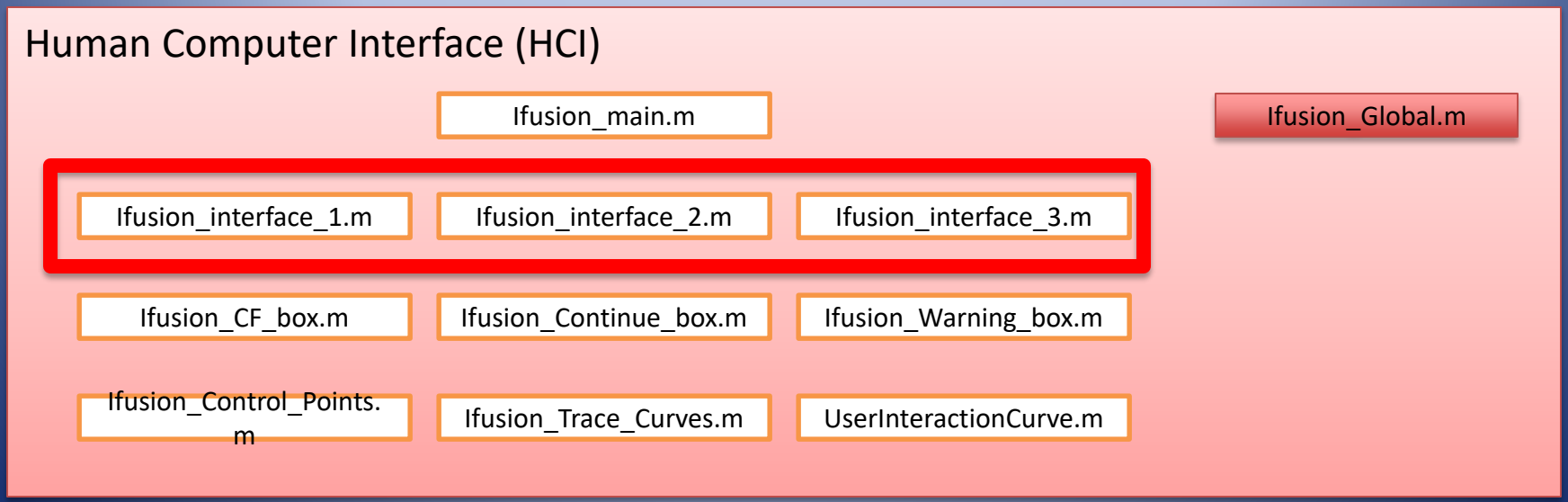
iFusion: INTERFACE

- *Ifusion_main:* Define Global variables and call the three interfaces in the correct order.
- *Ifusion_Global:* Contains global variables used by all the INTERFACE functions.



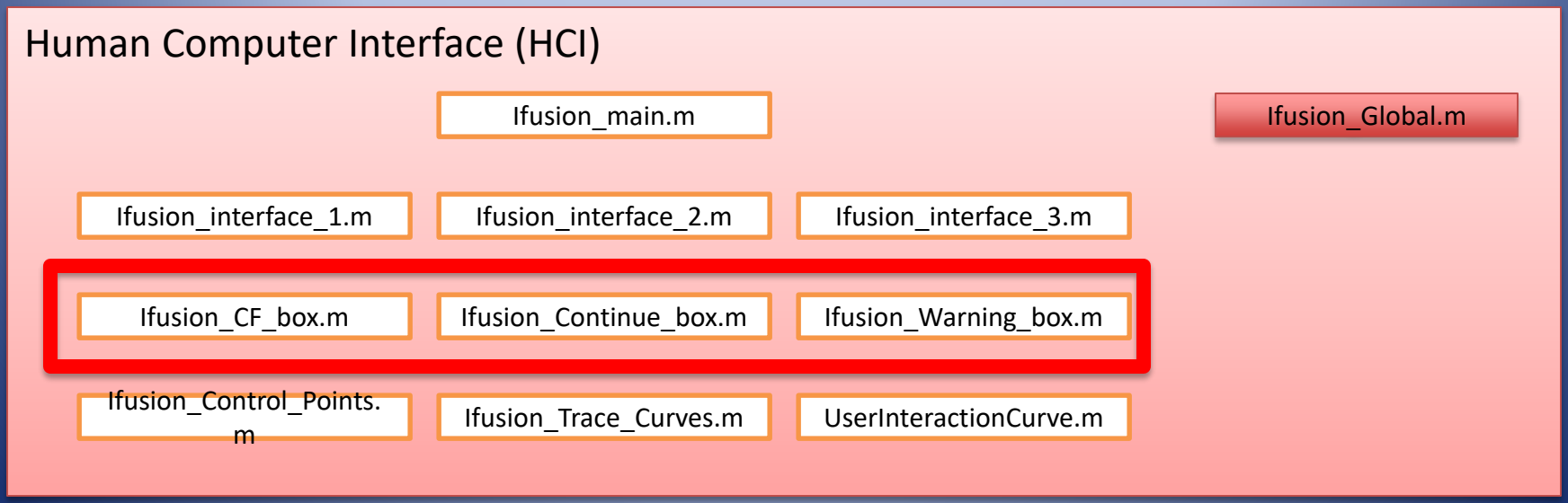
iFusion: INTERFACE

- *Ifusion_interface_1*: Load the X-ray DICOM files.
- *Ifusion_interface_2*: Interact with the user to find starting and ending points of the desired structure to be modeled; generate 3D model.
- *Ifusion_interface_3*: Load the pullback and visualize results.



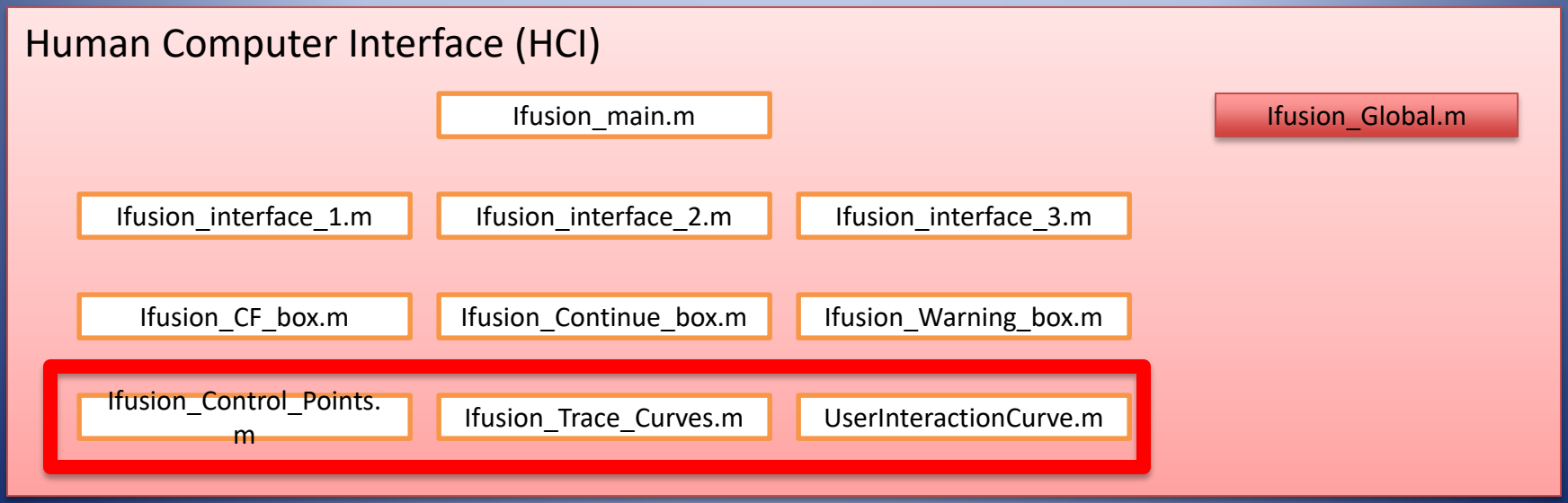
iFusion: INTERFACE

- *Ifusion_CF_box:* Interaction with the user to input the CF distance if not provided by the infos of the DICOM files.
- *Ifusion_Continue_box:* Yes/No box to assure the user desires to continue with the processing.
- *Ifusion_Warning_box:* Warning message if DICOM files are missing.



iFusion: INTERFACE

- *Ifusion_Control_Points:* Given two paths (left and right) appropriately assigns Control Points.
- *Ifusion_Trace_Curves:* Draw model curves on the display.
- *UserInteractionCurve:* Update the curves when a control point is moved.



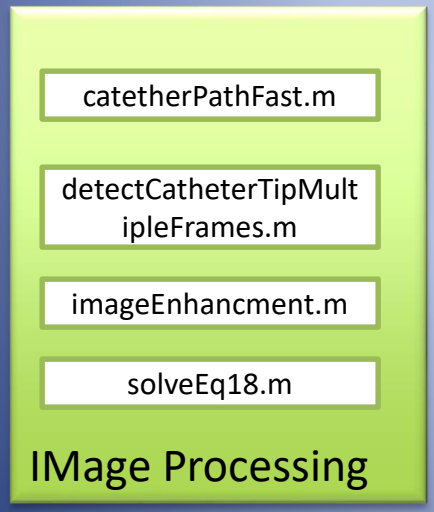
iFusion: CORE

iFusion: CORE

- IMP (Image Processing): 4 functions.
- GEO (Geometry): 7 functions.
- MISC (Miscellaneous): 1 function.

iFusion: CORE, IMP

- *catheterPathFast:* Implements the Global Minimum for Active Contour Models (see Cohen et. Al.).
- *detectCatheterTipMultipleFrames:* Detects rounded objects in a sequence of frames.
- *imageEnhancement:* Enhances the image by local histogram equalization.



- *solveEq18:* Specific routine used by the method catheterPathFast. Actually rapidly solves Eq. 18 in Cohen et Al.

iFusion: CORE, GEO

- *Project_point_line:* Projects a point to the closest point of a line, in 2D.
- *Epipolar:* Builds the epipolar line of a reference system, with respect to another.
- *StretchLine:* Takes an epipolar segment and crops it to fit with the image size.

iFusion: CORE, GEO

- *Intersection_plane_line:* Finds the 3D point that results from the intersection of a given plane and a line.
- *angles2refsys:* Computes the local reference system starting from CArm angles.
- *Intersection_line_line:* Finds the 3D point that is the closest of both given lines, or if it exists, the intersection between lines.
- *ComputeAngles:* Computes the angle of a 2D vector, i.e. the angle of polar coordinates of the vector.

iFusion: CORE, MISC

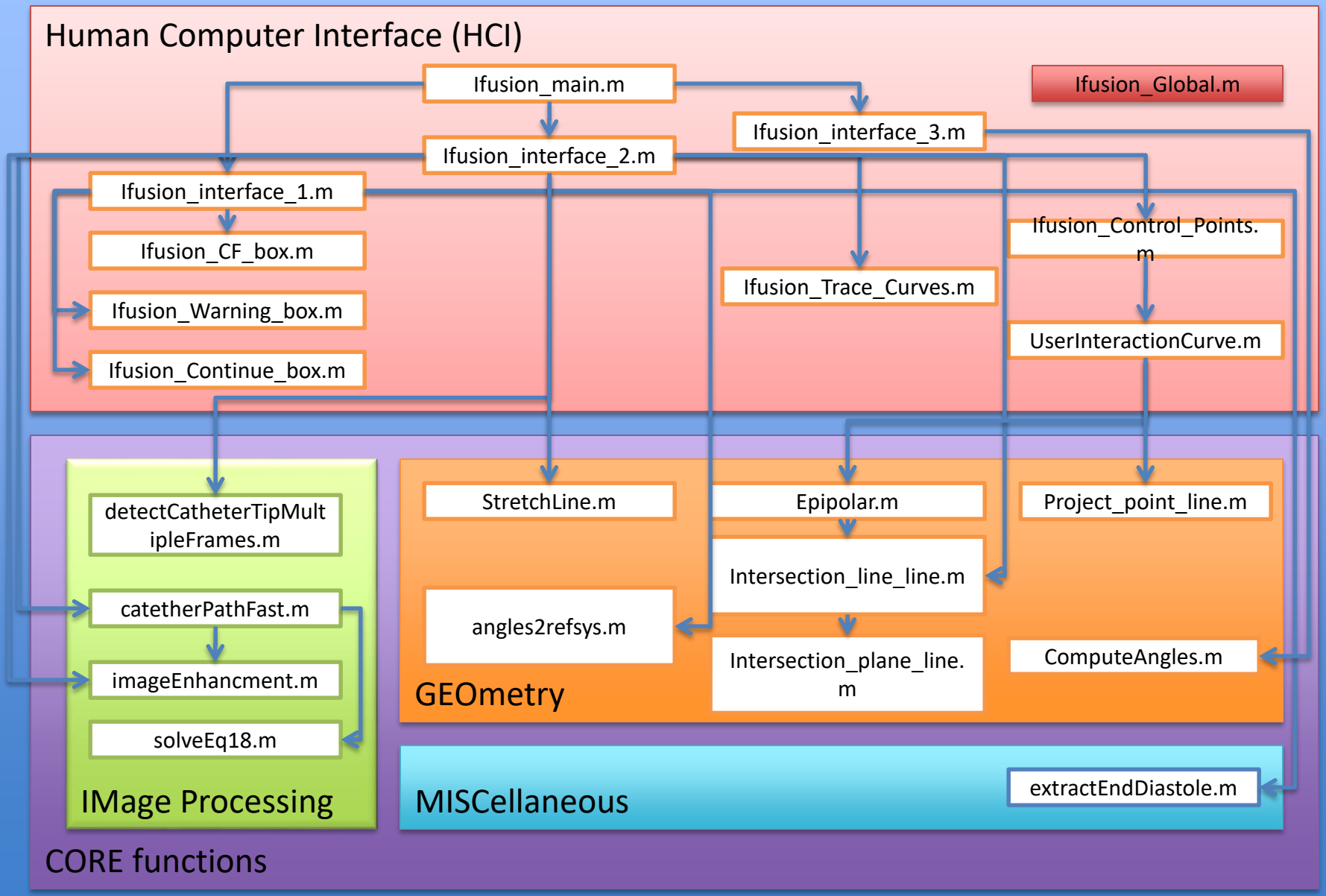
- *extractEndDiastole:* If the ECG signal is available, this function is used to perform an ECG gating at end diastole. It can be easily modified to perform the gating in different ECG positions.

MISCellaneous

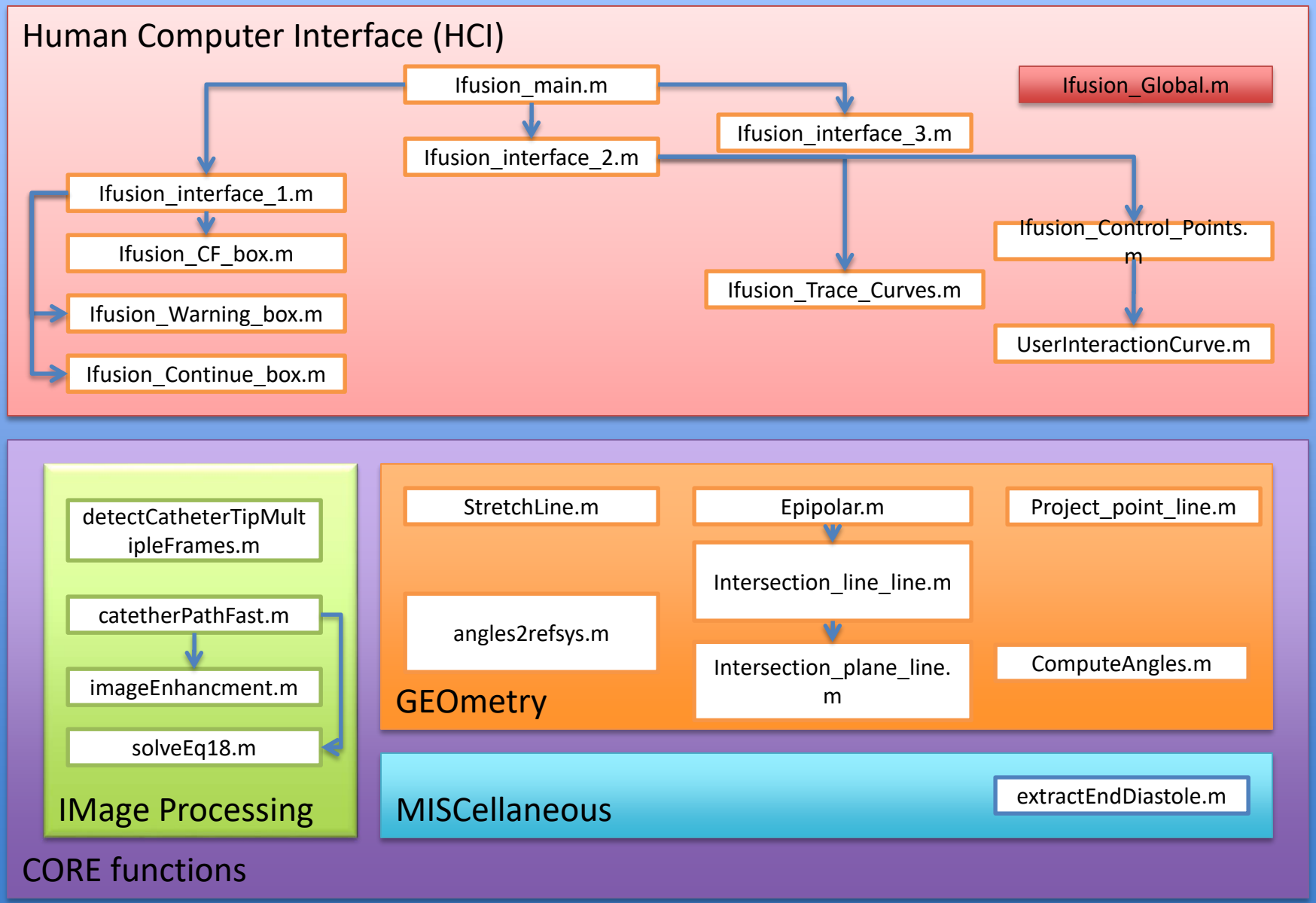
extractEndDiastole.m

iFusion: Calls architecture

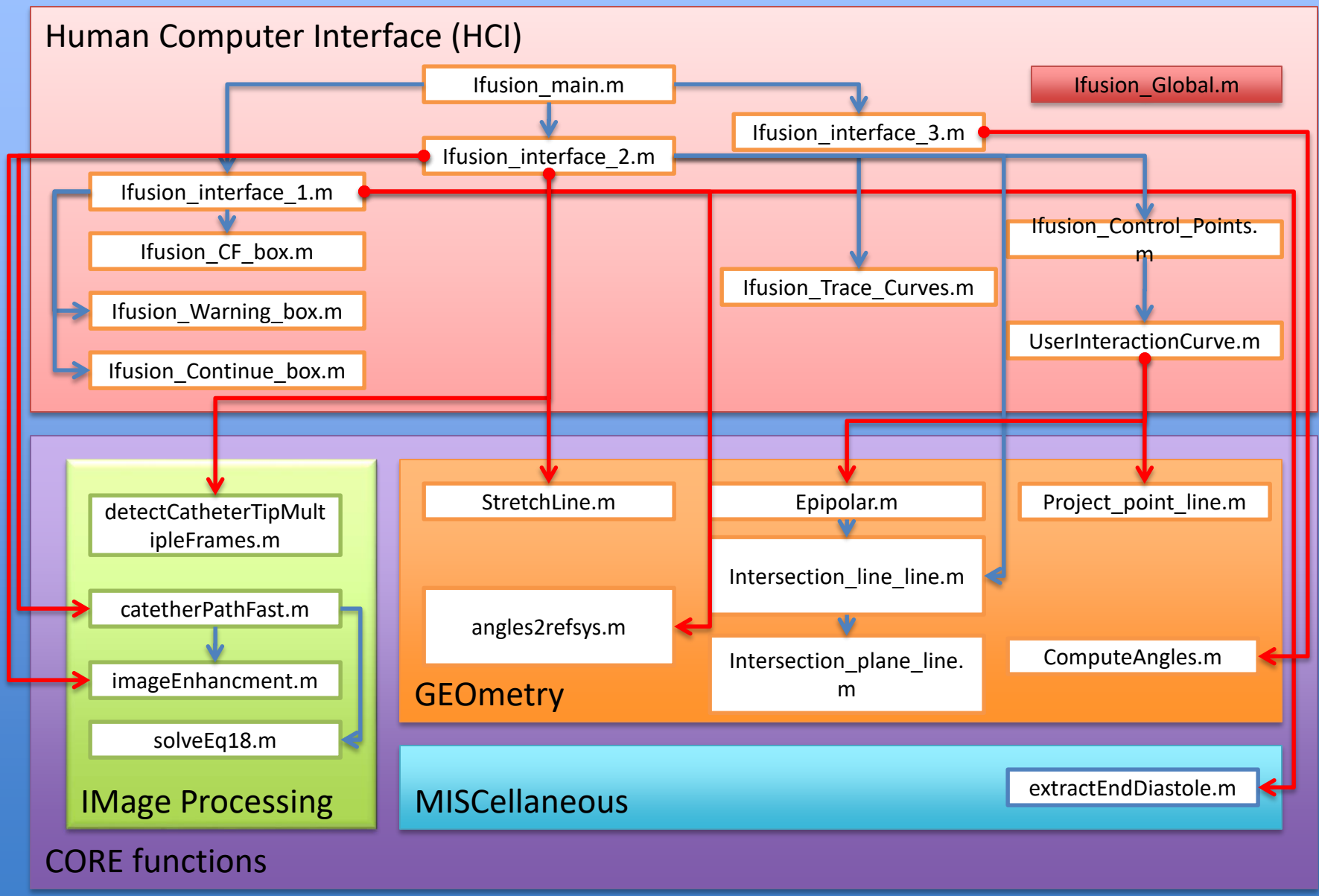
Schema of calls: ALL



Schema of calls: Intra-blocks



Schema of calls: Inter-blocks in RED

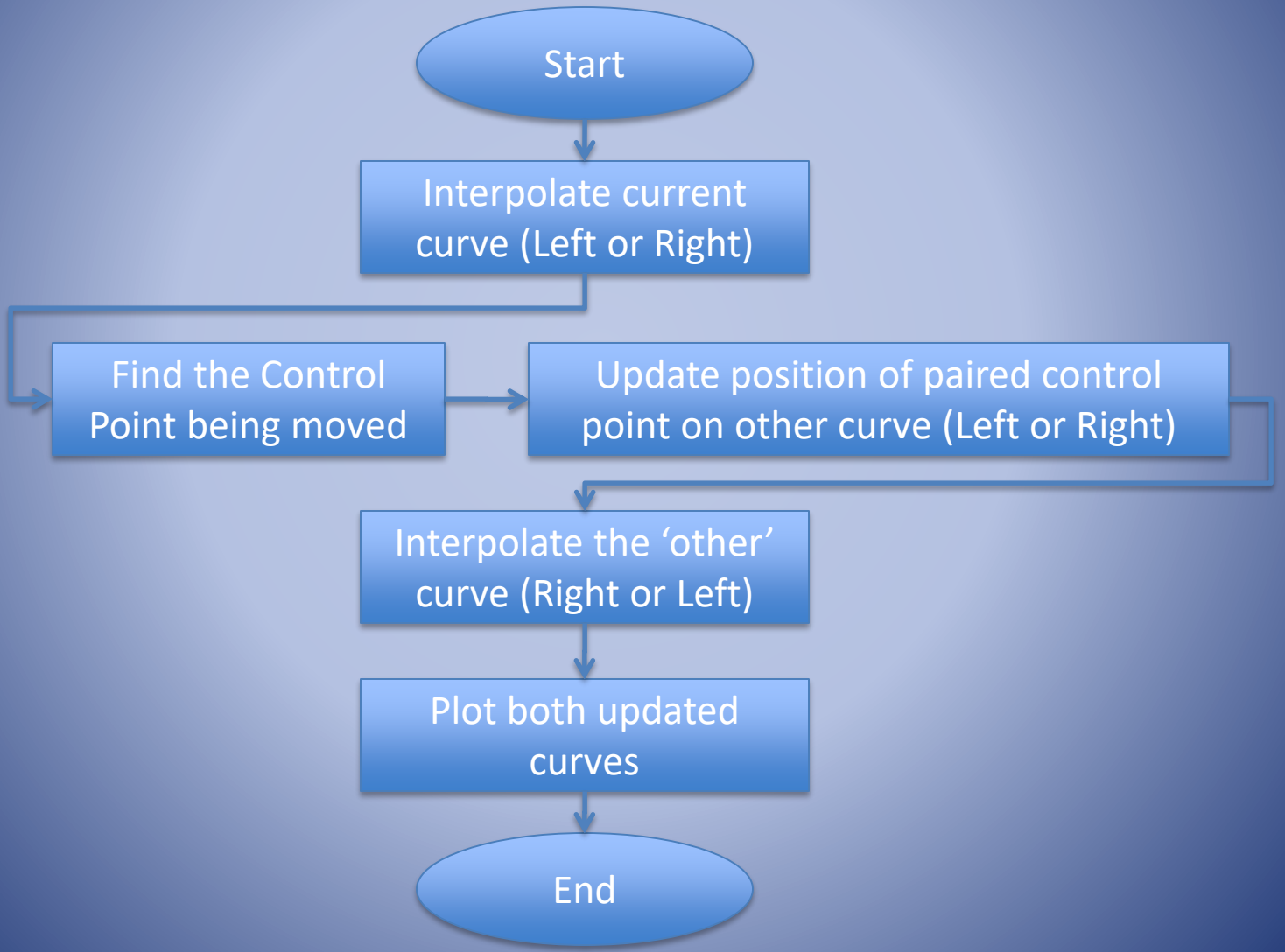


iFusion: Flow diagrams and Graphical explanations

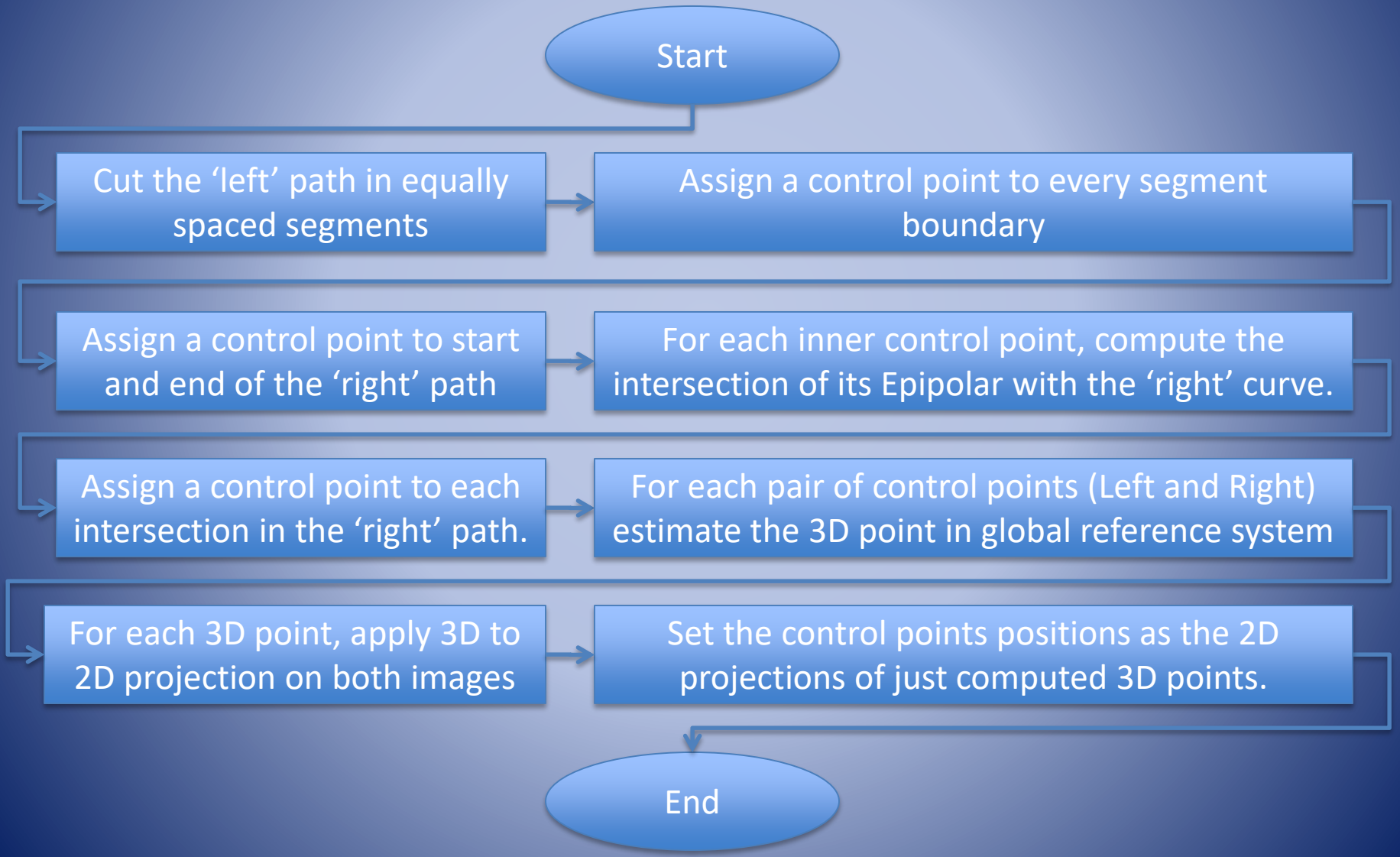
(only of some functions)

Flow Diagram

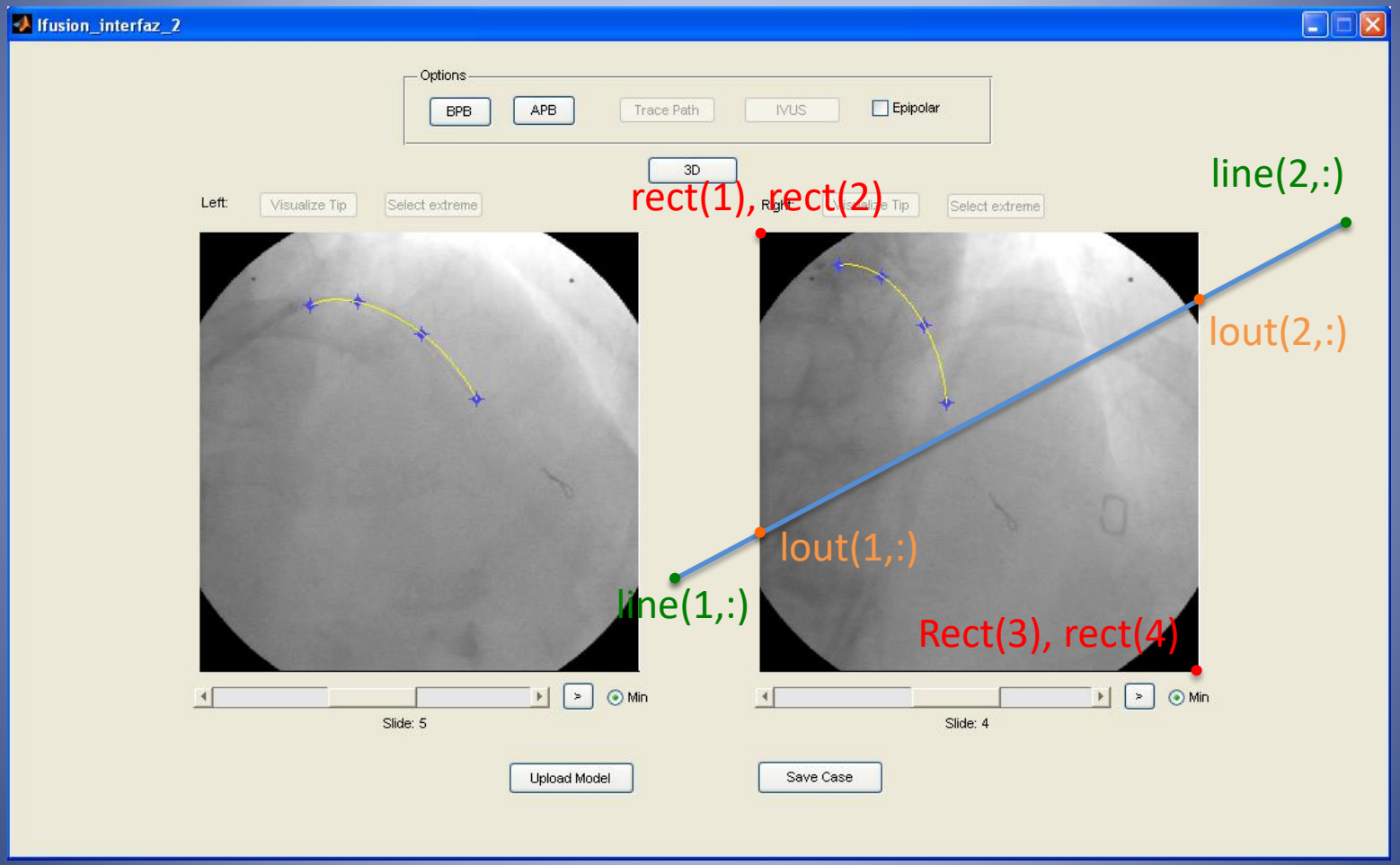
(HCI: UserInteractionCurve.m)



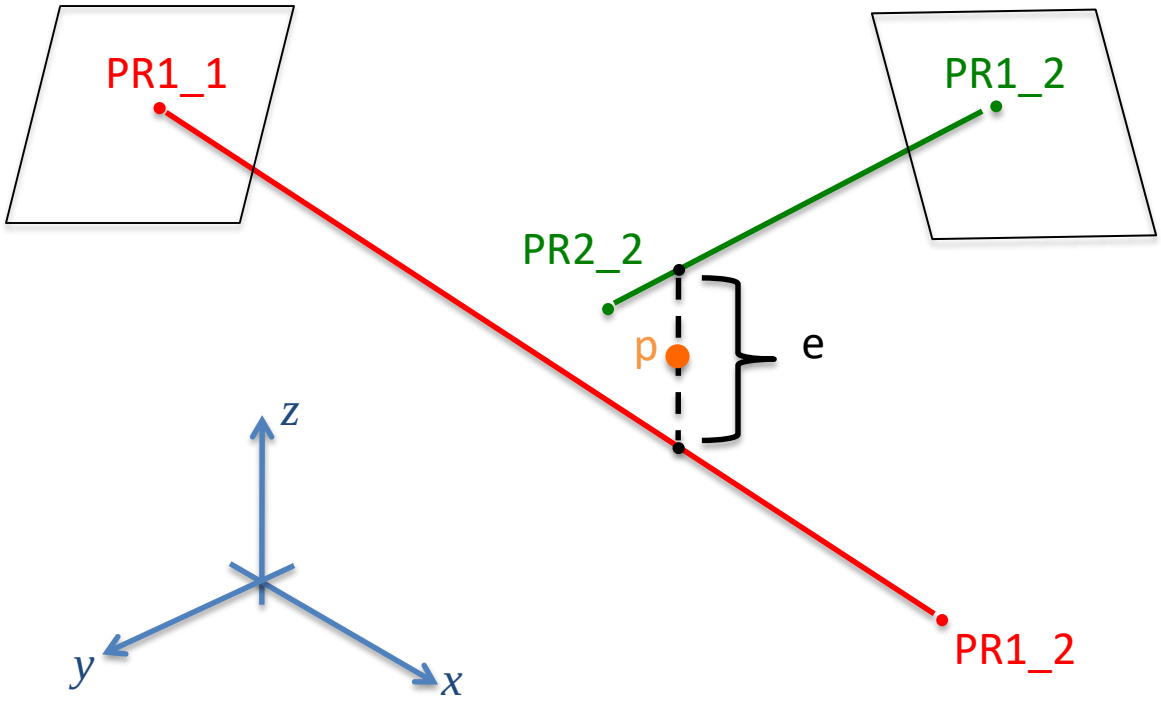
Flow Diagram
(HCI: Ifusion_Control_Points.m)



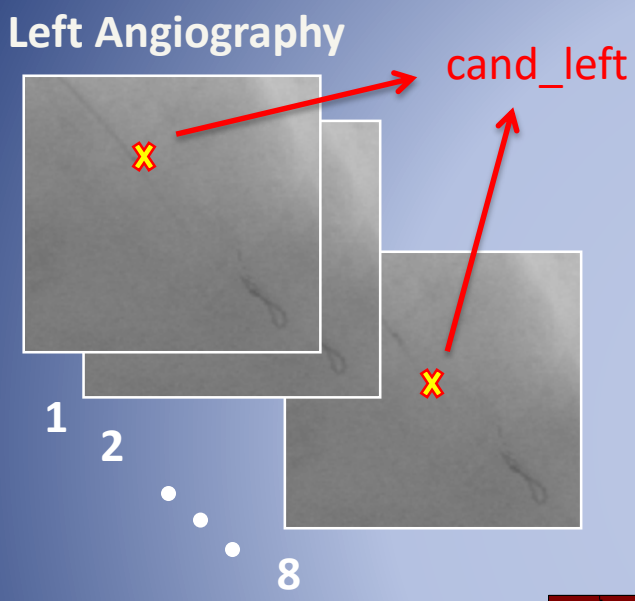
Graphical explanations
(HCI: StretchLine.m)

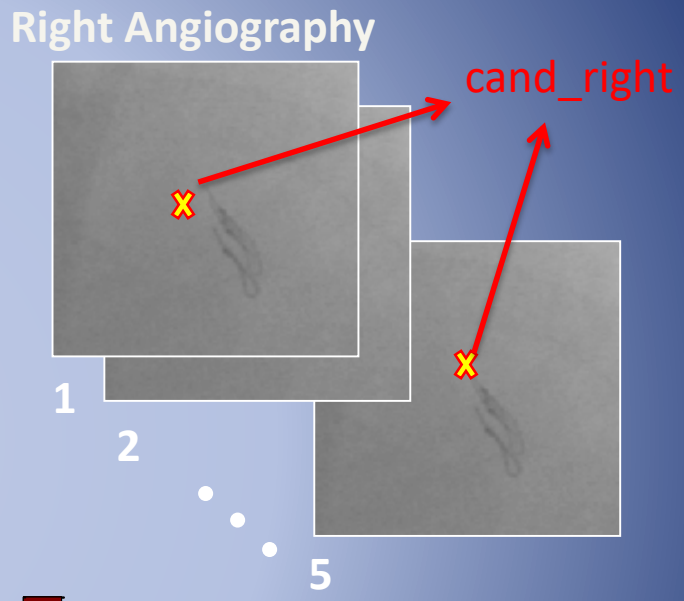


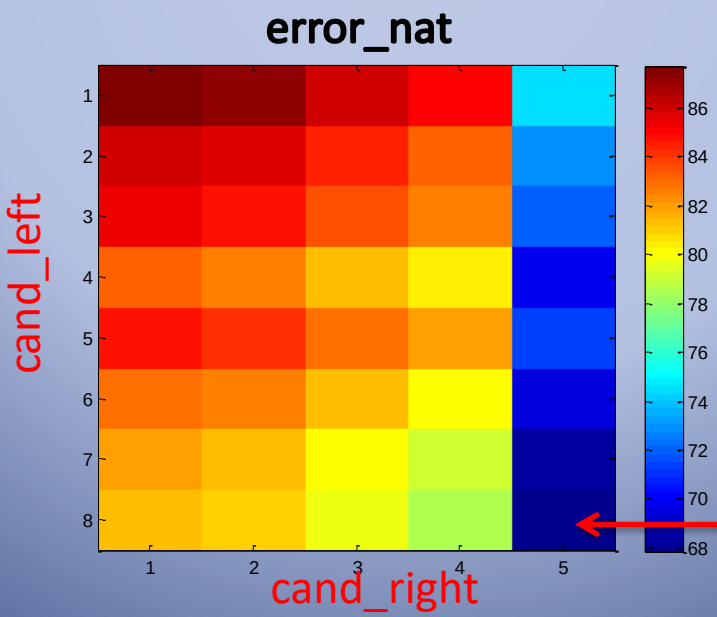
Graphical explanations
(HCI: Intersection_line_line.m)



Graphical explanations
(Find optimal point pair)







3D anti-projection error
(see 'e' in previous slide)

Best pair:
- Left: 8
- Right: 5

iFusion: In Code Documentation

(very detailed for all CORE functions)

(sufficient for all HCI functions)

Function Header Comments

- SYNOPSIS
- INPUT variables
- OUTPUT variables
- REFERENCES
- COMMENTS

Example

```
>> help imageEnhancment


  SYNOPSIS: Enhances the input image by means of local histogram
            qualization on a 27x27 neighborhood.

  INPUT:    im_in:  an image

  OUTPUT:   im_out: enhanced image

  REF:      - Fusion.pdf - Page 6, Sect 2.1
            - Fusion.pdf - Page 7, Figure 2.3
           - Equalization as in "MULTIMODAL REGISTRATION OF INTRAVASCULAR
           ULTRASOUND IMAGES AND ANGIOGRAPHY", page 2, formula (1).


  COMMENTS: This algorithm can be performed by means of the following
            (commented) for loop. However, the matlab function nlfilter
            allows to do it a bit faster.
```

Functions add-ons

- Flag ‘visualize’ (normally set to FALSE)
- If visualize = TRUE
  - open one or more figures
  - visualize the algorithm computation graphically.

iFusion: Technical requierements

Technical requirements and limitations

- The CODE has been developed on:
  - MATLAB 7.7.0.471 (R2008b)
  - Microsoft XP Version 5.1 (Build 2600: Service Pack 3)
  - Java VM Version: 1.6.0_04 with Sun Microsystems Inc. Java HotSpot™ Client VM.
- If the PullBack is too big (causing an Out-of-Memory error), the third interface works in ‘simulation’ mode, thus not visualizing the pullback.

iFusion: Credits

Original software developer: *David Rotger, PhD*

Leading Technology transfer: *Juan Diego Gomez, Ph.D. Student. and Carlo Gatta, Ph.D.*

MATLAB Coding:

Juan Diego Gomez, Ph.D. Student.

Head of the group: P*etia Radeva Ivanova, Ph.D.*