Exercise 31: Importing 3D Medical DICOM Image Data and converting into 3D Solids

2017-02-19: Tim C. Lueth, MIMED - Technische Universität München, Germany (URL: http://www.mimed.de)

Contents

- Summary of the already explained aspects of the VLFL-Toolbox
- Motivation for this tutorial
- 1. Reading DICOM models as voxel model from disk and resize voxel models (VM)
- 2. Solid skull bone reconstruction using SGofVMdelaunay
- 3. Solid skull bone reconstruction using SGofVMmarchcub
- 4 Show the Voxel model in quadrant 1-2-4 and surface model in quadrant 3
- 5. Create a surface model and convert it into a Voxel model
- 6. Plot the surface model in 4 quadrant plot
- 7. Write the surface model as DICOM directory on disk
- 8. Select Point in 3D
- Final Remarks

Summary of the already explained aspects of the VLFL-Toolbox

The previous examples of this series covered the following topics:

- Example 01: First Steps Using the VLFL-Toolbox for Solid Object Design
- Example 02: Using the VLFL-Toolbox for STL-File Export and Import
- Example 03: Closed 2D Contours and Boolean Operations in 2D
- Example 04: 2.5D Design Using Boolean Operators on Closed Polygon Lists (CPL)
- Example 05: Creation, Relative Positioning and Merging of Solid Geometries (SG)
- Example 06: Relative Positioning and Alignment of Solid Geometries (SG)
- Example 07: Rotation of Closed Polygon Lists for Solid Geometry Design
- Example 08: Slicing, Closing, Cutting and Separation of Solid Geometries
- Example 09: Boolean Operations with Solid Geometries
- Example 10: Packaging of Sets of Solid Geometries (SG)
- Example 11: Attaching Coordinates Frames to Create Kinematik Models
- Example 12: Define Robot Kinematics and Detect Collisions
- Example 13: Mounting Faces and Conversion of Blocks into Leightweight-structures
- Example 14: Manipulation Functions for Closed Polygons and Laser Cutting (SVG)
- Example 15: Create a Solid by 2 Closed Polygons
- Example 16: Create a Solid along a contour
- Example 20: Programmaticaly Interface to SimMechanics Multi-Body Toolbox
- Example 21: Programmaticaly Design of SimMechanisms Mechanisms and Simulink Signals
- Example 22: Creation of fixed nodes for the assembly of joints

- Example 23: Automatic Creation of a Missing Link and 3D Print of a Complete Model
- Example 24: Automatic Creation of a Joint Limitations
- Example 25: Automatic Creation of Video Titels, Endtitels and Textpages
- Example 26: Create Mechanisms using Universal Planar Links
- Example 27: Fourbar-Linkage: 2 Pose Syntheses and Linkage Export for 3D Printing
- Example 28: Fourbar-Linkage: 3 Pose Syntheses and Linkage Export for 3D Printing
- Exercise 30: Create Mechanisms using Universal Planar Links
- Exercise 31: Importing 3D Medical DICOM Image Data and converting into 3D Solids

Motivation for this tutorial

- VMreaddicomdir - reads in a voxel model
- VMresize - resizes of a voxel model
- SGofVMdelaunay - creates a surface model using delaunay (slow)
- SGofVMmarchcube - creates a surface model using marching cube (fast)
- SGcut - cuts surface models
- CPLofSGslice - creates a slice contour at a specific height/direction
- PLFLofCPLdelauny - tesselates the facets of a CPL using delaunay
- PLFLofCPLpoly - tesselates the facets of a CPL using mapping toolbox
- VMplot - plots a voxel mode
- VMplotslide - plots a voxel mode for slider navigation
- VMimage - plots a voxel image
- VMmontage - montage of voxel
- VMpseudo3D - creates a pesudo 3D image
- VMuidicom - select and read a voxel model
- VMreaddicom - read a dicom file

1. Reading DICOM models as voxel model from disk and resize voxel models (VM)

```
[V,vs]=VMreaddicomdir('/Volumes/LUETH-WIN/WIN AIM Matlab Libraries/VLFL-Lib/AIM_DICOMFILES'
);
vs
[a,as]=VMresize(V,[0.5 0.5 0.5],vs);
as
[a,as]=VMresize(V,vs,vs);
as
```

```
VMreaddicomdir: Analyzing 129 entries in path: /Volumes/LUETH-WIN/WIN AIM Matlab Libraries/
VLFL-Lib/AIM_DICOMFILES
.....................................................................................
.....................................
Stack of 126 DICOM images read from disk.

vs =

    0.4219    0.4219    1.0000
```

```
VMresize: Resize voxel image [512 512 126] to [256 256 63] with voxel size [0.84mm 0.84mm 2
.00mm]

as =

    0.8438    0.8438    2.0000

VMresize: Resize voxel image [512 512 126] to [216 216 126] with voxel size [1.00mm 1.00mm
1.00mm]

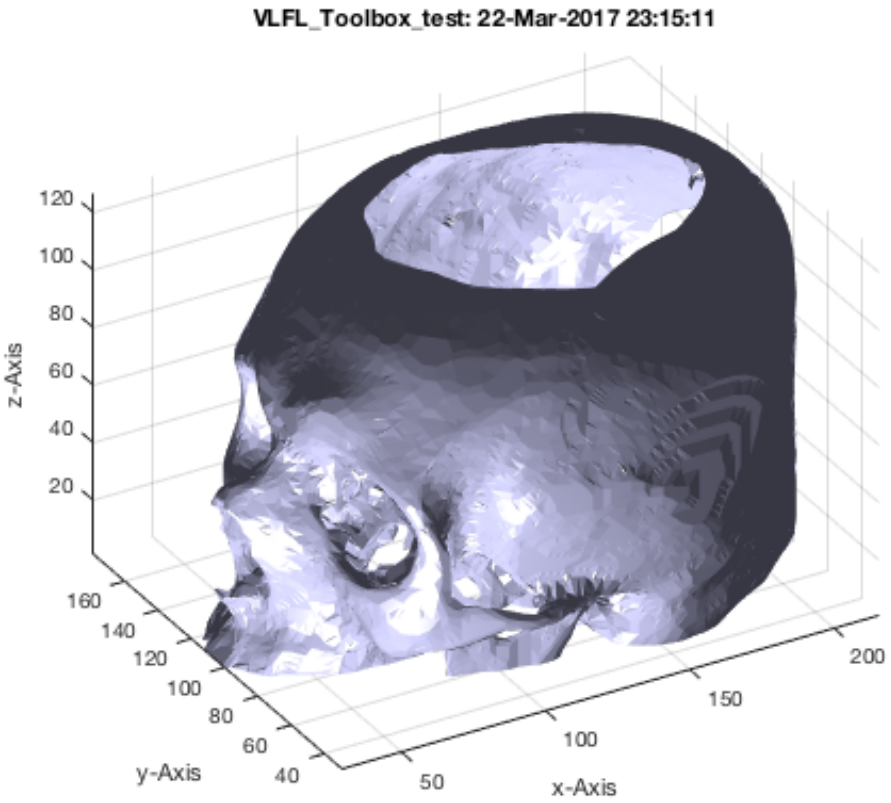
as =

     1     1     1
```

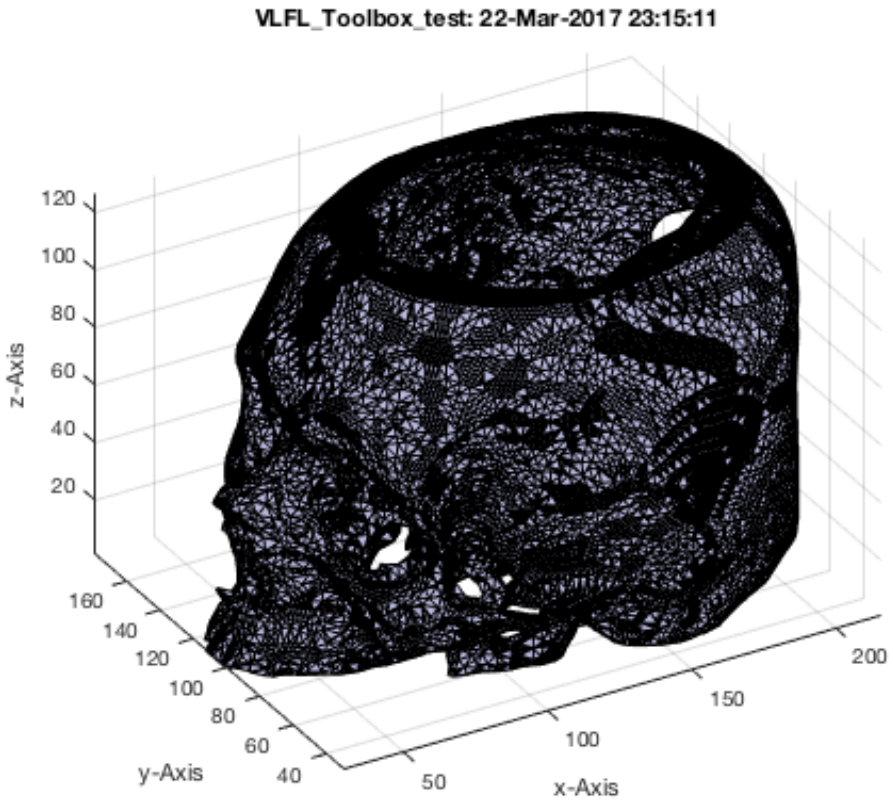
2. Solid skull bone reconstruction using SGofVMdelaunay

```
SG1=SGofVMdelaunay(a>1400,as);       % Takes about 30 seconds
SGfigure; VLFLplotlight (1,1); view(-30,30);
SGplot(SG1,'w');VLFLplotlight(1,1)
```

```
Elapsed time is 6.261116 seconds.
Elapsed time is 21.386026 seconds.
```



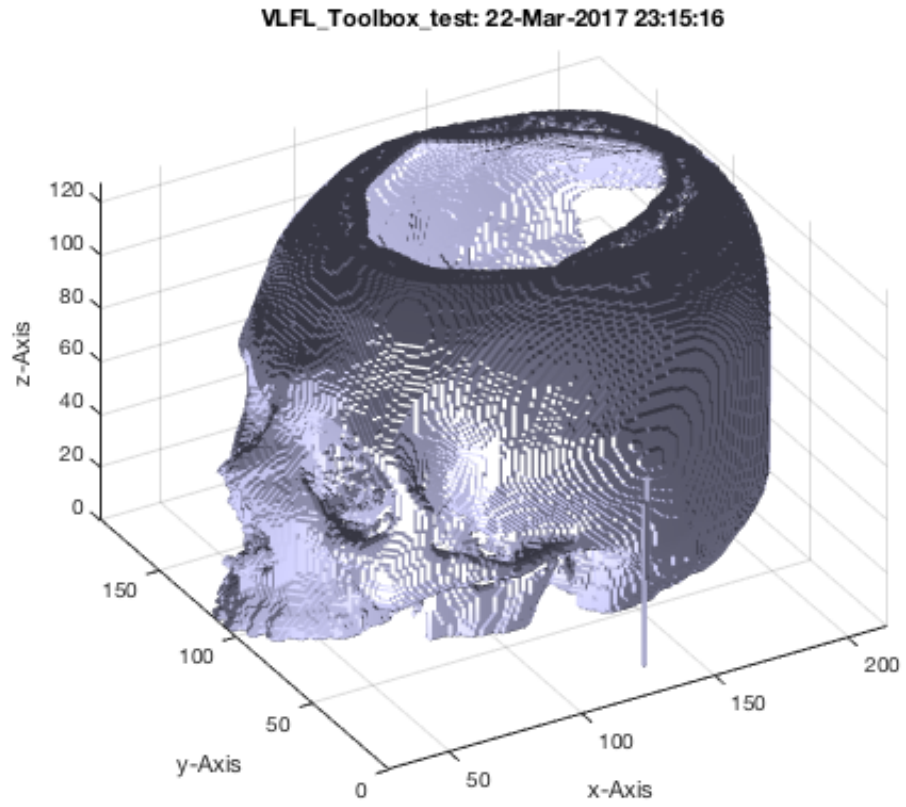
```
VLFLplotlight(0,1);
```



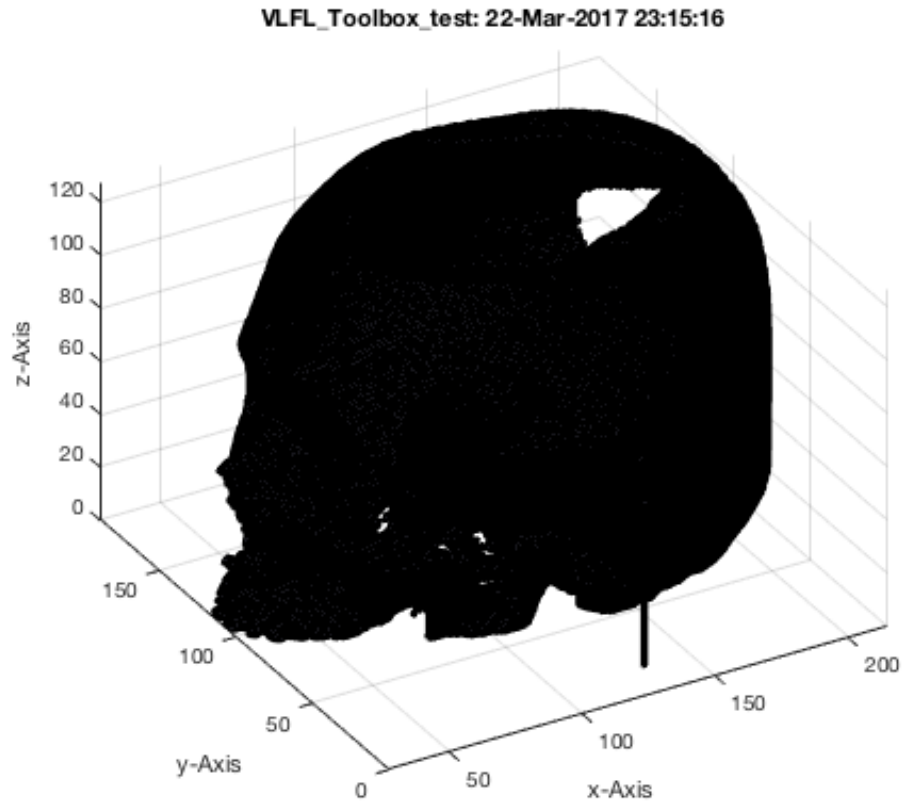
3. Solid skull bone reconstruction using SGofVMmarchcub

```
tic
SG2=SGofVMmarchcube(a>1400,as);       % Takes about 2 seconds
toc
SGfigure; VLFLplotlight (1,1); view(-30,30);
SGplot(SG2,'w');VLFLplotlight(1,1)
```

```
Elapsed time is 2.443624 seconds.
```

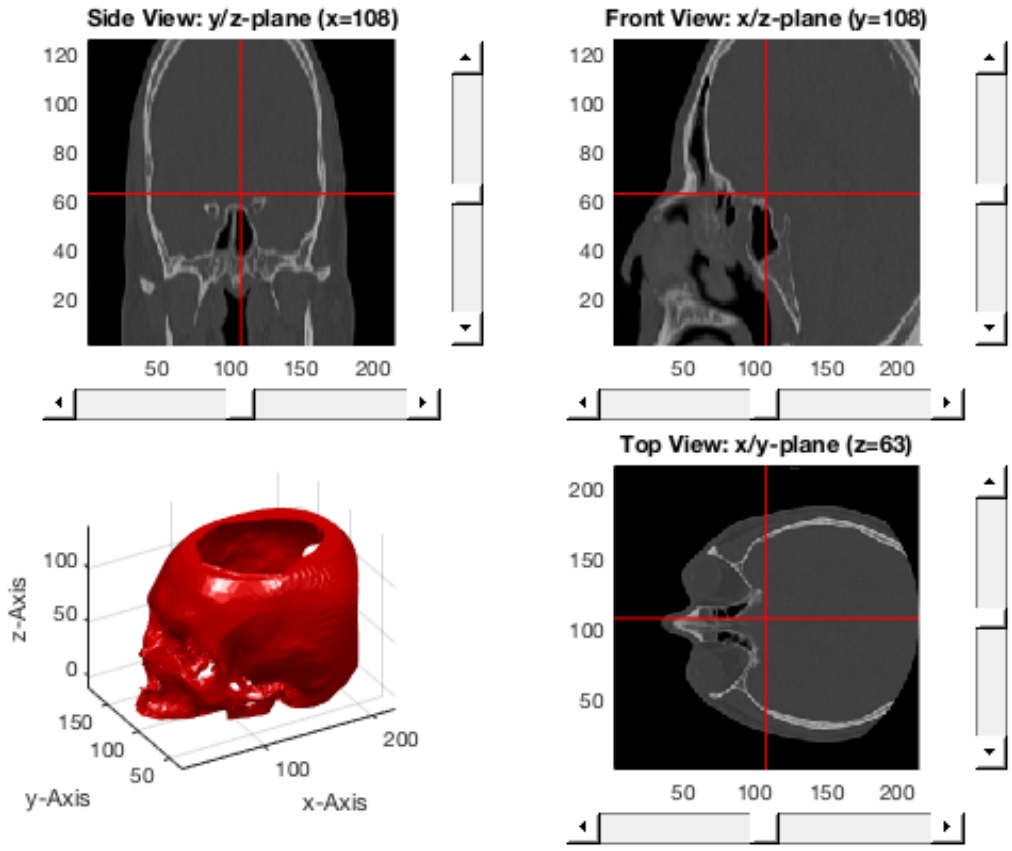



```
VLFLplotlight(0,1);
```



4 Show the Voxel model in quadrant 1-2-4 and surface model in quadrant 3

```
VMplot(a,'',SG1)
```



5. Create a surface model and convert it into a Voxel model

```
SGsample(17); VLFLplotlight(1,1);

[VM,ms,SG3]=VMofSG(SGsample(17),[128 128 128],true);
whos VM
ms
SG3
```

```
VMofSG: 5% 10% 15% 20% 25% 30% 35% 40% 45% 50% 55% 60% 65% 70% 75% 80% 85% 90% 95% 100%
  Name         Size                     Bytes  Class     Attributes

  VM         128x128x128             16777216  double

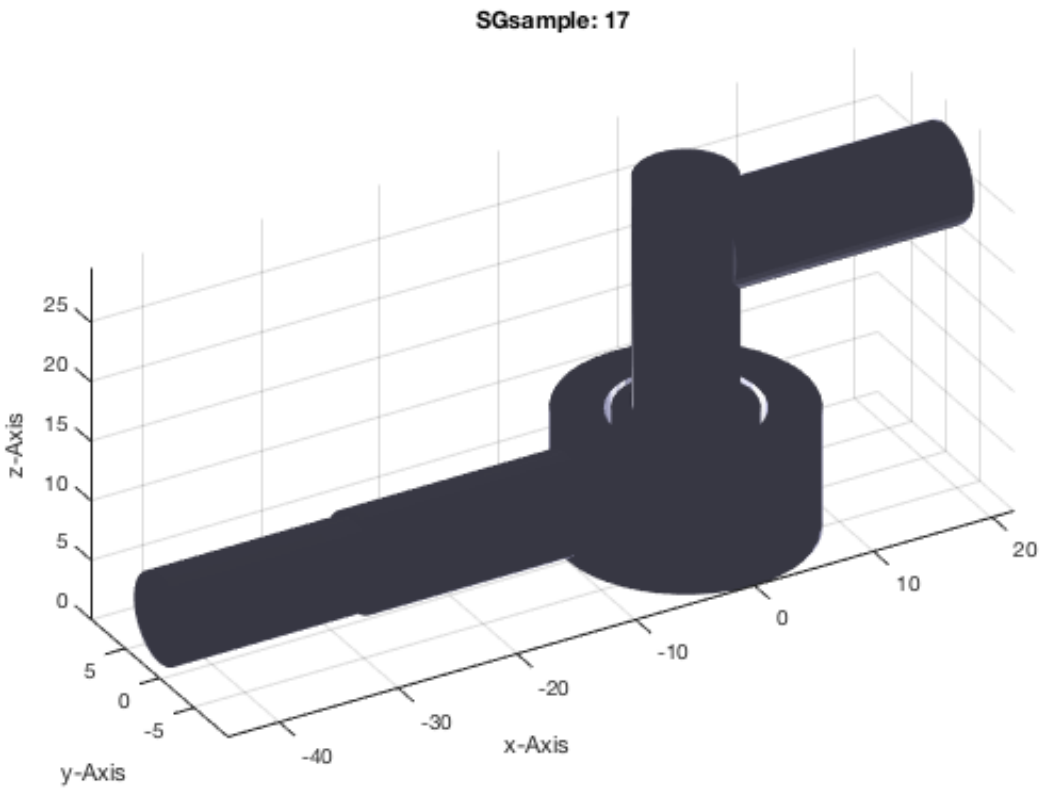

ms =

    0.5309    0.5309    0.2358


SG3 =

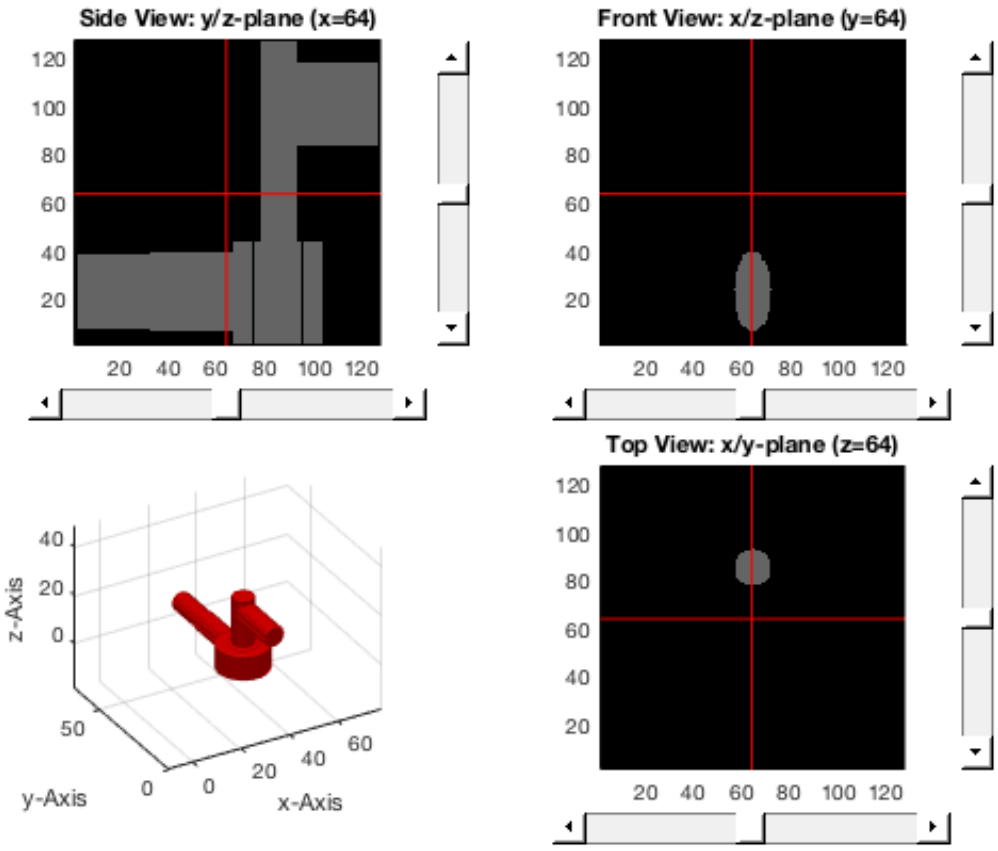
  struct with fields:

    VL: [20614×3 double]
    FL: [60580×3 double]
```



6. Plot the surface model in 4 quadrant plot

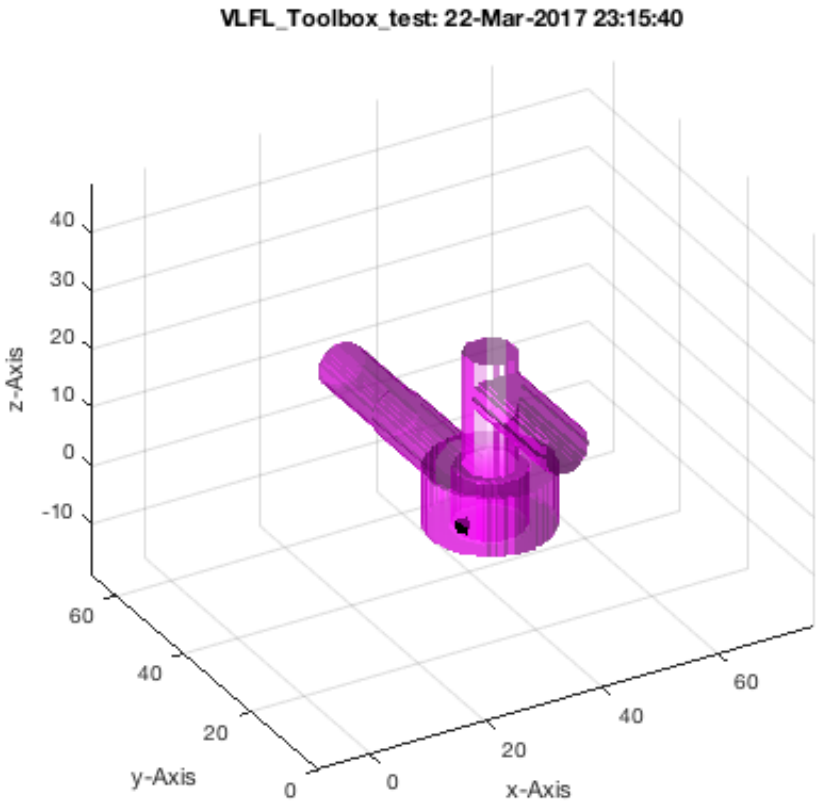
```
VMplot(VM,'',SG3)
```



7. Write the surface model as DICOM directory on disk

8. Select Point in 3D

```
SGfigure(SG3); view(-30,30); VLFLplotlight (1,.5);
ginput(1); p=select3d; pplot(p,'k*',4); rotate3d on
```



Final Remarks

```
close all
VLFLlicense
```

```
This VLFL-Lib, Rel. 3.6 (2017-Mar-23), is for limited non commercial educational use only!
Licensee: 3.6 Tim Lueth for academic and education 05-Mar-2017 !
Please contact Tim Lueth, Professor at TU Munich, Germany!
WARNING: This VLFL-Lib (Rel. 3.6) license will exceed at 18-Mar-2018!
Executed 22-Mar-2017 23:16:09 by 'lueth' using Matlab 9.1.0.441655 (R2016b) on a MACI64
======================================== Used Matlab products: ============================
==========
image_toolbox
map_toolbox
matlab
robotics_system_toolbox
simmechanics
simscape
simulink
================================================================================================
==========
```

Published with MATLAB® R2016b