

Readme

The VTK version installed is 9.1.0

Prequirements

1. Python3.9

Setup Environment

The following commands are based on Windows 10.

1. Clone the project. `git clone https://github.com/ziyunxiao/mm804_assignment2.git`
2. cd project folder `cd mm804_assignment2`
3. Create virtual environment `Python -m venv .venv`
4. Activate virtual environment `. .venv/bin/`
5. Upgrade pip `pip install --upgrade pip`
6. Install required library `pip install -r requirements.txt`

Run the code

1. run `python mm804_assignmet2.ipynb` .
2. You can then use mouse to rotate the model and watch it from different angle.
3. You can zoom in and out with the mouse middle wheel.
4. You need close the popup window in order to end the program.

View the report

please read `mm804_assignmet2.pdf` or from the output of `mm804_assignmet2.ipynb`

1. detail info about the model

The model is dolphin, from Microsoft Pain3d free library.

The file name is `dolphin.stl`.

The file size is `776,584` bytes

Number Of Points: 7767

Number Of Cells: 15530

2. Number of each part

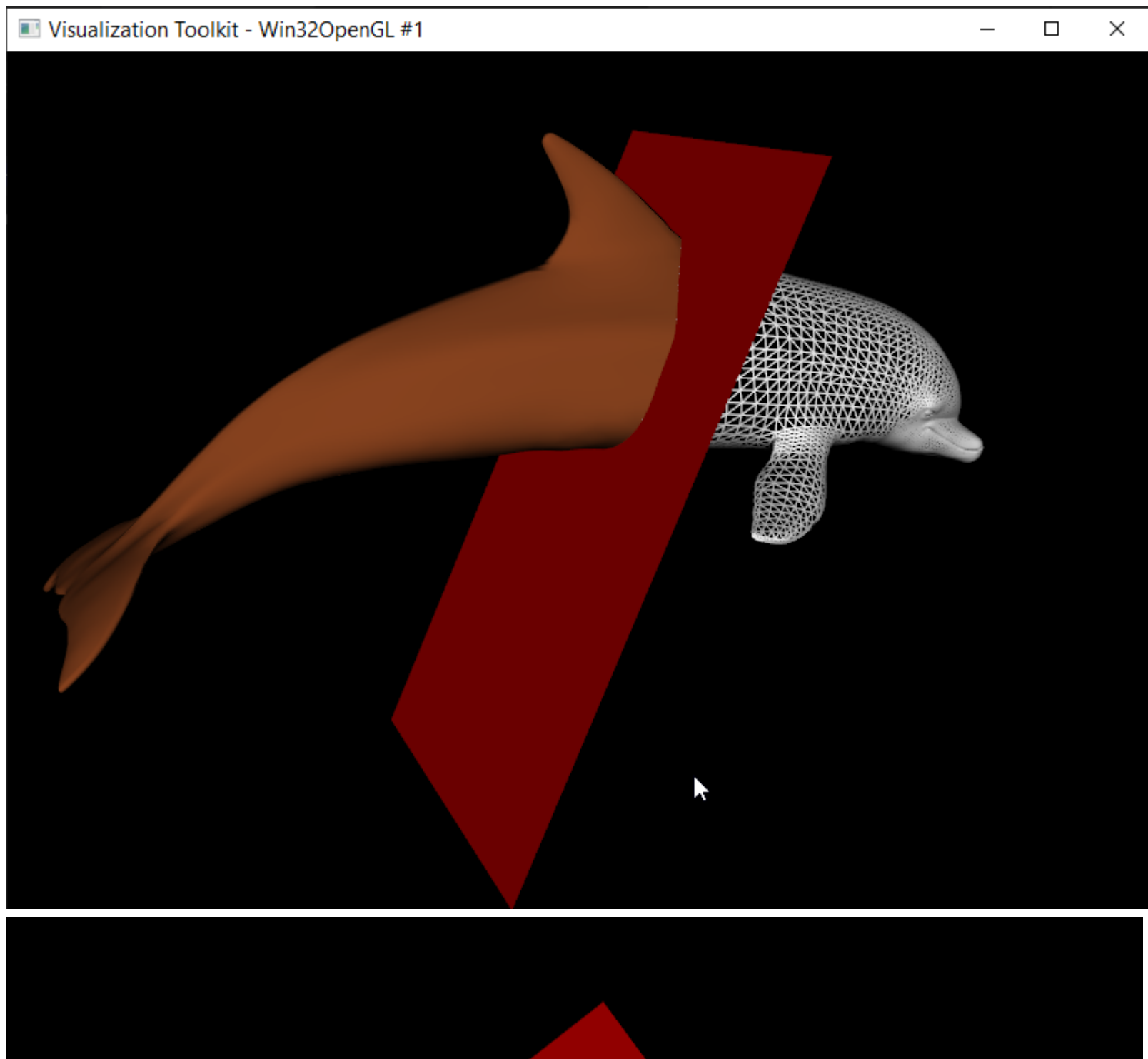
Number of points on the original object = 7767

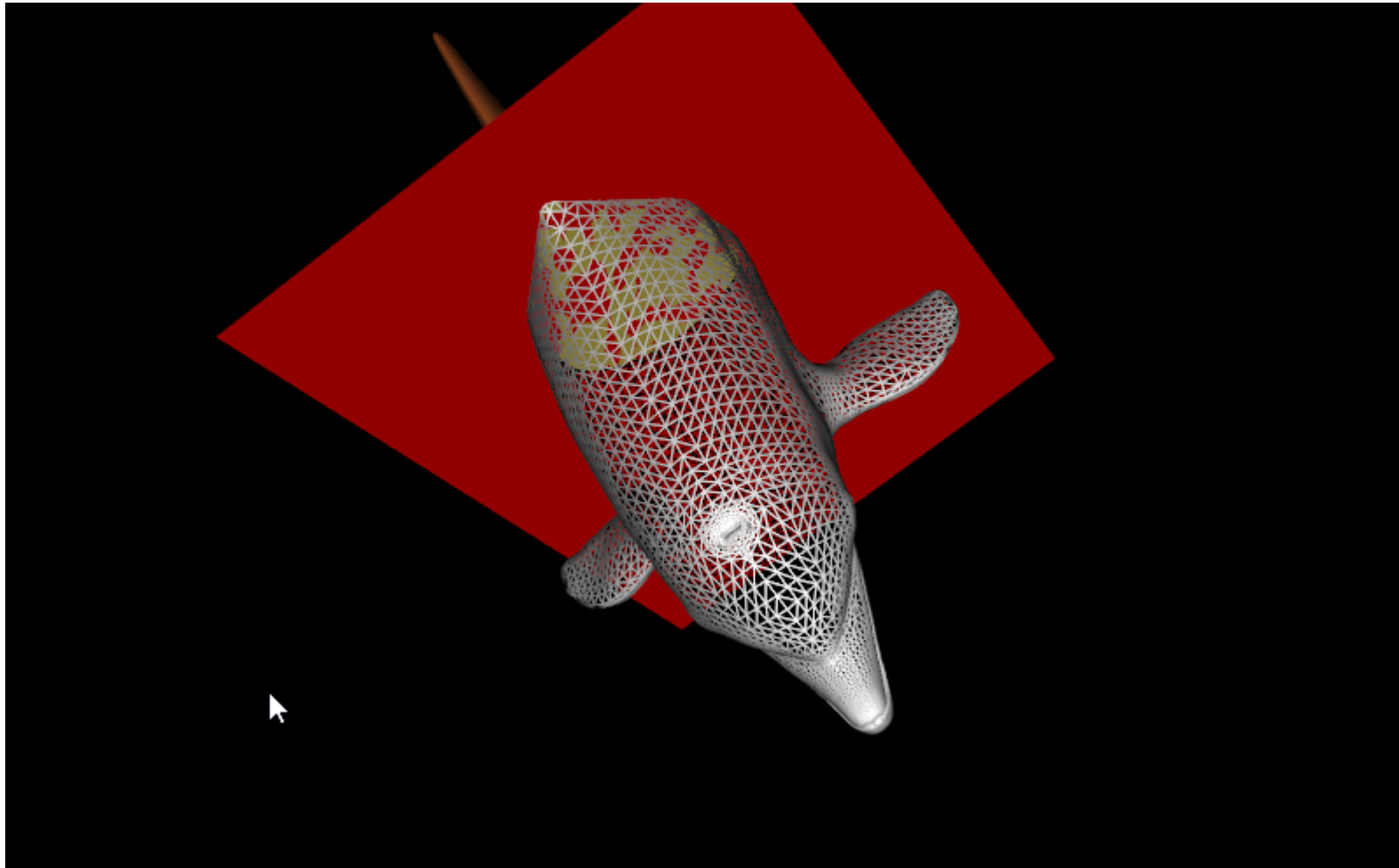
Number of points on the remaining part = 4614

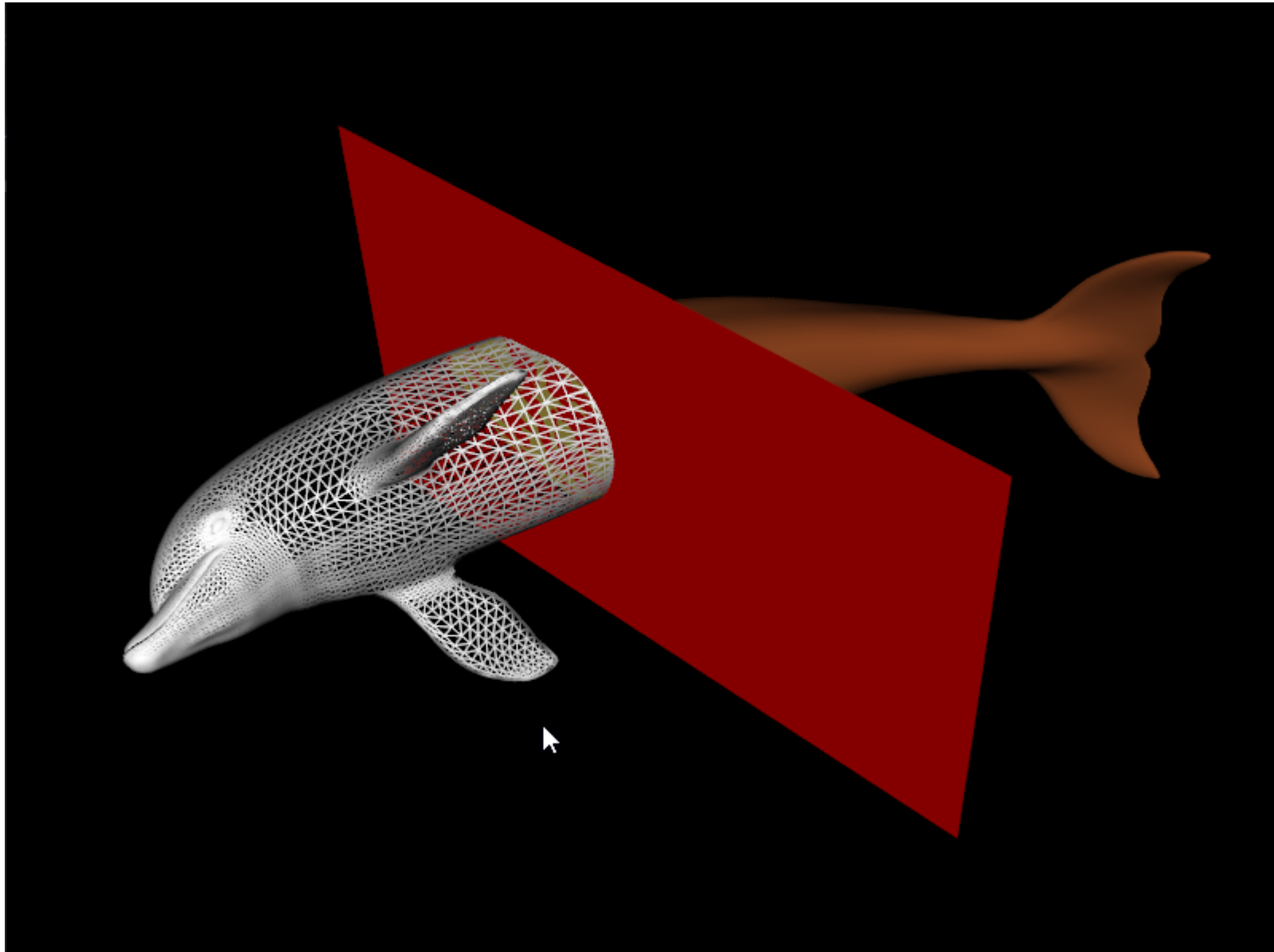
Number of points on the clipped part = 3399

Number of points on the sliced area = 123

3. Screenshots







4. Source code

The source is under this section.

In []:

```
## Install vtk

# Install VTK to virtual environment as needed

# windows
! pip install https://www.vtk.org/files/release/9.1/vtk-9.1.0-cp39-cp39-win_amd64.whl

# Linux
# ! pip install https://www.vtk.org/files/release/9.1/vtk-9.1.0-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
```

Collecting vtk==9.1.0

Using cached https://www.vtk.org/files/release/9.1/vtk-9.1.0-cp39-cp39-win_amd64.whl (44.7 MB)

Requirement already satisfied: wslink>=1.0.4 in c:\java\python39\lib\site-packages (from vtk==9.1.0) (1.3.1)

Requirement already satisfied: matplotlib>=2.0.0 in c:\java\python39\lib\site-packages (from vtk==9.1.0) (3.5.1)

Requirement already satisfied: numpy>=1.17 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (1.22.0)

Requirement already satisfied: pyparsing>=2.2.1 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (3.0.6)

Requirement already satisfied: fonttools>=4.22.0 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (4.28.5)

Requirement already satisfied: pillow>=6.2.0 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (9.0.0)

Requirement already satisfied: cycler>=0.10 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (0.11.0)

Requirement already satisfied: packaging>=20.0 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (21.3)

Requirement already satisfied: kiwisolver>=1.0.1 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (1.3.2)

Requirement already satisfied: python-dateutil>=2.7 in c:\java\python39\lib\site-packages (from matplotlib>=2.0.0->vtk==9.1.0) (2.8.2)

Requirement already satisfied: aiohttp in c:\java\python39\lib\site-packages (from wslink>=1.0.4->vtk==9.1.0) (3.8.1)

Requirement already satisfied: six>=1.5 in c:\java\python39\lib\site-packages (from python-dateutil>=2.7->matplotlib>=2.0.0->vtk==9.1.0) (1.16.0)

Requirement already satisfied: attrs>=17.3.0 in c:\java\python39\lib\site-packages (from aiohttp->wslink>=1.0.4->vtk==9.1.0) (21.4.0)

Requirement already satisfied: frozenlist>=1.1.1 in c:\java\python39\lib\site-packages (from aiohttp->wslink>=1.0.4->vtk==9.1.0) (1.2.0)

Requirement already satisfied: aiosignal>=1.1.2 in c:\java\python39\lib\site-packages (from aiohttp->wslink>=1.0.4->vtk==9.1.0) (1.2.0)

Requirement already satisfied: yarl<2.0,>=1.0 in c:\java\python39\lib\site-packages (from aiohttp->wslink>=1.0.4->vtk==9.1.0) (1.7.2)

Requirement already satisfied: charset-normalizer<3.0,>=2.0 in c:\java\python39\lib\site-packages (from aiohttp->wslink>=1.0.4->vtk==9.1.0) (2.0.10)

Requirement already satisfied: async-timeout<5.0,>=4.0.0a3 in c:\java\python39\lib\site-packages (from aiohttp->wslink>=1.0.4->vtk==9.1.0) (4.0.2)

Requirement already satisfied: multidict<7.0,>=4.5 in c:\java\python39\lib\site-packages (from aiohttp->wslink>=1.0.4->vtk==9.1.0) (5.2.0)

Requirement already satisfied: idna>=2.0 in c:\java\python39\lib\site-packages (from yarl<2.0,>=1.0->aiohttp->wslink>=1.0.4->vtk==9.1.0) (3.3)

In []:

```
import vtk
import sys

print(f"Python Version: {sys.version}")
print(vtk.vtkVersion().GetVTKSourceVersion())
```


Python Version: 3.9.9 (tags/v3.9.9:ccb0e6a, Nov 15 2021, 18:08:50) [MSC v.1929 64 bit (AMD64)]
vtk version 9.1.0

In []:

```
## Code for assignment2

from matplotlib.pyplot import plot_date
import numpy as np
import vtk
from vtk.util.colors import brown_ochre, tomato, banana
from vtkmodules.vtkInteractionStyle import vtkInteractorStyleTrackballCamera

# 1. Read model
reader = vtk.vtkSTLReader()
reader.SetFileName("dolphin.stl")

normals = vtk.vtkPolyDataNormals()
normals.SetInputConnection(reader.GetOutputPort())
normals.Update()

print("Number of points on the original object =", reader.GetOutput().GetNumberOfPoints())

# 2. set Plane center to model center
center = reader.GetOutput().GetCenter()

plane = vtk.vtkPlane()
# plane.SetOrigin(0, 0, 0)
plane.SetOrigin(center)
plane.SetNormal(-1, 0, 0.2)

clipper = vtk.vtkClipPolyData()
clipper.SetInputConnection(normals.GetOutputPort())
clipper.SetClipFunction(plane)
clipper.GenerateClipScalarsOn()
clipper.GenerateClippedOutputOff()
clipper.SetValue(0)

# 3. set clipper value
clipper.SetInsideOut(1)
clipper.Update()
print(
    "Number of points on the remaining part =", clipper.GetOutput().GetNumberOfPoints()
)
```

```
clipper.SetInsideOut(0)
clipper.Update()
print("Number of points on the clipped part =", clipper.GetOutput().GetNumberOfPoints())

clipper.GenerateClippedOutputOn() # turn this on for getting both parts

clipMapper = vtk.vtkPolyDataMapper()
clip_data = clipper.GetOutput()
# clipMapper.SetInputData(clip_data)
clipMapper.SetInputConnection(clipper.GetOutputPort())
clipMapper.ScalarVisibilityOff()
backProp = vtk.vtkProperty()
backProp.SetDiffuseColor(tomato)
clipActor = vtk.vtkActor()
clipActor.SetMapper(clipMapper)
clipActor.GetProperty().SetColor(brown_ochre)
clipActor.SetBackfaceProperty(backProp)

# 4 cutter
cutEdges = vtk.vtkCutter()
cutEdges.SetInputConnection(normals.GetOutputPort())
cutEdges.SetCutFunction(plane)
cutEdges.GenerateCutScalarsOn()
cutEdges.SetValue(0, 0)
cutStrips = vtk.vtkStripper()
cutStrips.SetInputConnection(cutEdges.GetOutputPort())
cutStrips.Update()

print(
    "Number of points on the sliced area =", cutStrips.GetOutput().GetNumberOfPoints()
)

cutPoly = vtk.vtkPolyData()
cutPoly.SetPoints(cutStrips.GetOutput().GetPoints())
cutPoly.SetPolys(cutStrips.GetOutput().GetLines())

cutTriangles = vtk.vtkTriangleFilter()
cutTriangles.SetInputData(cutPoly)
cutMapper = vtk.vtkPolyDataMapper()
cutMapper.SetInputData(cutPoly)
cut_output = cutTriangles.GetOutput()
# cutMapper.SetInputConnection(cutTriangles.GetOutputPort())
cutMapper.SetInputData(cut_output)
cutActor = vtk.vtkActor()
```

```
cutActor.SetMapper(cutMapper)
cutActor.GetProperty().SetColor(banana)
cutActor.VisibilityOn()

#
restMapper = vtk.vtkPolyDataMapper()
rest_data = clipper.GetClippedOutput()
# restMapper.SetInputData(rest_data)
restMapper.SetInputData(clipper.GetClippedOutput())
restMapper.ScalarVisibilityOff()
restActor = vtk.vtkActor()
restActor.SetMapper(restMapper)
restActor.GetProperty().SetRepresentationToWireframe()

#create renderers and add actors of plane and cube

ren = vtk.vtkRenderer()
renWin = vtk.vtkRenderWindow()
renWin.AddRenderer(ren)
renWin.SetSize(800, 600)
iren = vtk.vtkRenderWindowInteractor()
iren.SetRenderWindow(renWin)

style = vtkInteractorStyleTrackballCamera()
iren.SetInteractorStyle(style)

# 5. display plane
#Sample the plane
sampler = vtk.vtkSampleFunction()
sampler.SetImplicitFunction(plane)

#Set the bounds to be slightly larger
meshBounds = reader.GetOutput().GetBounds()
planeBounds = [0, 0, 0, 0, 0, 0]
for i in range(0,3):
    length = 1.2*(meshBounds[2*i+1] - center[i])
    planeBounds[2*i] = center[i] - length
    planeBounds[2*i+1] = center[i] + length

sampler.SetModelBounds(planeBounds)
sampler.ComputeNormalsOff()
sampler.Update()
```

```
#Extract the isosurface at 0
contour = vtk.vtkContourFilter()
contour.SetInputData(sampler.GetOutput())
contour.SetValue(0,0.)
contour.Update()

planeMapper = vtk.vtkPolyDataMapper()
planeMapper.SetInputConnection(contour.GetOutputPort())
planeActor = vtk.vtkActor()
planeActor.SetMapper(planeMapper)
planeActor.GetProperty().SetColor(banana)
planeActor.VisibilityOn()

# display data
ren.AddActor(clipActor)
ren.AddActor(cutActor)
ren.AddActor(restActor)
ren.AddActor(planeActor)

iren.Initialize()
renWin.Render()

iren.Start()
```

Number of points on the original object = 7767
Number of points on the remaining part = 4614
Number of points on the clipped part = 3399
Number of points on the sliced area = 123