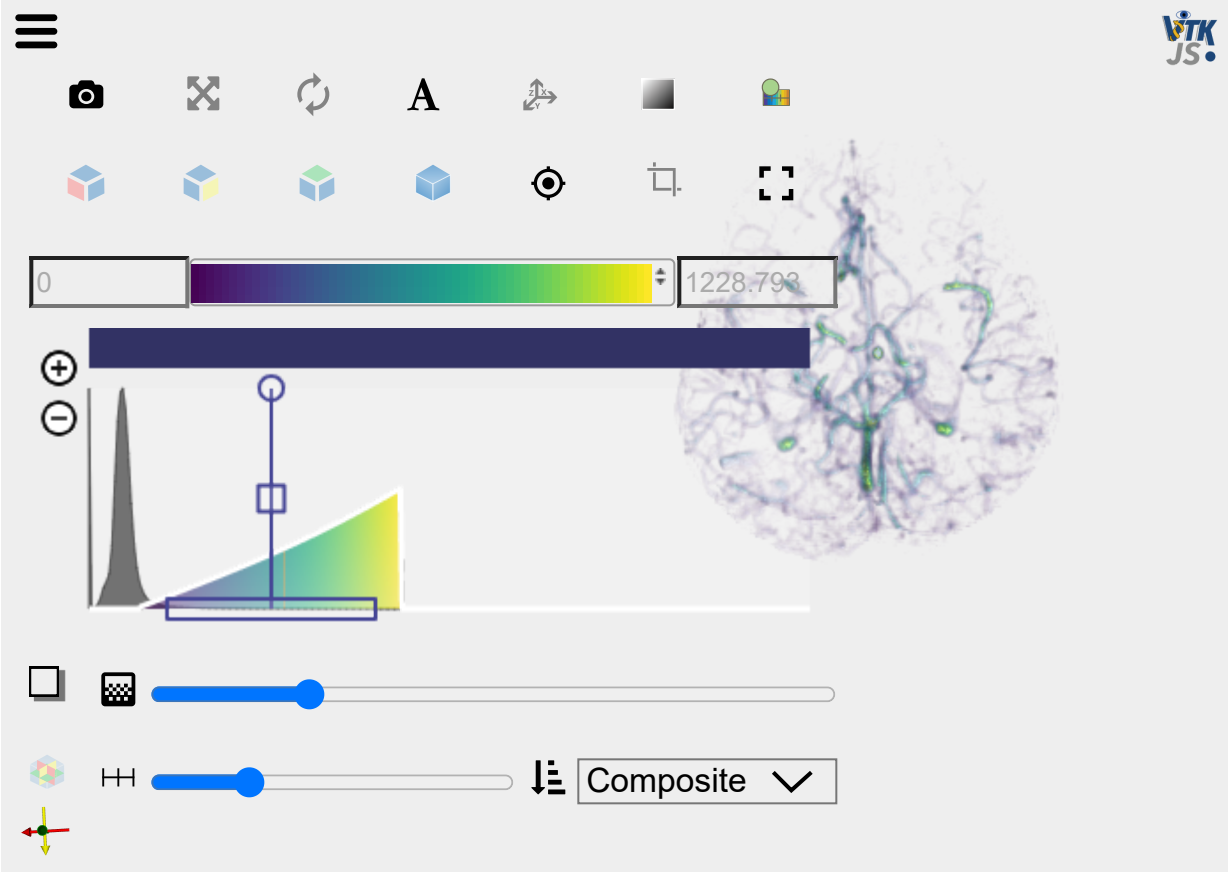
This notebook is intended to demonstrate how to extract vessels from a CTA image that contains only brain data (skull has been stripped).

In [1]:
```
import itk
from itk import TubeTK as ttk

from itkwidgets import view

import numpy as np
```

In [2]:
```
InputBaseDir = "../Data/CTA-Head/"

CTAFilename = InputBaseDir + "CTA.mha"
CTABrainFilename = InputBaseDir + "CTA-Brain.mha"

imMax = itk.imread(CTAFilename, itk.F)
imBrain = itk.imread(CTABrainFilename, itk.F)
```

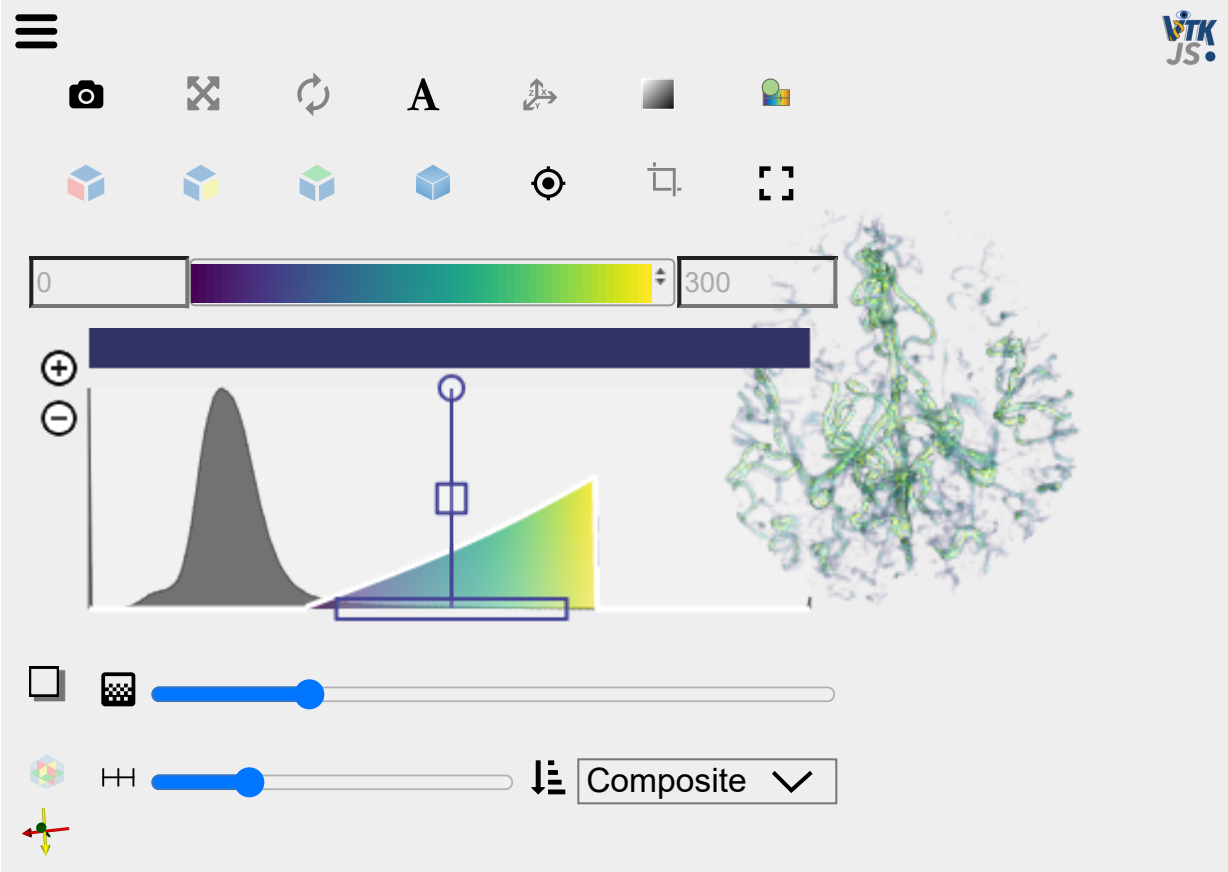
In [3]:
```
view(imBrain)
```



30

In [4]:
```
ImageType = itk.Image[itk.F, 3]

imMath = ttk.ImageMath.New(Input=imBrain)
imMath.Threshold( 0.00001, 4000, 1, 0)
imMath.Erode(10,1,0)
imBrainMaskErode = imMath.GetOutput()
imMath.SetInput(imBrain)
imMath.IntensityWindow(0,300,0,300)
imMath.ReplaceValuesOutsideMaskRange(imBrainMaskErode,0.5,1.5,0)
imBrainErode = imMath.GetOutput()
```

In [5]:
```
view(imBrainErode)
```



30

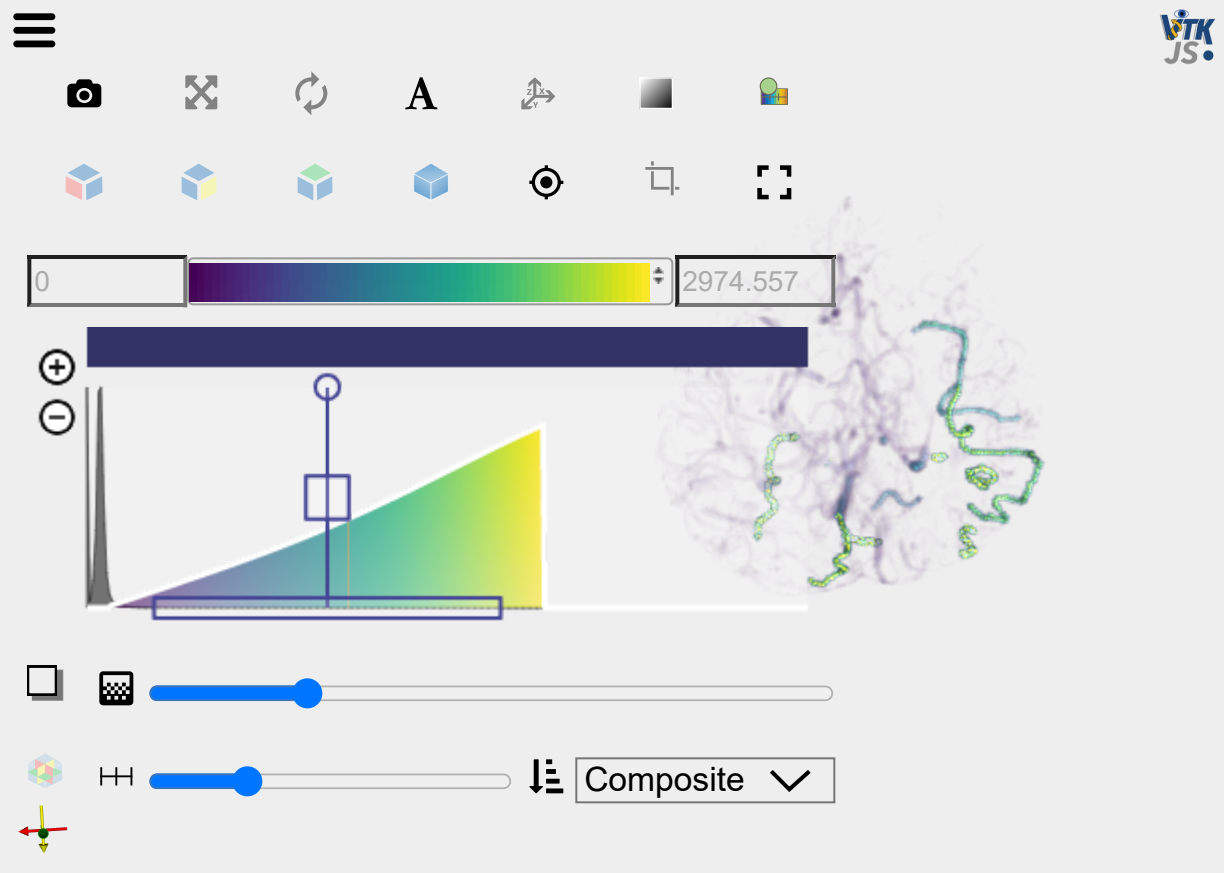
In [6]:
```
imMath = ttk.ImageMath[ImageType,ImageType].New()
imMath.SetInput(imBrainErode)
imMath.Blur(1.5)
imBlur = imMath.GetOutput()
imBlurArray = itk.GetArrayViewFromImage(imBlur)

numSeeds = 15
seedCoverage = 20
seedCoord = np.zeros([numSeeds,3])
for i in range(numSeeds):
    seedCoord[i] = np.unravel_index(np.argmax(imBlurArray, axis=None), imBlurArra
    indx = [int(seedCoord[i][0]),int(seedCoord[i][1]),int(seedCoord[i][2])]
    minX = max(indx[0]-seedCoverage,0)
    maxX = max(indx[0]+seedCoverage,imBlurArray.shape[0])
    minY = max(indx[1]-seedCoverage,0)
    maxY = max(indx[1]+seedCoverage,imBlurArray.shape[1])
    minZ = max(indx[2]-seedCoverage,0)
    maxZ = max(indx[2]+seedCoverage,imBlurArray.shape[2])
    imBlurArray[minX:maxX,minY:maxY,minZ:maxZ]=0
    indx.reverse()
    seedCoord[:][i] = imBrain.TransformIndexToPhysicalPoint(indx)
print(seedCoord)
```

```
[[  23.97288513 -158.18325806  302.26499939]
 [  -4.23602295 -146.56782532  298.39318848]
 [ -33.55116272 -160.94883728  299.49942017]
 [ -25.80754089 -144.90847778  278.48101807]
 [ -52.35710144 -207.41056824  296.73384094]
 [   0.74201965 -167.03311157  264.1000061 ]
 [ -48.48529053 -193.02955627  305.03057861]
 [ -45.16659546 -175.88296509  317.75224304]
 [  -5.34225464 -133.29304504  307.24304199]
 [ -58.44137573 -155.97079468  292.86203003]
 [  -6.44848633 -155.41767883  264.1000061 ]
 [ -43.50724792 -140.48355103  327.70832825]
 [  26.18534851 -133.29304504  285.1184082 ]
 [ -46.82594299 -157.63014221  276.82167053]
 [ -18.61703491 -128.86811829  290.64956665]]
```

In [7]:
```
# Manually extract a few vessels to form an image-specific training set
vSeg = ttk.SegmentTubes.New(Input=imBrain)
vSeg.SetVerbose(True)
vSeg.SetMinRoundness(0.4)
vSeg.SetMinCurvature(0.002)
vSeg.SetRadiusInObjectSpace( 1 )
for i in range(numSeeds):
    print("**** Processing seed " + str(i) + " : " + str(seedCoord[i]))
    vSeg.ExtractTubeInObjectSpace( seedCoord[i], i )

tubeMaskImage = vSeg.GetTubeMaskImage()
```

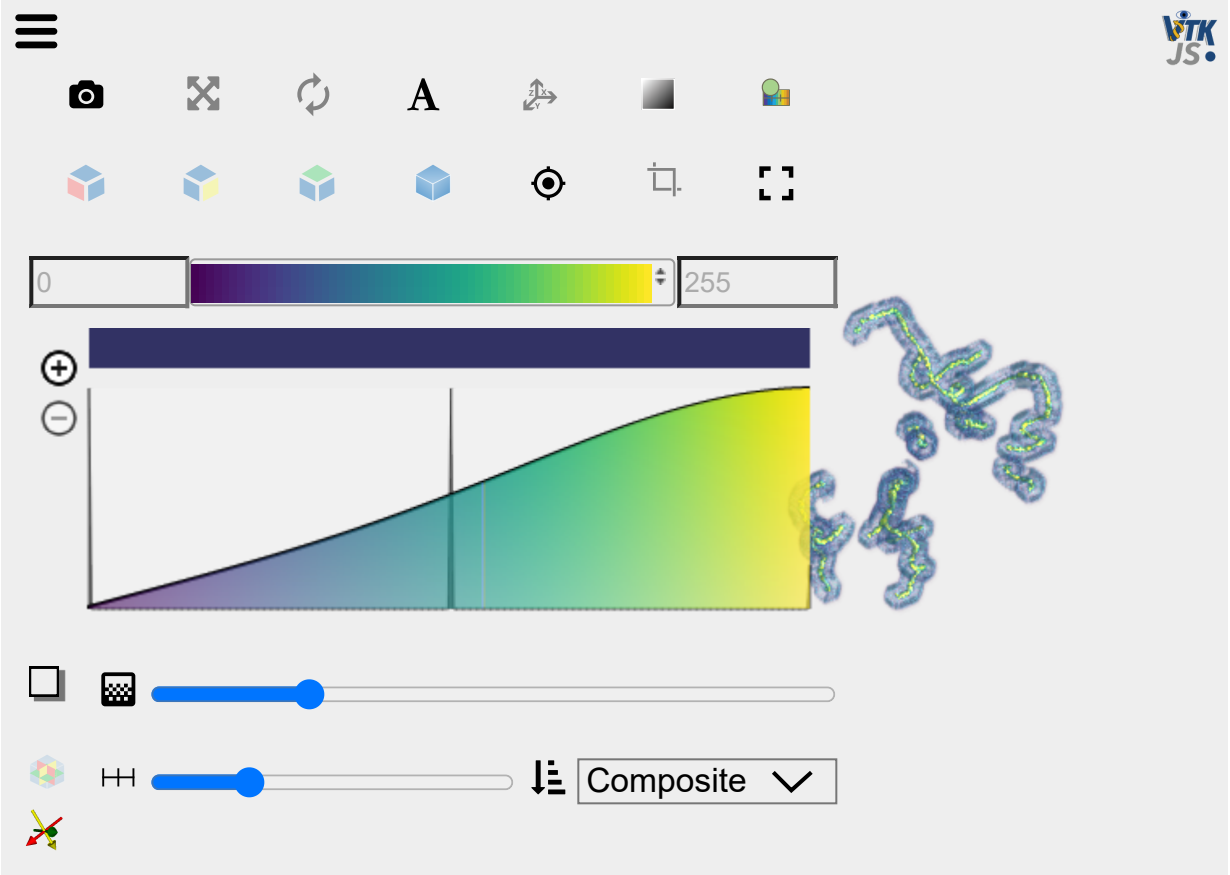
```
**** Processing seed 0 : [  23.97288513 -158.18325806  302.26499939]
**** Processing seed 1 : [  -4.23602295 -146.56782532  298.39318848]
**** Processing seed 2 : [ -33.55116272 -160.94883728  299.49942017]
**** Processing seed 3 : [ -25.80754089 -144.90847778  278.48101807]
**** Processing seed 4 : [ -52.35710144 -207.41056824  296.73384094]
**** Processing seed 5 : [   0.74201965 -167.03311157  264.1000061 ]
**** Processing seed 6 : [ -48.48529053 -193.02955627  305.03057861]
**** Processing seed 7 : [ -45.16659546 -175.88296509  317.75224304]
**** Processing seed 8 : [  -5.34225464 -133.29304504  307.24304199]
**** Processing seed 9 : [ -58.44137573 -155.97079468  292.86203003]
**** Processing seed 10 : [  -6.44848633 -155.41767883  264.1000061 ]
**** Processing seed 11 : [ -43.50724792 -140.48355103  327.70832825]
**** Processing seed 12 : [  26.18534851 -133.29304504  285.1184082 ]
**** Processing seed 13 : [ -46.82594299 -157.63014221  276.82167053]
**** Processing seed 14 : [ -18.61703491 -128.86811829  290.64956665]
```

In [8]:
```
imMath.SetInput(tubeMaskImage)
imMath.AddImages(imBrain, 200, 1)
blendIm = imMath.GetOutput()
view(blendIm)
```



In [9]:
```
LabelMapType = itk.Image[itk.UC,3]

trMask = ttk.ComputeTrainingMask[ImageType,LabelMapType].New()
trMask.SetInput( tubeMaskImage )
trMask.SetGap( 4 )
trMask.SetObjectWidth( 1 )
trMask.SetNotObjectWidth( 1 )
trMask.Update()
fgMask = trMask.GetOutput()
```

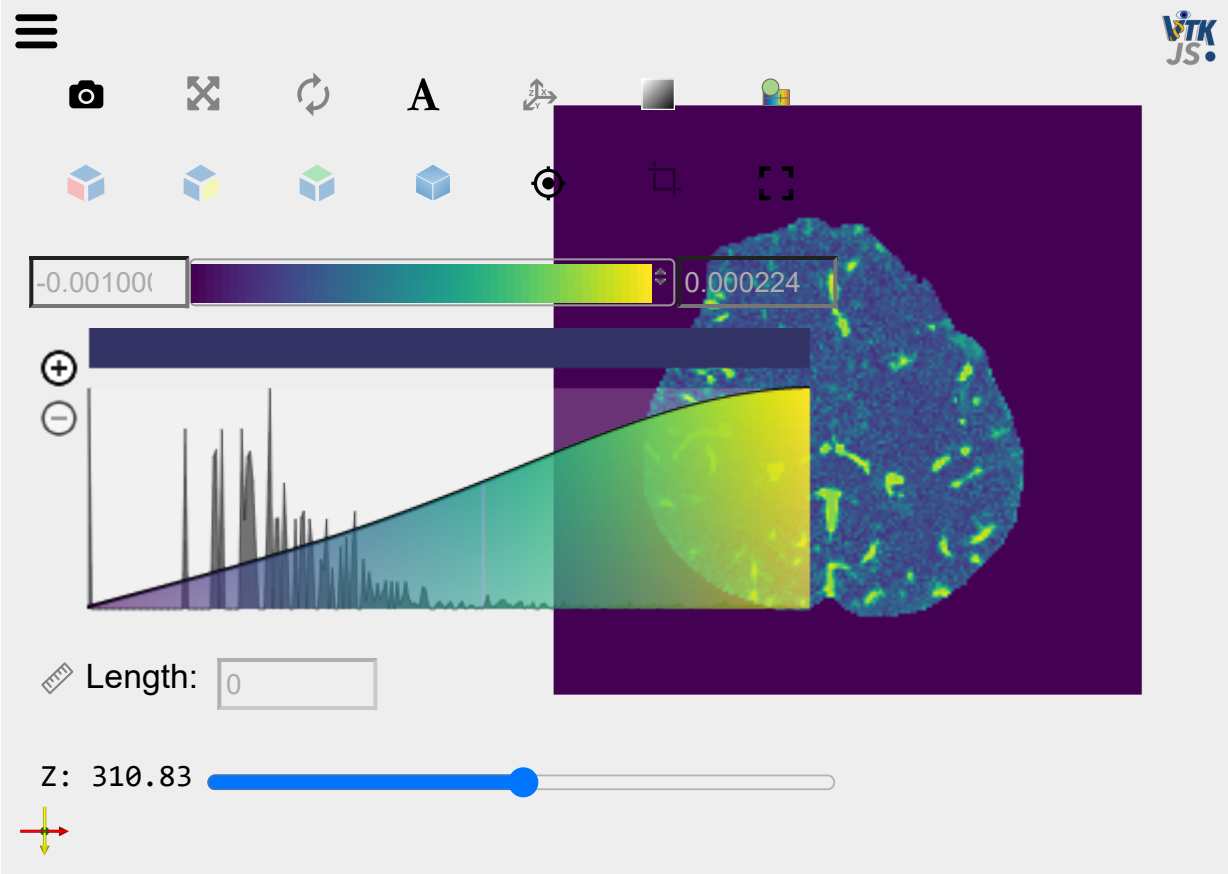
In [10]:
```
view(fgMask)
```



30

In [11]:
```
enhancer = ttk.EnhanceTubesUsingDiscriminantAnalysis[ImageType,LabelMapType].New(
enhancer.AddInput( imMax )
enhancer.SetLabelMap( fgMask )
enhancer.SetRidgeId( 255 )
enhancer.SetBackgroundId( 128 )
enhancer.SetUnknownId( 0 )
enhancer.SetTrainClassifier(True)
enhancer.SetUseIntensityOnly(True)
enhancer.SetScales([0.43,1.29,3.01])
enhancer.Update()
enhancer.ClassifyImages()
```

In [12]:
```
im1vess = itk.SubtractImageFilter( Input1=enhancer.GetClassProbabilityImage(0), I

imMath.SetInput(imBrain)
imMath.Threshold(0.0001,2000,1,0)
imMath.Erode(2,1,0)
imBrainE = imMath.GetOutput()

imMath.SetInput(im1vess)
imMath.ReplaceValuesOutsideMaskRange(imBrainE, 1, 1, -0.001)
im1vessBrain = imMath.GetOutput()
#view(enhancer.GetClassProbabilityImage(0))
view(im1vessBrain)
```



In [13]:
```
itk.imwrite( im1vess, InputBaseDir + "CTA-VesselEnhanced.mha", compression=True)

itk.imwrite( im1vessBrain, InputBaseDir + "CTA-Brain-VesselEnhanced.mha", compres
```

In [ ]:
```

```