

Exporting Qupath Labels

Owner: Susan Sheehan | Created at: January 25, 2021

Tags:

Description

This describes how to export Qupath labels,

Procedure

In Qupath, open the Scripting editor under the automate tab.

Then use the following code adjusting labels as needed use the run project command to run in bulk.

```
import qupath.lib.images.servers.LabeledImageServer

def imageData = getCurrentImageData()

// Define output path (relative to project)
def name = GeneralTools.getNameWithoutExtension(imageData.getServer().getMetadata().getName())
def pathOutput = buildFilePath(PROJECT_BASE_DIR, 'tiles', name)
mkdirs(pathOutput)

// Define output resolution
double requestedPixelSize = 10.0

// Convert to downsample
double downsample = requestedPixelSize / imageData.getServer().getPixelCalibration().getAveragedPixelSize()

// Create an ImageServer where the pixels are derived from annotations
def labelServer = new LabeledImageServer.Builder(imageData)
.backgroundLabel(0, ColorTools.WHITE) // Specify background label (usually 0 or 255)
.downsample(downsample) // Choose server resolution; this should match the resolution at which tiles are exported
.addLabel('Immune cells', 1) // Choose output labels (the order matters!)
.addLabel('Other', 2)
.addLabel('Positive',3)
```

```
.addLabel('Glomeruli',4)

.addLabel('BowmensSpace',5)

.multichannelOutput(true) // If true, each label is a different channel (required for multiclass probability)

.build()

// Create an exporter that requests corresponding tiles from the original & labeled image servers

new TileExporter(imageData)

.downsample(downsampling) // Define export resolution

.imageExtension('.jpg') // Define file extension for original pixels (often .tif, .jpg, '.png' or '.ome.tif')

.tileSize(512) // Define size of each tile, in pixels

.labeledServer(labelServer) // Define the labeled image server to use (i.e. the one we just built)

.annotatedTilesOnly(false) // If true, only export tiles if there is a (labeled) annotation present

.overlap(64) // Define overlap, in pixel units at the export resolution

.writeTiles(pathOutput) // Write tiles to the specified directory

print 'Done!'
```
