Simple bridges between 2D GIS & 3D visualization

For this activity, we’re going to explore one way you can bridge two worlds of geographical visualization: 2D desktop GIS & the 3D web of interactive visualization.

In particular, we’ll see how to use the QGIS2threejs ‘plugin’ for QGIS, which lets us take a 2D map design in regular QGIS, do some basic restyling for 3D, & export the results to a HTML/JavaScript viewer (or even export to the 3D object file format glTF that you can use to import objects into A-Frame, as in the sheep in your A-Frame activity). Similarly, remember that three.js is the underlying open-source visualization engine that A-Frame is making more accessible to you. So, with this activity, you’re learning more about how Desktop GIS, 2D cartography, 3D visualization, virtual reality, & associated data formats are all increasingly interconnected.

Many of you haven’t used QGIS, of course. Perhaps you’ve not used ArcGIS Pro or ArcMap either, or haven’t for a while. The interfaces can be complex, but a lot of that is for functionality that we don’t need right now, so walking you through it shouldn’t be too bad.

***Desktop GIS*** *is the general category of GIS programs that you may have used in the past for the manipulation, analysis, & visualization of spatial data on desktop computers. ESRI’s ArcMap has been the classic player in this space & is currently being replaced by ArcGIS Pro. The most widely used alternative at this point is QGIS, which is free & open source (FOSS) & runs on Windows, Mac, & Linux. Desktop GIS sometimes handles 3D data, but its core design & concepts arise from 2D.*

*QGIS offers* ***plugins****, which are ways to extend the basic functionality of the program. They are usually written & maintained by community members, often via GitHub.*

A relatively recent version of QGIS is already installed on the lab computers. Look for it—QGIS 3.16.10. Where is it? On some lab computers, you can’t Search for it in the Start menu. Instead, you need to use the Windows file explorer to go to *D:\Applinks* & double click on the ‘QGIS’ link—it may be underneath its own *D:\Applinks\QGIS 3.16.10* folder & the actual program link may be called “*QGIS Desktop…*”

It takes 20-30 seconds to start the first time, sometimes, & it’s not obvious it’s even running during that time. If it asks you whether you want to import your configuration from QGIS 2, there’s no need, unless you’ve been an active user of QGIS 2 on lab computers in the past.

*If you want to install it on your own system (Windows, Mac, Linux, etc.),* [*you can*](https://download.qgis.org/en/site/)*—and at least for this activity, you too might want to install an LTR (Long Term Release), such as QGIS 3.22.5 LTR. The “LTR” (Long Term Release) is generally the most stable available, even if some of the newest features aren’t there yet. Below, I’ve tested these instructions most thoroughly on the Windows 3.16.10 LTR version, as that is what is in the labs right now, though the other versions should be similar.*

*As you proceed, the user guide can be a helpful reference.* [*QGIS 3.16 User Guide*](https://docs.qgis.org/3.16/en/docs/user_manual/index.html)*.*

Regardless of how you’re getting access to QGIS, you’ll need to install the QGIS2threejs plugin. One way to do this is to:

1. Go to the Plugins menu & choose “Manage & Install Plugins…”
2. Search for *three* (among All plugins, which is the default). Choose Qgis2threejs.
3. Push the *Install Plugin* button.
4. Close out the plugin window when done. You should be back to the main QGIS interface

Some examples of what can be done:

* [European population in 3D](https://tjukanov.org/europe3d) – Topi Tjukanov
* Visualizing data where bathymetry matters – Liam Mason
* [Kauai](http://www.statsmapsnpix.com/2018/11/how-to-create-interactive-3d-landscape.html) – Alasdair Rae
* [Avalanche locations](http://www.statsmapsnpix.com/2016/12/interactive-terrain-mapping-in-qgis.html) – Alasdair Rae

# Our example: Visualizing UBC terrain, buildings, & trees.

We’ll take up our favourite example & visualize UBC campus using these new/different tools. If at any point below, you get lost in the interface, or want a more systematic introduction to the QGIS GUI, you may want to [check out this part of the QGIS help](https://docs.qgis.org/3.16/en/docs/user_manual/introduction/qgis_gui.html).

We will use the following files:

|  |  |  |
| --- | --- | --- |
| What | Downloadable file | Notes |
| Elevation data (DEM) | vancouver-area-dem.tif | I got 30 m SRTM DEM data [using a helpful map-based tool](http://dwtkns.com/srtm30m/) Rae recommended. I then stitched it using QGIS & trimmed it down. |
| Aerial imagery | UBC-2021-1m-lossy-geotiff-UTM10N.tif  (or if you want a slightly higher quality version later for clipping or because you have a computer with a lot of memory, you can choose this version.) | UBC has been getting 10 cm imagery regularly. Here’s [2021](https://abacus.library.ubc.ca/dataset.xhtml?persistentId=hdl:11272.1/AB2/R731P3). As some of your QGIS installations won’t have ECW file format support enabled by default, & as downloading all the GeoTIFFs & stitching them is unnecessary, & as we don’t need 10 cm imagery right now, I’ve prepared a simpler, smaller single file. |
| UBC Buildings | ubcv\_buildings\_records.geojson | Building data from <https://github.com/UBCGeodata> |
| UBC trees (selected) | Ubcv\_trees.geojson | We’ve used a version of this file. From <https://github.com/UBCGeodata> |

Download each of the four files above. Move them to their own folder in a place where they won’t get deleted (not the Desktop, remember!)

To load these files into QGIS, drag them from the Windows file explorer into the QGIS window.

After doing so, chances are, you’re not seeing all the data. Why? There’s a drawing order to the layers in the Layers box shown to the lower left, & it’s from the bottom to the top. So, here, I’m just seeing dark colours, because my DEM elevation layer is at the top & it’s relatively low elevation everywhere around UBC.

You can click & drag the layers to reorder them. Once you reorder them, you should be able to see some data.

Then a key step: let’s **save our projects**! Go to the *Project* menu, & choose *Save As*. **Save your project** regularly below!

If you need to zoom/pan the data, you can either right click on a layer & Zoom to Layer or use the tools on the toolbar.

You can see the coordinate system you’re using for the viewer in the lower right of the window.

If it isn’t EPSG:26910 (NAD83 / UTM10N), click on it, search for EPSG:26910, & set it. This doesn’t reproject any data, just set the projection we’re using in the viewer. We could choose a different projection & QGIS2threejs would be fine with it—it is a great advantage to not be confined to Web Mercator any longer—but this is good enough for what we’re doing here.

At this point, you probably have a map that looks roughly like this.

**Now, please note: the extent that is visible is the area that QGIS2threejs will default to rendering in 3D.** I’ll make things easier on my browser, by zooming in to a part of campus using the zoom tools shown above. To end up with a nice elevation effect in 3D, I’ll include the area down to the water. I also resize the QGIS application window to get a different shape. (*Remember this, as some of you may later have your software crash due to lack of resources in producing too large of a model; equally, you may find it is too big to store easily on GitHub, as there are file size limits. One solution can be to zoom in first here (& redo as needed)).*

The colour’s may not be what you like, though. Let’s change them. First, start with UBC Trees. Right click on the layer in the Layers area & select Styles. You can use the colour selector right away.

What about the buildings? You can totally change them the same way. But just for learning’s sake, let’s take the additional step of right clicking on the layer & choosing ‘Properties’, not ‘Styles’.

If you look to the left, we’re in the ‘Symbology’ tab at the moment, which is where the options we’re interested in are. But you should take a look at the other tabs before clicking back to Symbology. Once back, we’ll just click on the ‘Color’ & change the color of the buildings to something a bit less overwhelming. I’m choosing a light grey. When done, leave the properties window via the OK button.

At this point, you are ready to begin using the QGIS2threejs plugin to restyle the current map into 3D & export the results. You get to the plugin by choosing the ‘Web’ menu, then ‘Qgis2threejs’, then ‘Qgis2threejs Exporter’:

You should be looking at a two-paned window with your layers on the left & a (blue-skies) preview pane on the right. On some machines/versions (but not all), I’ve gotten this error in the preview pane, which has tended to mean that the preview window is only partially operational:

“Any 3D objects not rendered? There is a compatibility issue with QGIS 3D view. You need to close QGIS 3D view(s) & restart QGIS to use this preview.”

If you’re not getting that error, it’s not the end of the world. You’ll just have to use “File -> Export to Web” more often (more on that useful command in a minute!) & look at your results in the web browser when you need to see them.

Let’s get a first visualization going, then. Turn on the three layers we’re using by checking their boxes to the left.

Now, we’re ready to try looking at them exported. Go up to File -> Export to Web, & in the resulting dialog box, note that we could choose where to save the files (which you’ll need to do later when you turn this in, but for now, just let it go to a temporary directory that will be automatically cleaned up, as the defaults have). Let’s choose the Template, though. The ‘mobile’ option is intriguing, but for now, choose the ‘3D viewer with dat-gui panel’. You need to check “Enable the Viewer to Run Locally” or you will probably get an Error when it opens in the browser. Hit OK. After a few seconds of waiting, you may have to click on the link the plugin provides for the “Web page file” or you may get something like this in the browser that pops up.

Making progress! Use the mouse to click & drag & rotate the view. Notice there is a control panel on the upper right you can play with (open up options via those triangles.) You can also look at Help.

But quickly, it seems like we should be able to do better. Let’s change a few things!

**Why are the trees (& buildings) both underground & in crude raster form above-ground?** We can do better. First, the rendered 2D map is the default image being draped over the elevation DEM, which includes the 2D point visualizations for trees. We can change the image used to just be the aerial photography. Close the tab in the browser. Go back to the Qgis2threejs exporter window in QGIS (you can close the ‘Export to Web’ window though). *Right-click* on the ‘vancouver-area-dem’ & choose ‘Properties’. Under ‘Material’, click on the ‘map (canvas)’ & press the Red minus button to remove it—that’s the QGIS 2D map rendering we’re no longer going to be using. Instead, we’ll click on the green plus sign & ‘Select layer(s)’, where we will check the box for the UBC-2021-1m-lossy-geotiff image data & hit OK, then hit Apply. You should still be in the vancouver-area-dem Layer Properties window but you should now see the 3D image of the campus looking much better without the pixelated renderings of the trees & buildings.

**Can we make the image less blurry?** Also take a look at the other options in this Layer Properties window for the DEM for a minute. For example, increase the number of pixels in the ‘Image width’ up to 4096. This increases the detail at which the imagery is sampled (it’s not likely using the full resolution of your GeoTIFF, anyway), while reducing performance. Try it lower as well. Notice the difference.

Also notice if you choose to enable ‘Tiles’ (& hit ‘Apply’) then you get the full extent of all your data appearing, not just the initial view that we entered the plugin with. And the imagery is all much higher resolution (or hasn’t been downsampled as much). This can either be good or bad, depending on the application—it can easily be too much. And to really use this, you’d have to first trim all the data to the same extent… & choose that extent carefully. For now, turn tiles back off.

**Can we get rid of some of that stuff below the surface?** Notice that the left sides of these windows have little tabs youo can choose. Their labels are written vertically. We have been in ‘Main’ but there is also ‘Others’. Click on ‘Others’.

Choose ‘Build sides’, apply it, & suddenly your 3D landscape becomes a lot more substantial because the ‘sides’ of the surface get drawn.

It’s certainly not perfect yet. But we’re making progress. The [online documentation for Qgis2threejs](https://qgis2threejs.readthedocs.io/en/docs/Exporter.html) is relatively good. You can learn about what options in various parts of the plugin mean. Anyway, when ready, choose ‘OK’, & re-export to the web, opening it up in the web browser again. Now, at least the terrain is just the imagery.

…but our trees (& our buildings) are actually underground, if not sticking out of the image. At sea-level. This is unfortunate. We’ll move the buildings & trees above-ground soon enough. But let’s talk about the alignment issues where some of your buildings may be sticking out well beyond the edge of the aerial imagery, horizontally. This is caused by a recent design decision in the development of this plugin—earlier versions clipped polygon features at the same place DEMs/their imagery are clipped, but the current version draws any polygon that intersects with the visible DEM base. Later versions of the plugin may fix this issue too. But for now, let’s see if a bit of creative thinking can solve this problem… & perhaps even increase the expressive possibilities of the map along the way? Here are some ideas:

1. Sometimes, you’re able to minimize the visual problems by going to the Scene menu in Qgis2threejs Exporter, choosing Scene Settings, & changing the ‘Base Extent’—especially the ‘width’. You might be able to find a width that doesn’t have a polygon so visibly jutting beyond the base layer.
2. If you either zoom out in QGIS before opening up the Qgis2threejs plugin OR you go back & choose the ‘Tiles’ option I mentioned above… as long as your polygon layers have a narrower extent than your DEM data & its imagery, you may be able to trade your problems with ragged buildings for new problems With DEMs & images that are too wide (& not of the same extent themselves). Still, there are cases where this might work.
3. But really, the best thing to do here is to be deliberate about the view that you want your users to have & manually ‘clip’ any offending data layers to the same extent you want the viewers to see. In order to clip t he data, we need something to clip with. You may have a feature(s) that you want to use, but if not, I’ll show you how to make a circle instead of the usual rectangle. We want to make a new layer in QGIS outside of the Qgis2threejs. We go to the Layer menu & choose ‘Create Layer’—I recommend you choose making a GeoPackage as an open-source & modern alternative to Shapefiles. A window pops up where you have to:
   * Name the ‘Database’—that’s the GeoPackage filename, as GeoPackages are single files with multiple potential layers in them. I chose to call mine ‘clipping.gpkg’.
   * You will get an error unless you press the ‘…’ button & choose a valid folder where your file will go—I chose my Desktop, but you should choose better.
   * Then, after leaving the ‘…’ file/folder chooser, you should enter something for ‘Table name’, which is the name of the first ‘table’/layer in the file, which could be anything, but I have also just called it ‘clipping’.
   * Next, choose the Geometry type—I chose MultiPolygon.
   * There is an unlabeled selection box where you choose the coordinate system—I am choosing EPSG:26910 (which is a UTM 10N using NAD83 datum) as that is what I had been projecting my rasters into.
   * I don’t bother with the fields, because I’m just using this for the geometry to clip with.

Then, press ‘OK’, & you should have created a new dataset that is empty. It displays on top of your layers list. Right click on its name (in my case, ‘clipping’) & choose ‘Toggle editing’ so we can start adding to it.

Now we add our feature (which we will later use to clip other layers). Under the ‘Edit’ menu you can ‘Add Circle’ -> ‘Add Circle by a Center Point and Another Point’. I first clicked on the Geography Department as the center of my circle. At this point, as the mouse cursor moves, the circle gets bigger & smaller—I make it large enough to get some of the sea & not so big that it goes beyond the edge of the imagery. I right-click to choose the radius. It prompts me to add a feature ID system to the data layer & I agree, choosing the defaults it presents. Now, if all has gone well, I have ahuge circle over campus that I can use to clip with. Before we go on, though, I right click on ‘clipping’ in the Layers pane & ‘Save edits’ to the clipping.gpkg file, also toggling off editing.

There are two ways to clip data—one way for vector data & the other for raster data. We actually only have to clip the vector layers here because we can use a setting in Qgis2threejs to constrain the DEM data/imagery by the clipping geometry—though if you have reason to clip raster data, as it can be handy in cartography, [here’s a description of what ‘clipping’ looks like for raster data in QGIS,](http://www.qgistutorials.com/en/docs/3/raster_mosaicing_and_clipping.html) where it is called clipping with a mask. (Actually, if you’re interested, that tutorial also happens to show how you mosaic DEM data of the sort I provided you with, so you could use it for your own projects… this tutorial is [part of an excellent tutorial website for QGIS in general](http://www.qgistutorials.com/en/index.html).)

**To clip vector layers**, I click on a layer in Layers, then go to the Vector menu, choose Geoprocessing Tools, & Clip. The input layer should be set to the layer to be clipped; the overlay layer is my ‘clipping’ circle layer. There is a Clipped output that can be set to write out to the disk—better than the default [Create temporary layer] which will disappear when you exit QGIS (though you can right click on them in the layers menu & save them to disk beforehand if needed). Name the outputted layer something appropriate. Turn off the corresponding original non-clipped layer in the Layers menu so you can see the results. I also turn off seeing ‘clipping’ as that would mask everything, visually, for me. Once I do this for both trees & buildings, I have my results!

This is looking much more promising—though we lost the original styling on the clipped buildings & trees. You can copy the styling from an old layer & paste it onto the corresponding new layer, though—right click on *ubcv\_building\_records* in the layer box & go to Styles -> Copy Styles -> All Style Categories. Then click on the new clipped buildings layer, e.g., *ubcv-building-records*, & right click on it, choose Styles -> Paste Styles -> All Style Categories. You can do the same for trees.

And just to make things less confusing, once I do this, I remove the original trees & buildings entries from the Layers list—you right click on each of them & choose Remove Layer. As long as layers are actually layers saved on disk & are not temporary layers made by a geoprocessing tool, you don’t lose them when you do this, though you may lose their styling in the project. Here, we have their clipped versions, properly styled, so that’s fine.

Let’s go back into the Qgis2threejs plugin. You may want to zoom out first. And don’t forget to ‘save’ the QGIS project first! In the Qgis2threejs plugin, you’ll probably have some work to do getting the various layers styled properly again. Mainly, you repeat what we did above, but *the key difference is that when you are setting up the DEM layer you check the box that says “Clip DEM with polygon layer” & you choose the clipping layer you made.*

Now, let’s work on our **3D styling of trees & buildings**.

In the Qgis2threejs exporter window in QGIS, this time, right click on the *ubcv-trees-clipped* point layer & open up its Properties. Notice we can render the trees as something other than a Sphere, including cylinders & even glTF models (e.g., of trees)—pull down that option selection, but leave it as a Sphere at the moment. Also consider: one can add a second copy of the tree layer & have trees be rendered both as a sphere (for the tops of the trees) & as a cylinder (for the trunks). Anita Graser does that in her [visualization of a park](https://anitagraser.com/2014/03/15/3d-viz-with-qgis-three-js/).

But, for now, we’ll just settle for green spherical trees that are above-ground. For that, we want to change the Z-coordinate mode from being absolute (measured vis-à-vis the DEM of zero, sea level) to being set ‘Relative’ & ‘Relative to the vancouver-area-dem’. Next, make the ‘Altitude’ of the trees *4* units above the ground (type that into the Expression textbox), so they are not just hemispherical bushes on the ground. And let’s make the radius smaller—how about entering an Expression of *3*. Note that we could make these into formulas if we knew more about how the trees varies, but we’ll just keep them as constants here. Hit OK & go re-export the results. You should see the trees above ground!

The **buildings are next**. Right-click on their layer in the exporter. Change the Z-coordinate mode to be relative to the DEM, as before. This time, leave the Altitude 0 (do you want floating buildings?). but if the object ‘Type’ at the top of the window is changed to ‘Extruded’, we can set the height—we can make the height be a function of the building’s actual height. Remember that this dataset has a field for the number of floors on the buildings, which is roughly proportional to the height on many buildings.

Click on the E/epsilon next to ‘Height’s Expression’. The big text box on the left is our expression (it’s like a rater calculator or field calculator in ArcGIS, but for data-driven styling) & the right side has various things we can include in our expression. In our case, because our buildings have a property called *COUNT\_FLOORS*, we can have the extruded heights be simple functions of that field’s value. Let’s multiple the number of floors times 5, with the resulting expression as follows: *5 \* COUNT\_FLOORS*

After this expression is entered, go back & re-export the 3D map to the web. Now there are buildings! (if your buildings are floating building roofs, you may need to set the “Object Type” in the Layer Properties window to “Extruded”). There’s one more thing to modify. Go back to the exporter. Go to the Scene menu, & go into the Scene Settings window. Look around for a second. Increase the **Z Exaggeration** to 1.5 then hit OK. Re-export the model. This isn’t terribly important in a place where the terrain is as dramatic as Point Grey, but it can be useful in many other cases to know about vertical exaggeration.

Now, pause for a minute. Generally, what sorts of thematic mapping would benefit from the use of 3D? as the examples showed above, it’s not solely cases where terrain matters (what are some situations where terrain matters to geographical processes?), though those are good use cases.

**Submitting**: What you’ve done above is what you need for the basic level of credit (around 0.75). Good job!

You’ll be submitting a single URL to a working map at an https:// address, most likely in a web-enabled repository of yours at GitHub

*You’ll need to upload a decent number of files at once, which can either be done via dragging a whole folder at once into a relevant GitHub repository webpage or by using GitHub Desktop (remember that application?)*

**Challenges**: QGIS2threejs is a rich library with a variety of possibilities. Explore the functionality of QGIS2threejs to make this map more interesting, elaborated, or useful. This might involve removing or heavily modifying layers I’ve had you add (in which case, submit the version from the tutorial above in a link in a comment, please!). In thinking about possibilities, the documentation can help, as might tutorials (linked where mentioned above in the text & at initial example images). Among them, Alasdair Rae’s blog post, [from the section called ‘The Process’ onwards](http://www.statsmapsnpix.com/2018/11/how-to-create-interactive-3d-landscape.html), has some ideas. And/or:

* One idea might be to add visible labels for some places on campus that you find interesting & unusual ([labels are seen in this example](http://alasdairrae.github.io/madeira3d2/madeira3d2.html)). You can do this in various ways, but one might involve making a GeoJSON file with points using GeoJSON.io & entering text into an attribute. Load this GeoJSON layer into your QGIS project & configure it within QGIS2threejs, as we did above. Using QGIS2threejs, you can set up the Attributes & Layers to ‘Export Attributes’—[see the docs here](https://qgis2threejs.readthedocs.io/en/docs/Exporter.html#vector-layer-settings).
* What about exporting this map to glTF format (File -> Save Scene As -> a .glb (binary format of glTF) file & importing it into a basic A-Frame scene (no globe needed!)? Three warnings:

1. If you don’t specifically select .glb format when you are in the ‘Save Scene As’ window by clicking & opening up the file format selector, the exporter may crash, at least on some earlier versions of the plugin—it seems fine right now, but just beware;
2. Once you import the .glb into A-Frame, your camera is likely underneath/within the model—it will look strange. You’ll need to reposition the model or camera or both. In order to do this, go into the A-Frame inspector & drag the camera or model to a position that makes sense & use the resulting settings to modify your A-Frame html.
3. If your .glb file is too big, it may not be accepted by either Glitch or GitHub. Try with a smaller file, or try hosting it elsewhere (if you have another option). There are also some relatively advanced ways of having GitHub host large files, but they’re a bit beyond the scope of this class.

* You could add a layer or edit how a layer is stylized.
* You can [edit the exported HTML/JavaScript](http://www.statsmapsnpix.com/2018/11/how-to-create-interactive-3d-landscape.html).
* There is apparently [a phone-based augmented reality export option in the Web Viewer Templates](https://qgis2threejs.readthedocs.io/en/docs/WebViewerTemplates.html#mobile-template). I haven’t tested it myself or seen any examples. You’d need to upload the resulting files to a GitHub pages web repo to test it. I’ll be curious to hear about any of your experiments! (Perhaps one could even develop a narrative or tour with it?)