

MAKING ANIMATED MAPS WITH QGIS

An introduction to animated mapping

with Alasdair Rae



QGIS version: 3.10 (A Coruña)

Workbook version and date: v1.5, February 2021

Automatic Knowledge Ltd

www.automaticknowledge.co.uk

Contents

1. Introduction.....	4
2. Creating frames with the MMQGIS Plugin.....	7
3. Creating frames with QGIS Atlas.....	14
4. Other QGIS animation methods (for information).....	21
5. Intermediate steps, tips and tricks.....	22
6. Creating animated GIFs in GIMP.....	24
7. Other tools for stitching together frames.....	30
8. Credits.....	32
9. Useful links.....	33
Back page cheat sheet.....	34

A note about QGIS versions

This workbook was written for QGIS version 3.10.14, the latest long-term release (LTR). If any screenshots in the workbook look different to the ones on your screen it is probably because you're on a different version, or have your toolbars positioned differently, but this is unlikely to cause any problems. Most QGIS versions are named after where the developers have their meetings (e.g. 3.10 is called A Coruña). The LTR version of QGIS is the most stable one, but you will also see a newer version of QGIS on the website – these often have more features but are not yet finalised for official long-term release.

This workbook is also available online, at:

<https://automaticknowledge.org/training/workbooks>

Licence: Attribution-Non Commercial 2.0 Generic (CC BY-NC 2.0)

<https://creativecommons.org/licenses/by-nc/2.0>



Notes

About Automatic Knowledge

Our philosophy is all about sharing data and knowledge, so that we can all be a bit better informed about the world. The way we do this is mainly through spatial data analysis and visualisation, and as part of this we offer training courses in geospatial software – mostly QGIS. Alasdair Rae (pictured) founded Automatic Knowledge in 2019 and also runs the training courses.



Why 'automatic knowledge'?

The idea behind Automatic Knowledge is that we do the hard work of completing the journey from data to knowledge, so that you can then make more informed decisions. These training sessions will hopefully help you do that too.

Other activities (e.g. free stuff)

In addition to training, we provide a range of consultancy services, specialising in data, spatial analysis, the built environment and cartography. We also publish free and open datasets that you may find useful, at:



automaticknowledge.co.uk/resources




We're also a 'sustaining member' of QGIS, which means that we donate money to the QGIS project on an annual basis, to help fund its development. By taking this course, you're helping too.

Automatic Knowledge training sessions

The idea behind all our training sessions, and these workbooks, is to help you learn new things in an enjoyable way, without confusing you. We want everyone who takes one of our courses to come away with useful new skills that they can then put into practice in their day-to-day work, and build on in the future.

About this workbook

Following a workbook can be a great way to learn new software skills, but there's also a risk that it turns us into robots, following step-by-step instructions in a linear way. During the session we'll go off at tangents and do some demos of useful skills –  among other things. *The emojis?* I add them in for a bit of colour, but they also serve a practical purpose because they can help us find key sections of the document quickly.

Formatting

Most font is size 14. When switching between screen and workbook this is easier on the eyes. The following format will be used in relation to files/folders, QGIS windows/tools, click actions, text I want you to input, and websites. I've also added a 'Notes' section on each page where you can jot things down.

Files, folders and so on: e.g. world_countries.shp

QGIS windows, tools, items, options: e.g. Print Layout

Actions - click menu item/ button: e.g. Vector > Research Tools

Text input: e.g. "geounit" IN ('Japan','New Zealand')

Websites: e.g. www.automaticknowledge.org/training

1. Introduction

This training session looks at how we can create animations using geographic data, plus QGIS, and then other software to stitch together individual frames to make an animated gif (or mp4 video). This is the basic formula for all animations.

Animations of geographic activities and patterns can be very powerful, engaging and informative – as well as fun – so I have used them over the years for various things, as have others.

Geogiffery in a nutshell — introduction to QGIS Time Manager



Topi Tjukanov Dec 26, 2017 · 10 min read



Geography + animated gifs = geogiffery

I have been posting a looooot of map animated maps lately on Twitter and quite often people ask me how can they make their own animations. I bet there are lot of ways to achieve similar results, but I am using QGIS + Time Manager plugin. I wanted to make a simple tutorial for beginners on how to get in to the fascinating world of geogiffery. As far as I know, the word *geogiffery* was first introduced to mankind by the great geogifferer Alasdair Rae.

Source: <https://medium.com/@tjukanov/geogiffery-in-a-nutshell-introduction-to-qgis-time-manager-31bb79f2af19>

Notes

There are a few resources out there on how to do animations with QGIS and the other tools you'll need (including the article above) but overall it can be quite a confusing topic. That's why I've decided to run a short Automatic Knowledge introductory training session on the topic.

By the way, I say 'animations *with* QGIS' rather than '*in* QGIS' because not every step of the process happens within QGIS. We'll use it to do the majority of the work, but we'll use other free, open source software to stitch the individual image frames together.

You'll see from the screenshot of Topi's article that he says 'there are a lot of ways to achieve similar results', so keep that in mind today – we'll focus on a couple of methods, but we'll mention others as well.

Here's a previous tutorial that I posted about one approach to making geogifs with QGIS Atlas, for reference:

www.statsmapsnpix.com/2017/12/how-to-make-geogif.html

For now, just remember these key points about animations.

1. We need to create individual frames.
2. We need these frames to be different – e.g. in each one a line gets longer, or the text and data changes, or an object moves position.

3. We need to combine a sequence of individual images into a single animation file – most commonly a gif or an mp4.
4. We need to be patient! The whole process can be confusing, and doesn't always go to plan.

We'll use QGIS plus GIMP today – and I'll also talk a bit about FFmpeg and ImageMagick (these are scarier, but powerful). All the content for today can be found on our training website, including data, example images, and this workbook.

www.automaticknowledge.org/training

I'll get you to download some of the data from the [/data](#) folder in due course, as well as look at some of the material in the [/animations](#) folder.

Okay then, it's time to get going. Here's what we'll focus on for the next couple of hours.

- Create frames in QGIS with the MMQGIS plugin.
- Create frames in QGIS with QGIS Atlas.
- Create optimized gifs in GIMP. Without optimization, gif files can be HUUUUUUGE.

We'll discuss technical things along the way but the three activities above are the main things. I've provided all the data and project files you need, so you can experiment as much as you like after this session as well.


2. Creating frames with the MMQGIS Plugin

It's time to make an animation. In this instance we're going to animate a line, which in this case is a road going from central Scotland all the way to the far north of Scotland.

- First of all, **start QGIS now**.
- If you don't already have it installed, you need to install the **MMQGIS Plugin** – do this via **Plugins > Manage and Install Plugins...** and then you just search for it and install it – it's very quick. MMQGIS is a python-based plugin for QGIS, written by Michael Minn. It is very useful, and at the time of writing it has been downloaded nearly 1 million times. It does *so much more* than animation, so you really should check out the MMQGIS home page – where Michael provides example data, plus very clear documentation. <http://michaelminn.com/linux/mmqgis>

Once you've installed MMQGIS, you need the data. This project has just three different layers in it – a place names layer, a road layer (for the A9 road) and a raster layer showing Scotland's topography. Download the files below into a single folder.

- Get **scotland_no_lochs_small_int32.zip** from the [/raster](#) folder **and then unzip it**.
- Get **A9_5_mile_buffer_places.gpkg** and **a9_diy.gpkg** from the [/animation](#) folder.

- To save you time with styling, I have also prepared some **qml** style files for the layers – you can find the corresponding style files for the two A9 layers in the [/styles](#) folder – make sure you save them in the same folder as the two A9 GeoPackage layers. You may need to right-click and choose **Save link as...** (or equivalent) in your web browser when downloading the **qml** files.
- Also download **a9_route_for_MMQGIS.qgz** – this is the QGIS project folder for the animation we'll produce.
- Now **open the QGIS project** file you just downloaded – when you do, you'll have to tell QGIS where some of the layers are, but this shouldn't take more than a minute or so.
- Once you've mapped the correct file paths above, **save your project** as **mmqgis-anim**. 
- Now have a little look around inside this QGIS project – zoom and pan, see how I've styled and labelled things and generally just have a closer look. Note that if you do not have the font I used (Poppins), QGIS will use a different one but you can just change that if you want to.
- Open up the **Print Layout** that I created already – it's called **Animation** and in case you



forgot you access it via **Project > Layouts...**

Okay, so we have what we need here. The topography layer isn't going to be animated – that's just for context. I wanted to create an animation that helped explain the route of the A9. When you're in a car driving to the far north of Scotland it can be a long and winding journey but of course that's because of all the mountains in the way!

The route of the A9 can be obtained as open data from Ordnance Survey – and other sources – but in this case I took about 5 minutes to digitize my own line because I didn't want it to be too detailed (e.g. I didn't need roundabouts and every single vertex showing every kink in the road).

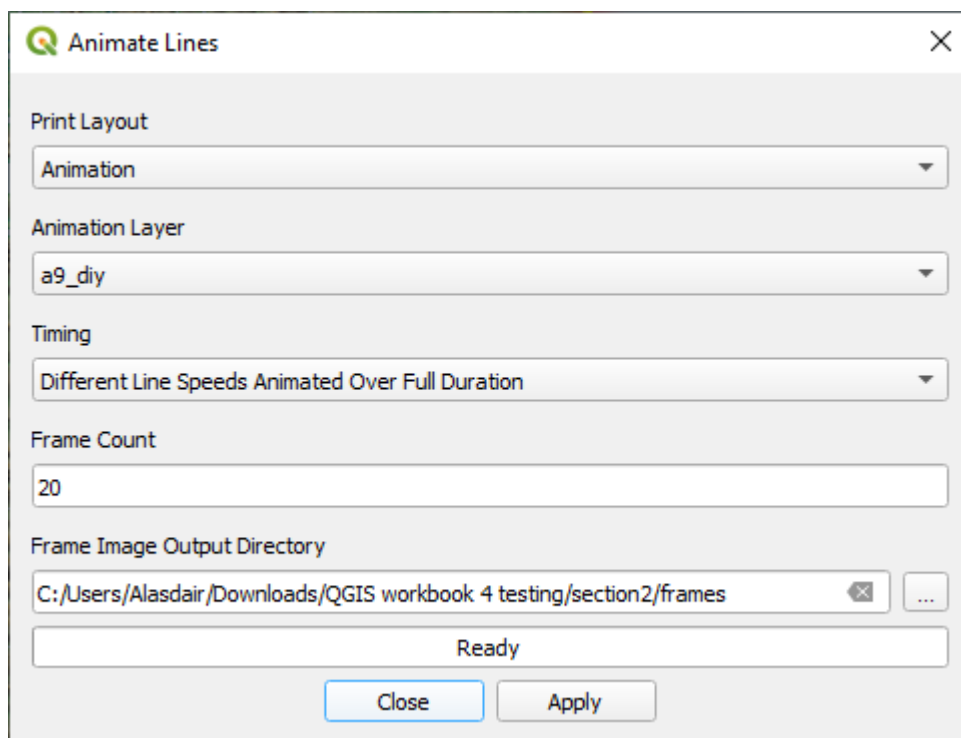
The red glow on the road helps draw attention to it and is achieved via **Draw effects** in QGIS.

Okay, now it's time to do some animating.

- Go to **MMQGIS > Animate > Animate Lines** and then you'll see the **Animate Lines** dialogue box. MMQGIS here is basically asking you 'where's your animation – i.e. what **Print Layout** is it in?', 'what layer do you want to animate?', 'how do you want the timing of the animation to work?', 'how many frames do you want to create?' and 'where do you want to save the frames?'. See below for how this looks on my screen. It should be very similar on yours.

Notes

Don't know how to create a line (or other new geometry) in QGIS? If you want to know, just ask me.



- I always do a little test first, with maybe 5 frames – just to make sure it's all working well. Go ahead and try exporting 5 frames and then you'll have an idea of how long it takes and how it works. Note that in the Frame Image Output Directory path above you need to select a **folder** to save the files to. When you hit Apply, it will create the frames.
- Okay, this worked for me. It took 15 seconds to export the 5 frames – and note that MMQGIS will also export a 'frame zero' with no animation in it. It also calls the frames frame000000.png, and so on, and outputs them as 300dpi png files.

- Delete the frames you just exported and repeat **Animate Lines** – this time export 20 frames.
- How fast the frames export depends upon how good your computer is. The computer I test this on isn't a top-end PC and when I exported the 20 frames it took 47 seconds.
- Open **frame000000.png** and then manually move to the next each frame using the arrow keys on your keyboard – this gives you an idea of what the animation would look like and you should see the line grow. But of course it's not very smooth because it **a)** is not yet an actual animation file yet and **b)** only has 20 frames in it.

The basic rule with animation is that the more frames you have, the smoother – and more aesthetically pleasing – your final animation will be. But of course we don't want to sit around and wait while we export 500 frames right now, so I've done it for you in advance. In the **/animation** folder you will see **a9_frames.zip** and this folder has 501 600x600 pixel images, like the ones we just exported. We'll download this folder in a moment.

Top tip

When you export frames from MMQGIS, they may be quite big. If you don't want to have the hassle of re-sizing your frames before importing them into GIMP, set the **Print Layout** page size to something like 100x100mm. You may have to reduce your label font size to make this look good but it can save you time.



- Go to the [/animation](#) folder and download the [a9_frames.zip](#) file. When it downloads and you unzip it you'll have a folder with 501 files in it.
- We're not going to create the animation yet – we'll do that later on, but [go to the link below](#) to see the final animation that I created from the frames.

[/animation/a9_route_opt_20ms_2000ms_end_pause.gif](#)

You can actually see some information about the file from the url above because I saved some of the details in the file name when I exported it from GIMP. Each frame is 20 milliseconds long. 1 millisecond is $1/1000^{\text{th}}$ of a second so each frame here is $20/1000^{\text{th}}$ of a second, or 0.02 seconds. There is also a 2000ms (2 second) pause at the end, to let the viewer take things in before the gif replays.

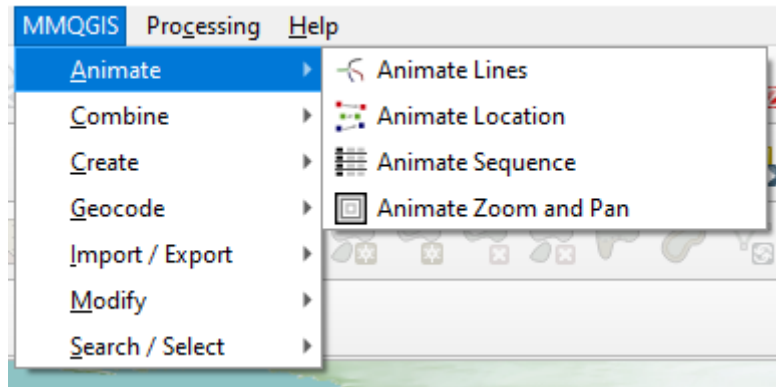
That means, the gif is 501 frames x 0.02 seconds long, with a 2 second pause at the end, for a total of just over 12 seconds. We can use command-line tools like ImageMagick to get full details of a gif (or other formats) but for now I'm just telling you this information so you understand a bit more about timing and the fact that we can adjust it.

- [Save your QGIS project](#) again now and then we'll move on to a more complicated animation project.



Before moving on, I just want to remind you that there are many different animation options in MMQGIS – as you can see from the MMQGIS menu screenshot below.

We have chosen to animate lines but there are three other ways. There are also loads of other useful MMQGIS tools, such as geocoding, modifying geometries, sorting tables, and more.



MMQGIS

Michael Minn (<http://michaelminn.com>)

16 January 2020

Describes use of MMQGIS, a set of Python vector map layer plugins for Quantum GIS

Introduction

MMQGIS is a set of [Python](#) plugins for manipulating vector map layers in [Quantum GIS](#): CSV input/output/join, geocoding, geometry conversion, buffering, hub analysis, simplification, column modification, and simple animation. MMQGIS provides verbose progress reporting, an intuitive user interface, direct file access, and some additional capabilities missing from other plugin sets such as the [Processing](#) toolbox.

Getting MMQGIS

MMQGIS is included in the [Quantum GIS Plugin Repository](#) and should be readily available in the QGIS Python Plugin Installer (Plugins -> Fetch Python Plugins). A [zip file of the current release is also available here](#) for manual installation. Links to older versions are listed in the [Version History](#) at the bottom of this page.

Data Formats

Input layers can be from any geospatial data source supported by QGIS.

Output file format are implied by the file extension given on the output file name. Formats currently supported through [OGR/GDAL](#) include:

- [ESRI Shapefile](#) (*.shp)
- [GeoJSON](#) (*.geojson)
- [SQLite \(Spatialite\)](#) (*.sqlite)
- [KML](#) (*.kml)
- [GPKG \(GeoPackage SQLite file\)](#) (*.gpkg)

Source: <http://michaelminn.com/linux/mmqgis/>

Notes

3. Creating frames with QGIS Atlas

Okay, so we've looked at one way of creating frames in QGIS. Now let's look at another.

One of our Automatic Knowledge training sessions focuses on how to automate the map production process in QGIS, using the Atlas tool, and that's what we're going to use now. Hopefully you're already familiar with it but if not, don't worry – I have prepared everything for you in advance and we're going to do this in more of a follow along way now, hence the little follow along icon I like to use.




Instead of taking a journey up the A9 to the north of Scotland, we're now going to fly from London to New York in only a few seconds!

Go to the link below to see what this looks like:

http://automaticknowledge.org/training/data/animation/LHR-JFK_v6_60in30out.mp4

Let's watch this together a few times and **I'll talk you through what's in it and what's going on**. This file is an mp4 but we can also see some gif versions in the same /animation folder.

- First, let's all open a New QGIS project  and also create a new folder for the files we're about to download.

- From the `/styles` folder we need to download all the files starting `LHR_out...` – there is one QGIS project file (`qgz`) and six style files (`qml`). On your browser you may have to do a right-click to download the `qml` because some browsers just open `qml` files as a text file in the browser.
- Then we need to grab the layers for the project. You'll find `LHR_out_ALL_01_segments.gpkg` in the `/animation` folder, the `HYP_HR_SR_W_clip` file in the `/raster` folder and then the `atlas_` layers in the `/data` folder. It will only take a few minutes to download all of these – just make sure you put them in the same folder.
- Then we open the QGIS project file and quickly tell QGIS where all the layers are stored, via the `Browse` button. Once we do that, we're almost ready.
- Where there should be a plane, you'll probably see a question mark. To fix that we just need to grab the `Avion_silhouette.svg` file from the `/animation` folder and then go into `Layer Properties` to change the source of the SVG marker. Once we've done this, we should see a plane crossing the Atlantic with a nice jet trail behind it.
- We'll then open up the `Print Layout` (it's called `Flights`) and probably see two big red X marks on the page – one in the top right, one in the bottom right. This is because QGIS doesn't know how to find the SVG files that were here when

I saved the project. We can remove these or, if you like, you can get the svg files from the `/images` folder and then tell QGIS where to find them, via the Image source button in each case, under Item Properties in the Print Layout.

- Make sure you save your QGIS project, now that you've mapped all the file paths, added in the SVG image source (possibly also for the Print Layout). I recommend saving this QGIS project as `atlas-anim`.

Remember this

There are many ways to package entire QGIS projects into one. In this example I've made all the files available individually on purpose because I want them to be easily accessible on an individual basis to anyone online. You can package whole projects inside a GeoPackage file now, or you can save and zip everything into a single archive. Other options include the QGIS Plugins QPackage and QConsolidate3.

Now I'm going to walk and talk through the QGIS Atlas project you see in front of you. We'll look at the following things, in this order.

1. Individual Layer Properties (styles, rules, filters, labels) – including how I made a line into a plane (and contrails).
2. We'll look at how I made the plane point forwards.
3. Then we'll switch to Print Layout and turn the Atlas on if it's not already on.
4. We'll navigate back and forth to individual frames.

Notes

Technical problems? Other problems? Don't understand what's going on? Feel free to interrupt me at any point, and ask in any way you like.

5. We'll look at the Output filename expression I've used and why this kind of things is important when exporting frames that we'll later use to create animations.
6. Anything else anybody wants to know.

I can go into more detail on the flight route data. but in case you are interested, here's a brief summary of my workflow.

1. Using the airports.dat and routes.dat text files from openflights.org I created a csv with flight origin and destination locations for the whole world.
2. I then imported this csv into QGIS and converted it into a line file. There is more information about this in our *Designing High Impact Maps in GIS* workbook in the [/workbooks](#) folder, for anyone who is interested.
3. Once I had the lines, going from origin airport to destination airport for just under 70,000 routes, I exported it to a new GeoPackage.
4. In order to make it possible for me to have the flight routes follow their true path on earth – rather than unrealistic straight lines – I used the *Densify by count* tool under *Vector geometry* in the *Processing Toolbox* (also accessible via *Vector > Geometry Tools* menu).
5. The step above is good because it allows us to wrap the flight routes round a globe projection – e.g. via the *The_World_From_Space* projection available by default in QGIS or by using the *Globe Builder* plugin.

6. Step 5 is really useful but observant people will notice that such flight paths do not follow great circle routes, which is more like what planes would actually fly. To convert my lines to great circle routes, I used the **Shape Tools** plugin.
7. Once I had my great circles, I ran a filter on the flights layer to I was only looking at routes out of LHR (London Heathrow). There were many multiple geometries (e.g. multiple flights from LHR to JFK) so I removed these using **Delete duplicate geometries** in the **Processing Toolbox**. That left me with 171 unique destinations from LHR.
8. In order to animate the lines using the QGIS Atlas method I then used the **Split lines by maximum length** tool to make the maximum length of a line segment 0.1 degrees. It is also possible to do it by distance but for that we'd need to re-project the layer first and since this was only for dataviz purposes I skipped that step.
9. Having done all of the above, I had the **LHR_out...** layer you now have. It has 89,966 line segments in it but doing that many frames would be a bit much so I filtered it just to show the London to New York route – hence the filter text you see in the layer in your QGIS project: **"DCOUNTRY" = 'United States' AND "DCITYO" = 'New York'**.

Baffled?

Don't worry if this all sounds a bit complicated, I realise it can be.


All you really need to know for now is that the routes the lines follow are a curved because they follow great circles and that because each route is actually made up of many individual segments it means can animate it.




In a moment or two I'm going to export a few frames to show you how this works, but I won't export all frames because that would take too long during this session.

You can find a full set of image frames for the London to New York flight route in the [/animation](#) folder. There are 758 frames in here, matching the number of line segments in the London to New York flight route.

[/animation/flight_frames.zip](#)

In order to make things a little bit more interesting, I have made it easy for you to choose a different destination. All you'd need to do is go into the layer filter  for the two **LHR_out** layers in the **Layers Panel** and change the filter from what it currently says.

- If you want the route to go from London to Chicago, for example, you'd replace New York with Chicago. If you want the plane to go to Tehran you'd just change the country to Iran and the city to Tehran.

Just note that when you change the filter everything might disappear so you may have to refresh the map and **Print Layout views and/or turn the Atlas on and off again using the  **Preview Atlas button**.**

Okay, let's export some frames now – see how on the next page.

Now I'm going to tempt fate and attempt to export all 758 frames, live! (I'll abort if it's taking too long)

Here we go...

- Atlas > Export Atlas as Images... then I need to select a folder and after that I can choose my image dimensions and/or dpi. I will try exporting images 1000 pixels wide and see how it goes.

I may only export the first 100, depending upon how long it takes. This kind of thing is best done when you can leave your computer to work away on its own for a while, rather than during a live training session but you have the link to the final frames already.

[/animation/flight_frames.zip](#)

- Let's all look at the files I created – hopefully you've managed to download them all but if not you can just look at my screen.
- I'm going to scan through the images using the forward arrow on my keyboard and hopefully you can do this too. What you should see is a plane flying from London to New York, with a jet trail behind it.

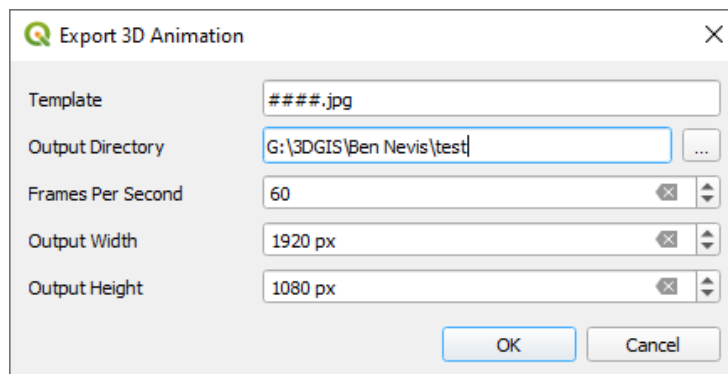


Okay, so now we have lots of frames – Save and move on.

4. Other QGIS animation methods (for information)

Before we move on to create some animated gifs with GIMP, it's worth letting you know about another QGIS animation methods.

The first is via the 3D map view (View > New 3D Map View) and then you create an animation and export frames. There's more to this than we can cover today but it's definitely



worth a look. Here's an example animation I made with this tool.

[/animation/ben_nevis_slow_music.mp4](#)

Another great animation option is the TimeManager plugin by Anita Graser. From QGIS version 3.14 onwards, this functionality is built in to QGIS itself, via Layer Properties and known as the Temporal Controller. Let's say you have a dataset from NASA, of when meteors hit the earth, or a dataset of traffic accidents by time and date – you could animate these with this tool.

Date formats are notoriously tricky, so be aware that you may need to mess around with date formats if you go down this route. Here's an example of the growth in short-term holiday rentals in the Isle of Skye that I made with this tool.

[/animation/skye_STL_growth.mp4](#)

5. Intermediate steps, tips and tricks

If you start getting into animating spatial data it can be really interesting, really useful and also good fun. But it can also suck up a LOT of your time.

For this reason I thought it would be useful to share a few of my tips with you.

- My normal method when exporting images for individual frames is to export them in a **16:9 width to height ratio**, to match the aspect ratio of modern screens. Otherwise, any animation you make will have dead space when viewed on many platforms. However, on Twitter and some other platforms a square gif (or mp4) may work well.
- **Think about images sizes in advance.** I normally export using dimensions of 1920x1080 pixels (16:9 aspect ratio and standard HD size). I can then create an mp4 from this very easily and I can resize images before creating a gif. If I don't want to create an mp4 file I will often just export smaller sized files – e.g. 960x540 – but still using the same aspect ratio. Think about end use at the start.
- **Think carefully about the end use of your animation** and it will save time when it comes to composition and export. After all, there's nothing more annoying than having to go back and re-export hundreds or even thousands of frames and stitch them back together again.

- **In general, it's best to name our files with sequential names**, with zeros for padding. This could be something like frame000000.png, frame000001.png and so on, or it could be as simple as 001.png, 002.png, 003.png. This is the default for MMQGIS exports but in QGIS Atlas you'll have to do this yourself – hence my QGIS Atlas example with padded zeros.
- If you do need to re-name and/or resize hundreds of files then it can be done in Windows using **IrfanView** (which I used a lot for this – just Google it) or on the command-line with **ImageMagick**. ImageMagick is very powerful.
- When it comes to creating gifs, if you don't **optimize** then you'll end up with massive file sizes – too big to easily share and way beyond the 15MB gif size limit on Twitter, for example. The LHR-NYC animation without optimization was about 140MB, and about 2MB with optimization.
- Lastly, even though it can seem confusing and daunting, **try to get into command-line tools like ImageMagick or FFmpeg** (below!). They can seem really mysterious and strange but they can do so much. I am by no means an expert though – but I have found them so helpful.

```
Side data:
  cpb: bitrate max/min/avg: 0/0/0 buffer size: 0 vbv_delay: -1
frame= 256 fps= 27 q=-1.0 Lsize=      365kB time=00:00:06.32 bitrate= 472.1kbits/s du
video:361kB audio:0kB subtitle:0kB other streams:0kB global headers:0kB muxing overhea
[libx264 @ 00000283c71ada80] frame I:2      Avg QP:17.48  size:151938
```


6. Creating animated GIFs in GIMP

In this part we're actually going to create some animated gifs!



We've already looked at two different ways to create the individual frames – and we have two separate batches of files that I shared in zipped folders.



Depending upon how you got on, you may have your own set of frames as well – that's great.



In order not to put too much stress on your computer, I recommend starting small, with a batch of 50 images and then depending upon how that goes we can experiment further. But once you know the method the only thing holding you back will be the power of your computer, plus your own patience.

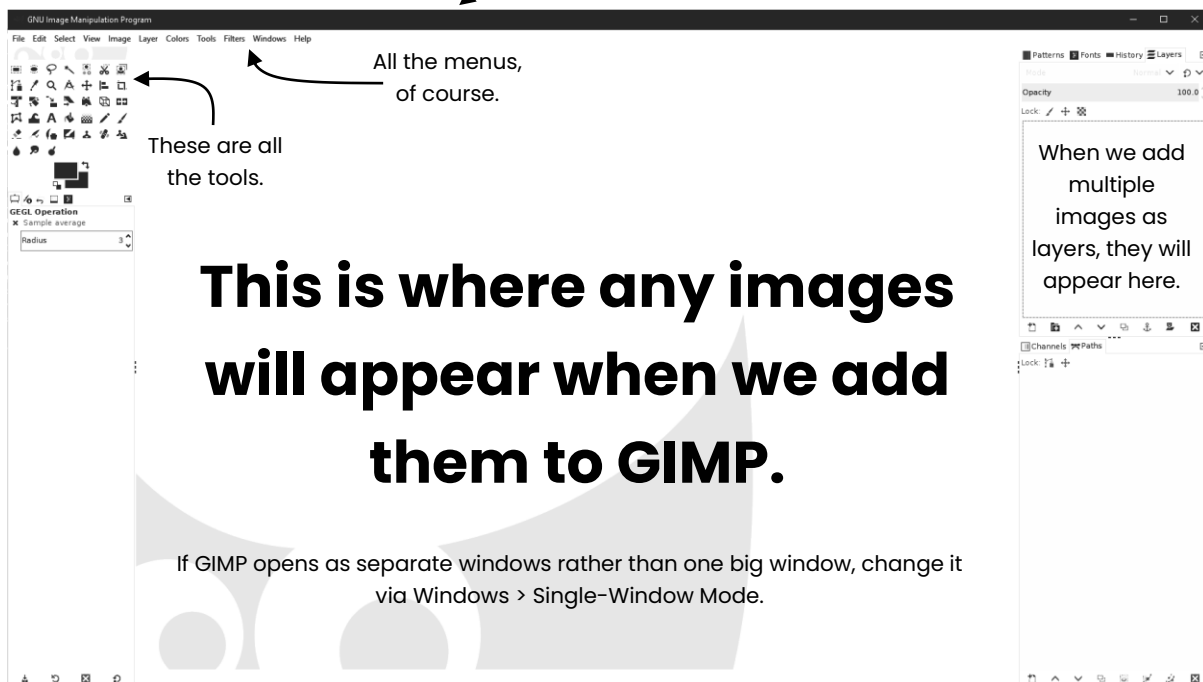
Follow along with me now

You don't need QGIS open any more so you can close it down if it was still open. Also, shut down any other software you don't need right now.



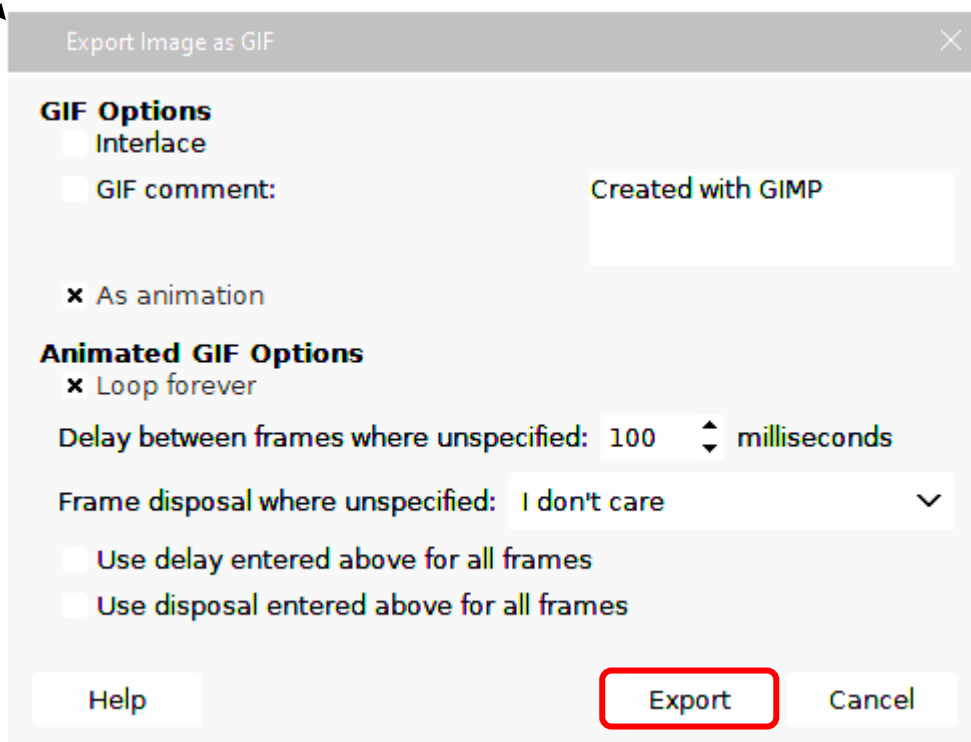
- Let's all **start up GIMP** if it's not already running. If you're not used to using it, it can be a bit confusing, but I'll walk you through the steps either way.
- My screenshots are based on GIMP 2.10 so they should be identical or very similar to yours – the only difference in this

workbook is that I have turned the screenshot white to save ink when printing. This is what I see when I open GIMP.



- Now let's try to add 50 of our frames to GIMP. I'm going to use the unzipped folder with the LHR-JFK flight images in it because these are nice and small, sized at 960x540 pixels.
- To add the first 50 of these images to GIMP, we want to go to **File > Open as Layers...** and then navigate to the folder where the LHR-JFK frames are. Then we just select the first one, then hold down **Shift**, and then select the 50th one and then we click **Open**. After a few moments – perhaps a 30 seconds or more – you'll see all the numbered PNG files in the **Layers** panel to the right of the GIMP interface.

- Now go to **File > Export As...** and then in the **Name** box type in **lhr-jfk_raw.gif**. Then click on **Export** and you will see another little window – this is the **Export Image as GIF** window. This time, we'll just make sure the **As animation** box is checked, as well as **Loop forever**. Leave **100** as the **Delay** and then click **Export**, making sure you set it to save somewhere easy to find (e.g. your folder for this session).



- Look in the folder you saved it to and you should see that the file size (if using the same images as me) is 5.29MB. Open the gif file and you'll see the plane leaving London now, but not going very far yet – it's only 50 frames.

Okay, so that shows you how easy it is to create a gif from a series of images in GIMP **but the file size is not good**. That is with only 50 frames, so with 500 the file size would be HUGE. That's why we need to **Optimize**. Here's what the GIMP help docs online say about **Optimize**. We'll use **Optimize (GIF)** below in a moment.

16.7. Optimize

16.7.1. Overview


An animation can contain several layers and so its size can be important. This is annoying for a Web page. The Optimize filters let you reduce this size. Many elements are shared by all layers in an animation; so they can be saved only once instead of being saved in all layers, and what has changed in each layer can be saved only.

GIMP offers two Optimize filters: **Optimize (Difference)** and **Optimize (GIF)**. Their result doesn't look very different.

Source: <https://docs.gimp.org/2.10/ko/plugin-optimize.html>

The key point above is 'many elements are shared by all layers in an animation; so they can be saved only once instead of being saved in all layers' – **only the parts of the image that change are saved**, and the file size is reduced, often by a large amount. This is what Optimization does.

- Go back to GIMP again and go to **Filters > Animation > Optimize (for GIF)** and then watch the progress bar at the bottom of the GIF window – this will take a little while. You'll see that there are a new series of images on the right and that on the main view there should be a little yellow/black hashed line round the part of the image that will move.

- To see the above process in effect, scroll down to image 001 and turn it off by hitting the  eye icon. You will now see the very small area of the frame that is to be animated –this helps explain how the process can reduce the file size. When GIMP moves between frames, only a small fraction of our overall image is changed each time.
- Use **File > Save As...** to save this newly optimized project as something like **LHR-JFK_opt_layers.xcf**. GIMP uses the **xcf** extension for saving projects. Note that this saves the PROJECT, not the image.
- Now re-export your gif using the same method as before – **File > Export As...** but this time call it **lhr-jfk_opt.gif** to indicate in the file name that it has been optimized. Look at the file size now – mine has gone from 5.29MB down to 154KB. **That's around 3% of the size of the original, raw gif.** You will notice very little (if any) difference visually.
- Now let's go back into **File > Export As...** and change the **Delay** time to **50**. The delay time is expressed in thousands of a second, so if you want 1 frame per second you'd enter 1000. With a delay of 50 you'll get 20 frames per second. Once you've done this we can head to the next page.

Task

Now you know how to create an animated gif from frames!



Excellent – so now you can experiment with all the options.

You can go to **File > Close All...** to clear the previous GIMP projects (you don't need to worry about saving them when the warning comes up). Then you can just go ahead and add in layers again via **File > Open as Layers...** and this time perhaps start by adding in all the A9 layers or, if your computer is struggling, just half of them.

Here are some ideas. Ideally you should optimize each time, unless it is causing your computer to run extremely slowly.

- Can you create a gif with a frame rate of 10?
- Can you create a gif with a frame rate of 200?
- Can you create a gif with the full A9 route, all the way to Thurso?
- Can you create a gif with the full flight from London to New York? (don't worry if this is too much for your computer – it may be better to try this afterwards. Now you know the method, it should be straightforward.

You can see several gif and mp4 examples that I created with the same data at the following link.

<http://automaticknowledge.org/training/data/animation>

Note how fast some of the journeys are!

7. Other tools for stitching together frames

Depending upon how much time we have, we'll also look at and/or talk about the following tools that can be used to stitch together individual images to create animations.

FFmpeg – described on the website as 'A complete, cross-platform solution to record, convert and stream audio and video'. Unlike GIMP or QGIS, it's not point-and-click software. Instead, it is designed for command-line processing. On a Mac or Linux this would mean you run it via a Terminal window, and via the Command Prompt on Windows. The mp4 files you see in the [/animation](#) folder were created with FFmpeg. There's also a bit of code that will allow you to create an mp4 yourself.

ImageMagick – like FFmpeg this works on all platforms and also runs via the command-line. You can use it for all sorts of image processing, as well as for converting file formats. I used it to convert the png images to a gif. Once again you can see the results of this, plus a bit of code, in the [/animation](#) folder.

Gifsicle – another command-line tool, dedicated to making animated gifs easier. I don't use it much but it's a useful option.

ezGIF – if you really don't want to try any command-line stuff then you can always upload individual frames to ezGIF and make an animated gif that way. You can actually upload up to 2000 files with each file being a maximum of 6MB and the total no more than 100MB. You can find it at [ezgif.com](#) and then go to

the GIF maker tab. If you go to the [about](#) page of their website you will see that it's powered by, among other tools, ImageMagick, FFmpeg and Gifsicle. ezGIF is developed and hosted by Open Idea, from Latvia.

A GIMP script – you can change the frame timing of individual layers in GIMP one by one*, but if you want to change them individually (e.g. pause the final frame) then you will find the animation settings plugin useful. You can find it online but I've also put it in the [/animation](#) folder (it's also open source). You just need to download the `sg-anim-settings.scm` file and put it in the correct folder to use it in GIMP – for me it was this folder: `C:\Program Files\GIMP 2\share\gimp\2.0\scripts`. Once it's there, you can access it in GIMP via `Filters > Animation > Settings...` I find this particularly useful when I want to hold the final frame for a few seconds so that people can pause and reflect on what they've seen.

If we have enough time, I'll demo FFmpeg and ImageMagick so that you can see how it works.



I don't like to leave people high and dry after a training session so if you're stuck or need me to clarify anything, feel free to get in touch – email, website contact form, Twitter DM, whatever.

Thanks for joining in!

Notes

* In GIMP you can double click a layer on the right and manually change 100ms to whatever duration you want for a frame. But, you can't do this for all of them at once without the script above.

8. Credits

QGIS – a user friendly, free open source GIS. QGIS is an official project of the Open Source Geospatial Foundation (OSGeo).

<https://www.qgis.org/en/site/about/index.html>

FFmpeg – a free, complete, cross-platform solution to record, convert and stream audio and video.

<https://ffmpeg.org/>

GIMP – a free, cross-platform image editor available for GNU/Linux, OS X, Windows and more operating systems.

<https://www.gimp.org/>

ImageMagick – Use ImageMagick® to create, edit, compose, and convert digital images. Resize an image, crop it, change its shades and colours, add captions, and more.

<https://imagemagick.org/>

Natural Earth – ‘no permission is needed to use Natural Earth data’. All versions of Natural Earth raster + vector map data found on this website are in the public domain.

<https://www.naturalearthdata.com/about/terms-of-use>

Emojis are from Twemoji, at twemoji.twitter.com, licenced under the CC-BY 4.0 licence.

<https://twemoji.twitter.com/>

Poppins Font – Designed by Indian Type Foundry, Jonny Pinhorn, licenced under the Open Font Licence.

https://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=OFL

9. Useful links

Natural Earth – the best source for easy-to-use global map data, e.g. countries, places, states, roads, coastlines and much more.

<https://www.naturalearthdata.com>

simplemaps – world cities csv files, with lat/long coordinates. This is a great source for place name data.

<https://simplemaps.com/data/world-cities>

Sentinel-2 cloudless – for adding satellite imagery to QGIS. Right-click WMS/WMTS in the QGIS browser panel, then click New Connection... add Sentinel-2 cloudless to the Name and the second url below to the URL field.

<https://s2maps.eu/>

<https://tiles.maps.eox.at/wms?service=wms&request=getcapabilities>

OS OpenData Downloads (Ordnance Survey) – a huge amount of free, open geospatial data are now available from Ordnance Survey via their Open Data Hub. The data covers Great Britain and no registration is required.

<https://osdatahub.os.uk/downloads/open>

ONS Geography Portal (UK) – see the Boundaries section of this website for a wide range of UK boundary data. This site includes data for the whole of the UK, whereas Ordnance Survey data only covers Great Britain. A great resource.

<https://geoportal.statistics.gov.uk/>

QGIS Documentation – the official help docs from the people who make QGIS.

<https://www.qgis.org/en/docs/index.html>

Back page cheat sheet



These are things I use myself – to make things work faster, better, smoother.

- **CTRL+Tab** turns panels on/off so the map area is full screen width.
- Select some features, **Edit > Copy Features** then **Edit > Paste Features As** a new temporary layer (called a 'scratch' layer) or a new vector layer.
- **F11** for full screen mode, **F1** for QGIS User Guide, **F6** to open Attribute table (on Windows).
- **CTRL+Shift+Tab** for full screen map mode (on Windows). Same again to undo.
- Change projection for project (it doesn't alter any data) via **EPSG** button, bottom right.
- Black is too harsh, try **#222222** / RGB 34, 34, 34 instead. Go beyond default colours!
- Where is your layer stored? Double-click a layer, go to **Information** and see **Path**.
- Turn on the thousand separator (1,000 vs 1000) – **Settings > Options > General**, tick box.
- Want to save the position of your map? Use **View > New Spatial Bookmark**.
- **Print Layout** looking fuzzy? Just hit refresh and that will fix it.
- **Print Layout** page not zooming to full extent? Re-size your page, then reset to original.
- Save your most commonly used data source paths as **Favorites** in the **Browser**.
- When styling layers, you don't have to use only the columns you already have – e.g. you can style based on calculations or even by extracting portions of text.
- Hey, my lovely XYZ base map looks a bit fuzzy! Set projection to **EPSG:3857**.
- Hey, my joined layer has weird column header names! Go back into **Joins**, and edit the join so that the **Custom Field Name Prefix** is ticked and the box is blank.
- How do you do those glowing lines/polygons? That's via **Blending mode** in **Symbology**.
- Help! My side panel things have disappeared. Go to **View > Panels**.
- Use **Inverted polygons** with a filter and a satellite base layer to create nice area maps – make layer white/black and use about 75% **Opacity**.
- Go to **Processing > History** to view or re-run your recent geoprocessing tasks.
- A general carto tip: try to make your map as simple as possible, remove any junk.
- Type **world** into the **Coordinate** box if you want a world countries layer to use.
- Use a 16:9 aspect ratio if your maps are mainly for screens (e.g. 320mm x 180mm).
- Map a list of x and y coordinates quickly via **Delimited Text** in **Data Source Manager**.
- Drag and drop files from your file browser directly into QGIS (e.g. **shp**, **gpkg** etc).
- You can add layers via the **WMS/WMTS** option in **Browser**. Just Google it.
- Want to save/re-use a map style for a layer? **Layer Properties > Style > Save Style...**
- If you give the **qml** file the same name as a layer file and put it in the same folder as the file then when you add it to QGIS that style will be applied by default.
- Learn about **Draw effects**! Use them to add a glow or a shadow to features.
- **View > Preview Mode** to see your map in colour-blind safe modes or greyscale.
- Best Plugins? **MMQGIS**, **SRTM Downloader**, **Build Globe View**, **Qgis2threejs**, **qgis2web**.
- Add base maps via **XYZ Tiles** in **Browser** panel. **OpenStreetMap** is there by default.