

Final Report
IMT4951
Digital Workflow
How to automate the process of creating KML files
to publish pictures in Google Maps

Christopher Bjerkelund
Student number: 071306

December 1, 2009

Contents

1 Project Background and Goal	3
1.1 Planing	3
1.2 Methods	3
1.3 Legal Implications	3
2 Documentation	4
2.1 Process	4
2.2 Workflow	5
2.2.1 Collecting Data	5
2.2.2 Add Metadata	6
2.2.3 Move/Resize	6
2.2.4 Presenting using Mashup	7
2.3 End product	7
2.4 Automation	7
3 Future works	8
4 Conclusion	8

1 Project Background and Goal

This project is a result of the course IMT4951 Applied Digital Workflow¹ at Gjøvik University College. Among different project proposals the course lecturer encouraged to focus on mashup of pictures in Google Maps. By this we hoped to improve the mashup process and use the result in future development of solutions to be used in relation to Gjøvik Citys 150th birthday next year.

The goal of my project is to: Create a workflow to simplify the process of collecting, describing, transforming and publishing photos in Google maps and make a web pages who displays a mashup of photos taken today in Vigelandsparken in Oslo together with old photos collected by the National Library of Norway.

All the results, scripts and documentation is available at the project homepage found at URL:

`"http://www.stud.hig.no/ 071306/IMT4951/"`

1.1 Planing

The Project was planed in week 44 and 45. A project proposal with risk chart and proposal plan was written and delivered in week 45. The proposal report is available online².

1.2 Methods

The project has been accomplished by the use of open source technology. The scripts are made in Unix environments and the O'Reilly book "Learning the Unix Operation System"[1] have been used frequently. The scripts are dependent on free software like Exiftool³ by Phil Harvey and ImageMagik⁴ by Image Magik Studio LLC.

1.3 Legal Implications

All parts of this project are released with a Creative Commons BY-SA-NC⁵ license. This means that the work is free for others to share and remix

¹<http://www.ansatt.hig.no/kjellr/imt4951/index.html>

²<http://www.stud.hig.no/ 071306/IMT4951/>

³<http://www.sno.phy.queensu.ca/ phil/exiftool/>

⁴<http://www.imagemagick.org/script/index.php>

⁵<http://creativecommons.org/licenses/by-nc-sa/1.0/>

under conditions of this license and that Gjøvik University College can use the code and workflow in creative work for example in correlation with the 150th birthday celebration.

2 Documentation

The project consists of different parts and steps towards the final workflow. In this section I will review the process and describe the different parts. I also want to give some information about issues during the project process.

2.1 Process

I had four weeks to complete the project and the final-report after I had delivered the project proposal. Week 46 was used to collect data according to the project plan. The weather was not as good as I had hoped so I waited to the end of the week, but the results are still a bit darker and grayer than I had hoped for in advance. This is also because my re-shoot was in November and most of Anders Beer Wilses⁶ photos was taken in June or July.

I used iTrail⁷ to collect GPS information during the hours collecting images with my Canon G9 digital camera. iTrail was set to collect GPS information every fifth second. Later I found out that this interval was too long and that some of my pictures do not have a totally accurately GPS tags. This issue could be a result of some incoming phone calls while I was running iTrail. A possible solution to the problem could be to put the phone in flight mode while collecting data with iTrail. Another issue I discovered with the iTrail software was that the pictures I took with the phone while iTrail was running did not contain any geographical information. iTrail cost 17 kr and only runs at a Apple iphone/iTouch devices. A free alternative is the Easy Trails GPS Lite software for iPhone. Different dedicated devices like cameras, phones and GPS receivers can also collect GPX information.

Week 47 was dedicated to scripting and creating the workflow. This week I was stroked by illness and experienced the vulnerability of a project consisting of one person. After my period of sickness I was running late, and had to increase the work speed and time spent to reach the delivery deadline.

⁶http://no.wikipedia.org/wiki/Anders_Beer_Wilse

⁷<http://sites.google.com/site/iphoneitrail/>

The workflow and scripts was developed in week 48. I was researching different technologies and techniques to automate the creation of the KML file. Java or Action Scripts would definitely do the job, but I wanted to be able to do the task with as little code as possible and preferably by using bash scripts. Using the linux `catinate`⁸ command to combine files in combination with the use of metadata pulled by Exiftool was described by Phil Harvey on Exiftools forum⁹. The `catinate` command makes it possible to create the start and end tags needed in KML in separate files. This again makes it possible to generate a well formed KML document with placemarks for each picture containing metadata. The Google KML documentation was useful and enlightening¹⁰.

I ran the script on the server. The pictures had to be uploaded and available online. I see it as a great advantage to run the scripts, generating KML, in the picture folder on the server to simplify the workflow and for automation purposes. I installed exiftool in my home folder and uploaded script and pictures. This ran fine, except from some issues with the exiftool perl command. On the server exiftool needed perl explicit in front of the commands. This is not needed in for example OSX. I later used crontab¹¹ to get the script on the server to run automatic at given times.

The last part of the project was spent with Latex¹² and with the writing of this report. I had guidance sessions with my lecturer Kjell Are Refsvik during this period with a lot of constructive feedback.

2.2 Workflow

In this section I will describe the workflow created and explain the different parts.

2.2.1 Collecting Data

The data used in the project was collected in week 46 by the use of a Canon G9 digital camera and an iPhone 3G. I had in advance selected five pictures made available in GalleriNor¹³ that I wanted to make similar copies of today. These are pictures taken by Anders Beer Wilse made available online. In view of the fact that my camera doesn't a built in GPS tracker I used

⁸http://www.gnu.org/software/coreutils/manual/html_node/cat-invocation.html

⁹<http://www.cpanforum.com/dist/Image-ExifTool>

¹⁰http://code.google.com/intl/nb/apis/kml/documentation/kml_tut.html

¹¹<http://en.wikipedia.org/wiki/Cron>

¹²<http://www.latex-project.org/>

¹³<http://www.nb.no/gallerinor/>

iTrail on my iPhone to track the positions of my pictures. This application generates a GPX-file with time and coordinates that I later can combine with the image files by using Exiftool. It is also possible to add the longitude and latitude metadata manually in Exiftool if I have another picture with this data recorded, like a low resolution picture taken by my iPhone. I choose the GPX-file because this makes it more convenient to geotag a large number of pictures in one operation.

2.2.2 Add Metadata

It was important that the pictures contained the metadata needed by the scripts. In my case the most important metadata information was the geotags. It would be impossible for Google Maps to display the pictures in the correct places on the map if no geodata was provided. I used the GPX file to complete this task if the pictures did not already contain geotags added automatically by the camera. All of the metadata tags in the pictures can be used by the script to be displayed in the KML code. Useful tags could be filename, creator, description or Date/Time. In this project I have focused on displaying the pictures at the right position on the map, but it is without doubt a lot of possibilities in extending the script to use different descriptive metadata in the output.

The scriptfile Script1_GPS.sh can be used to geotag pictures. The script uses Exiftool so this has to be installed. You have to run the script in the same folder as the pictures, and you have to provide a .GPX file for the script to read geoinformation.

2.2.3 Move/Resize

I choose to transform my pictures size before I upload them to the loke.hig server because this reduce the uploading time and optimizes the pictures for display on a regular sized desktop monitor. If I did not resize the images they would be available on the server in full size resolution. This could be good if someone found special interest in small details in the images that are hard to se on a low-resolution image.

Script2_resize.sh is a standalone script provided in the workflow and is optional to use. Either way the next script in the workflow (script3) will create 300px wide thumbs to be displayed in Google Maps. After the pictures are geotaged and resized they are copied to a picture folder on the

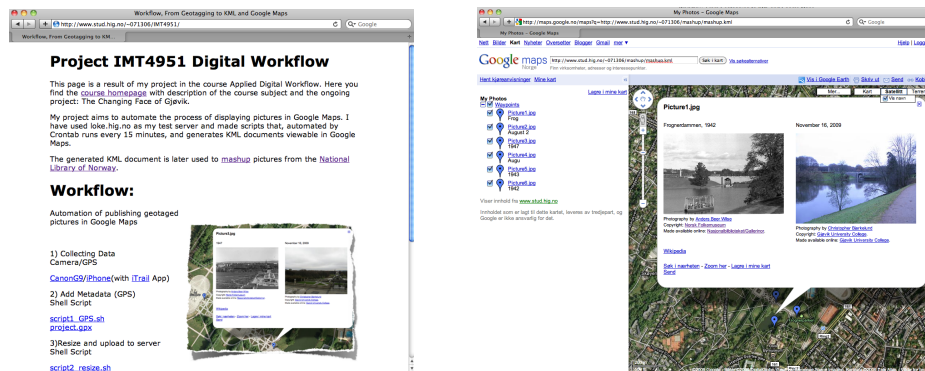
loke.hig server. This task is done by the use of secure copy¹⁴. The script will prompt for the password of the user account.

2.2.4 Presenting using Mashup

The server script3_remote.sh generates a kml file with all the pictures in the picture folder. To make the mashup with the National Library pictures I edited this file to also include selected pictures taken by Anders Beer Wilse. The result is published on the project homepage¹⁵.

2.3 End product

The project consist of:			
Script files:	KML files:	PDF report files:	Image files:
Script1	picture.kml	Pre-proposal.pdf	Picture1.jpg
Script2	mashup.kml	Final-Report.pdf	Picture2.jpg
Script3			Picture3.jpg
kml-start.xml			Picture4.jpg
kml-placemark.xml			Picture5.jpg
kml-end.xml			Picture6.jpg
timer_remote.ctab			



Images: project homepage

2.4 Automation

The script (script3_remote.sh) use Exiftool to read the metadata from the image files. To automate the generating of the KML file I wanted to make

¹⁴[http://en.wikipedia.org/wiki/Cp_\(Unix\)](http://en.wikipedia.org/wiki/Cp_(Unix))
¹⁵<http://www.stud.hig.no/071306/IMT4951/>

the script run at a given interval on the server. This would make the KML file up to date with the images provided in the picture folder. This part recured some testing, but I ended up using a Cronjob to complete the task and set the script to run every 15 minutes. The interval time was chosen based on the activity in the picture folder. A shorter interval would be preferred if the folder had been updated regularly and it was important that all changes to the folder was updated right away. Crontab allow down to one minute. The “timer_remote.ctab” used for the cronjob is available on the project homepage.

3 Future works

Some issues and questions have appeared during this project and have to be addressed to complete a total automation of mashup of pictures in Google Maps. For example how to automate the combining of the old and new pictures and how to add or use metadata from the pictures provided by the National Library without copying them from their original location at nb.no. The mashup could be done by naming the re-shoot pictures after the corresponding picture-id in the National Library. Another way would be to add specific metadata tags to hold the id-information. To use different metadata tags in combination with the tools provided by Exiftool and bash commands is a great way to start and our fantasy is the only limitation.

The code and workflow presented in this report could be used as a standalone product. To be able to put pictures into geographically positions online can be very handy and useful. This could be used in projects like “The changing face of Gjøvik” where people work together to collect images in a specific geographical area. Other areas of usage could be expeditions or travellers. The geotagged pictures on the map is a great way of summing up the trip and get an idea of where the pictures were taken. The scripts could further be included in a web service providing easy upload of pictures to Google Maps. A service like this would make it possible for ordinary web users to easily share their pictures online on the map.

4 Conclusion

The project was completed according to the project plan except for week 47 due to illness. The goal of the project was to: Create a workflow to simplify the process of collecting, describing, transforming photos in Google Maps and make a webpage which displays a mashup of pictures taken today in

Vigelandsparken together with old photos collected by the National Library of Norway. The project has provided code that simplifies publishing of pictures in a KML file and has created a mashup webpage of old and new photos. The automation of the creation of the KML file with the use of Exiftool and metadata tags is one step towards the complete automation of mashups of Google Maps and pictures from the National Library of Norway.

“An Earth browser without a KML file will offer a richly detailed world to explore and enjoy - it will lack only the annotation information that would turn the planet into a storytelling mechanism”

— *Michael T. Jones, Google*[2]

References

- [1] G. Todino, J. Strang, and J. Peek. *Learning the UNIX operating system*. O'Reilly & Associates, 1993.
- [2] J. Wernecke. *The KML handbook: geographic visualization for the Web*. Addison-Wesley Professional, 2008.