

Automatization of Time-lapse movies using Open Source software Final Report

By Jose Mario Perez Velazquez (091338)

December 2, 2009

1 Introduction

The present document has as purpose be the final report to describe the project "Automatization of Time-lapse movies using Open Source software", its plan, implementation and its results. This project was developed by Jose Mario Perez (jose.velazquez@hig.no) for the IMT4951 Applied Digital Workflow¹ course and requested for the Prof. Kjell Are Refsvik at the Høgskolen i Gjøvik² in Norway

Now days, it is difficult and time consuming to create Time-Lapse movies, resulting in a tedious and long work to perform. The main goal of this project and a challenge to overcome is to solve this problem by creating an effective and fast Automatic Workflow that would help users to create their own Time-Lapse movies in a shorter period of time and in an efficient way, and, in addition, having the opportunity to upload their videos to their iPods or even to post them on YouTube in a more easy way.

2 Automatic Workflow

What is an Automatic Workflow? An Automatic Digital Workflow is a Digital Workflow that will be executed with no need of participation from an user or with just a minimal participation from the user. In the previous course IMT4891 Digital Workflow Fundamentals we learned that a complete workflow consists in:

1. Collect data
2. Describe data
3. Transform/Transcode data
4. Move data
5. Present data

¹<http://english.hig.no/content/view/full/16725/language/eng-US>

²<http://www.hig.no>

Therefore the intention of this project was to automatize the process of the workflow and get an output much more faster and in a more automated way. The user will be on charge on collecting the data, taking the pictures in this case, and the script will handle the description, tranformation and transcoding of data, and prepare the data to be moved to a server or to an iPod.

In order to achieve this in a faster and efficient way, a shell script was created. The script handles and modifies the images in a preformatted way that the user does not have to worry about how to modify the images or adjust the settings for the video, this is done automatically by the script. Once the movie is created the addition of the music to the movie is done just by choosing the music the user wants to add to to the video.

Open Source software was used in the creation of this project for several reasons, such as:

1. Reduce the cost of creation, this means NO cost for the payment of licenses since the software is free to use
2. Open Source Operative Systems are more stable and faster than some operative systems like Windows
3. Open Source software and operative systems are Virus free!

3 Time-lapse movies

A Time-lapse photography (or movie) "is a cinematography technique whereby each film frame is captured at a rate much slower than it will be played back. When replayed at normal speed, time appears to be moving faster and thus lapsing. Time-Lapse photography can be considered to be the opposite of high speed photography"³.

But what is this exactly? The key point is take a large amount of pictures of one place or one thing from the same potition at the same interval of time. In other words, you set your camera to take a picture of the view from the window in your room every 10 seconds. After 5 hours you will have 1,800 pictures. Then, with the help of a software, you organize the pictures to be played, let's say, 25 every second. Now you will have a video of 1:12 minutes and will see that 5 hours in just something more than a minute.

This sounds pretty good, and it is, BUT there are some things that we need to deal with. First of all you need a good camera that is able to take high quality pictures. Also, there is needed that the camera could be set to take the pictures automatically, like with a remote controller or by software. In the following link you will find a great tutorial to make Time-Lapse movies:

Time-Lapse Tutorial - digitalartwork

³<http://en.wikipedia.org/wiki/Timelapse>

4 Planning

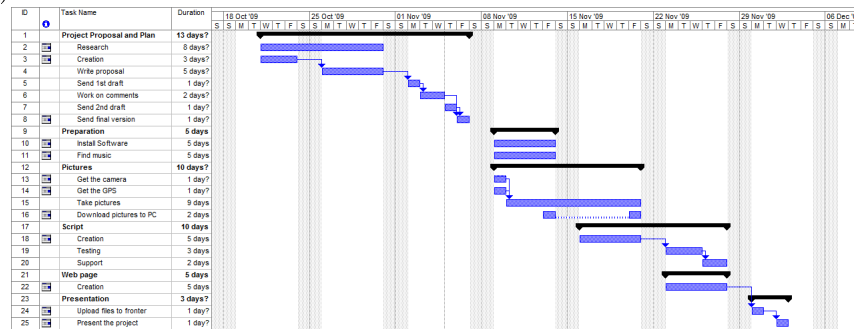
When you are about to make or design a project, it is always necessary to make a plan in order to prevent mistakes, problems or other things that could cause the project to end or to not to be a successful project and to keep track and don't lose the path you trace yourself. There is a saying that there is no perfect plan and always unexpected situations might come up, but when a plan is designed, it needs to be created thinking that everything is going to be perfect and also thinking that everything is going to be wrong in order to be prepared to everything that might come during the process of the project.

Here are some definitions of Plan according to Dictionary.com⁴

plan -noun

1. a scheme or method of acting, doing, proceeding, making, etc., developed in advance: battle plans.
2. a design or scheme of arrangement: an elaborate plan for seating guests.
3. a specific project or definite purpose: plans for the future.
4. Also called plan view. a drawing made to scale to represent the top view or a horizontal section of a structure or a machine, as a floor layout of a building.
5. a representation of a thing drawn on a plane, as a map or diagram: a plan of the dock area.
6. (in perspective drawing) one of several planes in front of a represented object, and perpendicular to the line between the object and the eye.
7. a formal program for specified benefits, needs, etc.: a pension plan.

This is the "*Perfect*" Plan that I created for my project (not so perfect after all).



⁴<http://dictionary.reference.com/browse/plan>

Comments on my plan

- I have to said that my planning was not perfect at all. It is funny that just the first part of the plan (Project Proposal) went well. A lot of unexpected situations came up during the project's development. In fact, I think that all the things that could go wrong went wrong, so I think that my project had a tendency to go wrong, but at the end was the other way around. Some of these unexpected things were:
- Since I was the first people to make a reservation for one of the cameras for the project, there were a lot of lack of information on how to respond or what to do in order that the person responsible of giving the cameras is not available. Unfortunately the day I was supposed to pick the camera up, the teacher got sick and he couldn't give me the camera, but he delegated the task to another person. We could not meet each other since he was not available at school when I was there and viceversa. He left the camera in the media laboratory, but since I have no access to the lab, I was not able to pick the camera. Fortunately I have a friend who borrowed me his camera to take the pictures
- Another problem that I had was that my laptop started to freeze every 5-10 seconds. Since I am using Ubuntu from a Virtual Machine, every time the laptop froze the Virtual Machine crashed and I had to restart the script and all the process again. Therefore I had to reinstall Windows and I lost a lot of valuable time.
- I also lost one whole day trying to install the library libx264 since my laptop was freezing all the time and this prevented to install the library and all the intructions that were available on internet didn't worked.
- One thing that was funny for me and I have never thought about it was the day I want to take pictures inside CC Gjøvik and take advantage of the crowd, but when I was about to start to take the pictures a police man told me that is not allowed to take pictures inside CC Gjøvik. I was shocked, but the law is the law, so I have to settle with taking pictures outside the building.
- I think the main difficulty I faced during the creation of the project was that I thought that the script creation would be more easy than it really was. I underestimate the capacity of FFMPEG and I have to deal with a lot of situations I didn't think of like all the settings you have to establish to create videos of one format in specific.

5 Implementation

This is how the Implementation of the project went.

1. Taking pictures

- (a) The first thing that I have to deal with was about where to take the pictures. The selection of the places was a key point for the project because the most basic script can be enlightened with a great video, or the best code can be opaqued by low quality video. Some key places were selected for the pictures like the Mjøsa lake and the CC Gjøvik.
- (b) Even though the city of Gjøvik have beautiful places, the main problem is that this is not a crowded city and to be able to take pictures with a lot of people or a lot of cars it is very difficult. Nevertheless, great sets of pictures were taken.
- (c) A Canon EOS 1000d camera was used to take the pictures. At the moment to take the pictures all the settings were handled by the DSLR Remote Pro for Windows Breeze systems software, except for the Auto Focus, that was set manually to Manual Focus.
- (d) With the DSLR Remote Pro software all the settings of the camera were set, like the White Balance, drag the shutter, the exposure time and the Time Lapse option as well.
- (e) The patience was a basic principle when taking the pictures. Sometimes I had to be on the cold wind without a proper jacket for an hour or so or be hungry for some hours just waiting the pictures were taken.

2. Code Development

- (a) In order to create the script, the shell script available on Ubuntu was used.
- (b) The first thing to deal with the code was about how to handle and modify the images. Thanks to our course of Digital Workflow Fundamentals, I already have the experience to use ImageMagick software and the convert command was used to resize the images and to create the intro and credits clips.
- (c) The ffmpeg software was used to create the videos from the images that were modified with the convert command
- (d) I had to do a lot of research about shell scripting since I had no sufficient knowledge on this language and it causes a small delay on the creation of the script.
- (e) Here is a basic step-by-step list on how the script works:
 - i. Name of the project.
 - ii. Validate name. If valid go to step 3, if not go back to step 1.
 - iii. Localization of the images (Path).
 - iv. Verify if it is a valid directory. If valid go to step 5, if not, go back to step 3.
 - v. Select a song to play during the video.
 - vi. Select the Frame rate (frames per second).

- vii. Validate frame rate. If valid go to step 8, if not go back to step 6.
- viii. Creation of the working folders.
- ix. Resize and rename images.
- x. Create intro images.
- xi. Create credits images.
- xii. Create avi movie.
- xiii. Create movie for YouTube.
- xiv. End.

3. Web Page creation

- (a) Creation of this web page. In order to create this web page I had to put together all the information recollected in the creation of the script and in the taking pictures process.

4. Final Report

- (a) In order to create the Final report, a research about how to write academic reports was made. One of the most important things to have in mind when writing an academic report, is to know who is going to read the report and try to make it understandable to all people that will review or read this report.
- (b) After some research and reviewing some academic reports, I started with the creation of the final report on LaTeX with Lyx.

Results

The results of the project were:

- A script that creates Time-Lapse videos
- A Time-Lapse video that was uploaded to YouTube Time-Lapse Video.
- The location of the places on the images in a Google Maps window. This was possible to the execution of the script made by ian james daniel gonzales (090285) also for this course. **Thank you Daniel!**

Conclusion

First of all, the plan didn't go as I expected. I faced a lot of difficulties and problems that prevent me to finish this on time. I think that at some points of the project I thought it was too easy that no much time were required to finish this. When I realized I was wrong, it was too late.

I would like to state that I liked a lot the making of this project. It was very funny and at some points it didn't looked like a school project. I learned a lot in some fields that I had no experience at all, such as

- the creation of videos
- manipulation of images in a more professional way
- I didn't know almost anything about photography, even though my sister is studying the bachelor in photography
- how to change all the settings on the camera is amazing. I had always thought that I just needed to take the camera and press the button to take the picture, and that's it, maybe to focus a little the image, but this opened my eyes in how this works.
- I increased my knowledge in shell programming. I had a few lessons of Linux before this and shell programming, but never this deep. I had to investigate a lot of commands, how to use the loops and even to extend my knowledge on the most basic commands such as ls.
- I liked this a lot and enjoy it so much that I think that I will have a new hobby!

I think that this project would need a further work, first of all, to cover all the details I had left or were not covered the right way for lack of time like validation of files, no image files, and so on. I see some strong points where the script could be improved and maybe some other changes that may improve the performance in the execution.

One problem that I have is that my laptop was failing a lot and it froze every 5 or 10 minutes, so when I had to create the final video and process almost 2,500 pictures it took more than 2 hours to create the video!

I also think that in addition to this, some other features could be added to the script in order to make it more interesting, but this could be against the purpose of the script, since the interaction with the user should be minimum. I guess that even though the script does its work it could do better. I would like to have the opportunity to work on this script and improve it even though it is not longer my project to deliver. I think it would be nice if the teacher gives me some feedback about the script and let me know what can I fix or improve of this script, because I think it is a good tool to create and manipulate videos... using a better computer.

Legal

Everything on this document and all the files created for this project are under the BY-NC-SA license by Creative Commons.