Final Year Project 2018-2019

Intelligent Photo Gallery

Detailed Project Plan

Kwan Hoi Shun
UID: 3035280972

Supervised by Dr. Anthony Tam

1.0 Introduction

1.1 Project Background

Nowadays, people are used to take a lot of photos with their smartphones. Many people have accumulated a large collection of photos and is in need of a photo gallery application to help organizing and managing their own collection of photos.

1.2 Existing Solution

Currently, there many social network application and photo gallery application that can fit in the role as a photo gallery application. These application provide good photo management features such as albums and face recognition to help organizing the photos. However most of them have disadvantages in regard to storage size, image quality and privacy issues.

For example images store in Facebook are of reduced quality, and in Google Photos you are facing a choice between unlimited storage and reduced photo quality when used for free. Furthermore, users may not want to sacrifice privacy on their personal photos to store them on a cloud storage where users does not have full control. Storing on cloud storage also means that accessing large amount of photos can be slow due to limited bandwidth.

1.3 Project Objective

The intelligent photo gallery aims to protect user's privacy by storing them in a private storage device and improve the photo management experience by introducing efficient face recognition and innovative photo management features.

The gallery application allows users to choose storing their photos in their local device or a Network Attached Storage (NAS).The NAS device is essentially hard disks connected to the local network, this allow users to access the photos without bandwidth limitation, and the privacy of the photos is ensured as the photos are stored locally.

Adding tags and sorting photos into albums can be a tedious job. The intelligent photo gallery strives to reduce most of the photo management workload by allowing users to set up filters for adding tags and albums to newly uploaded photos. An efficient face recognition algorithm will also be employed to allow users to search photos more accurately.

2.0 Project Scope

2.1 Features

The application will contain the following features with a user friendly GUI interface.

2.1.1 Face Recognition

Two different approach of face recognition will be compared for their accuracy and efficiency. The approach with a reasonable accuracy and high efficiency will be chosen as the face recognition algorithm for the application.

If time is allowed, improvement for age progression will also be tested. The improvement will only added to the application if it provide significant improvement in accuracy and require relatively small computation time.

2.1.2 Photo Searching

Users can search for specific photos by selecting multiple criteria, including the date, people, tags and album. Users can then select multiple photos based on the search result and apply the same action over the selected photos.

To increase the search efficiency, searching conjunction(AND, OR, NOT) can be used to specify the relationship between multiple criteria.

Users can also search for duplicate photos to remove duplicates.

2.1.3 Tags and Albums

Users can create tags and albums to help organizing photos. Tags is used for searching the photos more efficiently, users may tag photos with the same feature such as food, nature etc. If there are remaining time after implementing other features, an auto tagging feature will be added based on the events on the Google Calendar. For example, automatically tagging all the photos you have taken from a trip that is registered on Google Calendar.

Albums in the other hand, is used for grouping relevant photos together for viewing. For example, grouping photos in the same trip or event. By organizing photos into an album, user can then view the photos in the album one by one in full screen.

2.1.4 Photo Upload

Users can import the photos into the storage location using a photo upload interface, the interface will allow users to specify the tags and albums to be added to the photos uploaded.

Alternatively, the user can also set filters to automatically add tags and albums to uploaded photos. The filters are similar to filters on Gmail which allow user to specify the criteria (i.e. date, person identified etc.) and apply specified tag or album to the matching photos.

2.1.5 Storage Options

When the application is first launched, users can select the storage option. The storage options are as follows:

- Local Storage
- NAS (Network Attached Storage)
- NAS storage with personal cloud configuration (Personal cloud)

2.2 Features Priority List

Feature	Priority
Face recognition	1
Face recognition with age progression	4
Searching with multiple criteria	1
Apply same action to multiple selected photos	1
Search for duplicates	2
Add and remove tags in photos	1
Auto tagging using events in Google Calendar	5
Create, add and remove photos from albums	1
Upload photo to the gallery	1
Automatic filters	2
Storing in local storage and NAS storage	3
Storing in NAS storage with personal cloud configuration	5

* The priority range from 1-3, with 1 being the most important and 5 being least important.

2.3 Project Deliverable

A photo gallery application that allow users to store photos and manage the photos efficiently, the application will be compatible on both Windows and Mac.

3.0 Methodology

3.1 Face Recognition

Two different approach of face recognition will be tested and evaluated based on their accuracy and efficiency. The approaches can be divided into 3 stages, namely face detection, feature extraction and classification.

The face detection stage will be done using OpenCV due to time limitation. OpenCV provides 2 different method for face detection, the HAAR classifier and the local binary pattern classifier, both method will be tested to determine which will be best fitted for the application. The detected face will then be extracted and processed before moving on to the feature extraction stage.

The feature extraction stage is the stage where the 2 approach differs from each other, the details on the extraction stage are explained in section 3.1.1 and 3.1.2. After feature extraction, a classifier or clustering algorithm will be used to classify the person in the photo using the extracted features.

All of the training in the feature extraction phase will be done during the development process, the resulting application will use the trained model for feature extraction. This is to reduce the computation time required to update the feature extraction model. However the classifier / clustering algorithm used in the classification phase will use an incremental learning approach, that is to update the model for each batch of new photos uploaded to the gallery application.

3.1.1 Face Recognition using machine learning to extract facial features

This approach will use a deep convolutional neural network to select facial landmark from the extracted facial image. During each training iteration, 2 photo of the same person and a photo of a different person will be used, the neural network will then be tweaked slightly such that the feature points of the same person will be more similar and vice versa.

3.1.2 Face Recognition using ratios between facial landmarks as features

This approach will use a face landmark estimation algorithm to locate the facial features (e.g. eye, nose, mouth). The ratios between different facial landmarks (e.g. eye to nose distance to eye to chin distance) will then be used as features for classification. This approach is invented by Ramesha, Raja, Venugopal and Patnaik, however in the original approach the facial features are located using cranny edge detection [1].

3.2 Datasets

The VGGFace2 dataset created by Q. Cao et al. [2] will be used for training and testing the face recognition algorithms. The dataset is selected as it have a large amount of photo per individual (362.6 photo per individual) and have information on the age of the individual in the photos for training for face recognition for age progression.

The Helen dataset created by Vuong Le et al. [3] will be used for training in the face landmark estimation algorithm, the dataset is selected as it contains 2230 images each annotated with 194 accurate marked facial components.

3.3 Development Process

The development of the Intelligent Photo Gallery will be divided into 3 stages, ‘Design’ stage, ‘Implementation’ stage and ‘Optimization and testing’ stage.

3.3.1 Design Stage (Sept to Oct)

During this stage, the features of the application will be finalized and preferences on development tools will be selected. The logic flow of each use case, and the application design will also be settled in this phase.

3.3.2 Implementation Stage (Oct to Mar)

The application will implemented in 3 parts in this stage, basic functionalities, face recognition and storage options.

a) Basic functionalities

The basic functionalities covers the photo upload, search and management features, tags and albums are also included in this part. The features will be implemented progressively, from implementing simple functions such as upload and search to implementing more complicated functions such as filters and duplicate search.

b) Face recognition

Each face recognition approach will be tested for efficiency and accuracy, then the final approach will be selected based on the efficiency and accuracy.

c) Storage options

This part covers the alternative storage options other than storing in the local storage. The storage part is separated from the basic functionalities so that implementing new storage options will not affect the basic functionalities of the application. The NAS storage options will be implemented in this part, if there is remaining time, other storage options such as personal cloud will also be implemented.

3.3.3 Optimization and Testing Stage (Mar to Apr)

In the final stage, all the functionalities of the application will be tested. Bugs found in this stage will be fixed and the application will be optimized to enhance the performance.

3.4 Development Tools and Libraries

TensorFlow will be used for constructing the neural network, it is chosen as it is the most commonly used deep learning library and is well documented. It is also

OpenCV will be used for image processing and face detection as it process image faster than Pillow and is well documented.

Python will be used as the application language as Python has a large collection of packages for dealing with different problems especially in Deep learning.

4.0 Schedule

4.1 Overall Schedule

Phase and Time	Work	Milestone
Inception (Sept 1 - Sept 30)	- Research for previous literature, relevant APIs - Determining the features in the application - Planning the project schedule - Write up the project plan - Setup project website	- Finish the project plan
Elaboration (Oct 1 - Jan 20)	- Design the use case and the logic flow in each of the features - Build the basic functionalities with priority level 1 - Build the face recognition prototypes for comparison - Write up the interim report	- A demo application with the basic functionality of the gallery implemented. - Trained models of the face recognition prototypes. - Evaluate the face recognition results by comparing the trained models - Finishing the Interim report
Construction (Jan 21 - Apr 14)	- Integrating the face recognition prototypes into the application - Adding more storage options to the application - Implement automatic filters and basic functionalities with priority level 2 - Optimization and testing - Write up the Final report	- Final application - Finish the final report

4.2 Feature implementation Schedule

Task	Subtask	Expected time	Expected finish date
Implementing the important basic functionalities (Priority 1 features)	Design the use case and logic in the important basic functionalities	2 weeks	Oct 16
	Implementing the upload and search function	2 weeks	Oct 30
	Implementing the tags and albums	2 weeks	Nov 13
	Testing	1 week	Nov 20
Implementing the face recognition feature	Figuring out how to implement the face recognition algorithms	2 weeks	Dec 4
	Developing the face recognition prototypes and training the models	5 weeks	Jan 8
	Integrating the trained models into the application	2 weeks	Jan 22
	Testing	1 week	Jan 29
Implementing the less important basic functionalities (Priority 2 features)	Design the use case and logic in the less important basic functionalities	1 weeks	Feb 5
	Implement filters	2 weeks	Feb 19
	Implement duplicates search	1 week	Feb 26
	Testing	1 week	Mar 5
Implementing other storage solutions (Priority 3 features)	Figure out how to store photos in the NAS storage device	1 week	Mar 12
	Implementing the NAS storage option and testing	1 week	Mar 19
Final Testing and optimization	Testing and Optimization	4 weeks	Apr 14
[Extra] Priority 3+ features		If time is available	If time is available

5.0 References

[1] K. Ramesha, K. B. Raja, K. R. Venugopal, and L. M. Patnaik, “Feature extraction based face recognition, gender and age classification, 2010.

[2] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, “VGGFace2: A Dataset for Recognising Faces across Pose and Age,” 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), 2018.

[3] V. Le, J. Brandt, Z. Lin, L. D. Bourdev, and T. S. Huang, “Interactive facial feature localization,” In European conference on computer vision, pp. 679– 692, 2012.