Card Free Service

Project Report for CSE 535 Numerical Computation

Fall 2017

School of Computer Science and Engineering

California State University San Bernardino

Adeeb Alqahtani

**November 30th 2017**

# 

# 

# 

# **I. Introduction**

Using Non-negative Matrix Factorization (NMF), we implemented a face recognition program. A picture of a person that is taken will be evaluated and, ideally, correctly identify that same person that is in the database. This program will implement a “Card Free Service” at California State University, San Bernardino, where students would not be required to carry and show an student identification card anymore. Upon initial registration, all students will a take a picture to create a virtual student ID card and be saved in a database. When required to show ID at such locations like the financial aid, library, and others, the students have a picture taken that will correctly match up with their virtual ID. The system will return a image, name, identification number, and date of birth of the student to be verified. This program will require no material to be used, i.e. plastic and ink, no extra responsibility on the student to carry a ID card, and a more automatic and reliable way of identifying the students. The goal is to create a more efficient way to identify students, using a reliable user friendly system that can be setup and installed wherever necessary across campus.

**II. Related Work**

In the paper by the Institute for Infocomm Research by Menaka Rajapakse and Lonce Wyse­­, facial recognition is approached by Non-negative Matrix Factorization (NMF) algorithm. The authors behind the scientific paper emphasize that NMF does not add any negative values when factoring the matrix so there’s no cancellation of any sort. Having only positive constraints, they were able to localize facial features thus having a collection of features and representing a face. They start with a function, *F*,that is a data matrix and *W* and *H* are the decomposition and all positive values. *W* and *H* go through an iterative process to update the matrix that goes through multiplication and summation of positive values which converges to a matrix to represent a face. Contributions include comparing NMF and ICA algorithms in accuracy. Through their analysis­­­ one can see that NMF out performs PCA and ICA when the rank is 25,49,81,121 and 144 thus being the optimal algorithm.

Classifying Faces with Non-Negative Matrix Factorization by David Guillamet and Jordi Vitri`a discuss the obstacles when doing facial recognition under different lighting and facial accessories. A data matrix is where *W* and *H* are approximating to the original matrix. In conclusion, under specific feature such as same background or angle, NMF was able to return high recognition rates than compared to PCA. Having specific features allows NMF to reflect input data as global. Having different variations such as lighting, accessories, noise affects the accuracy of NMF.

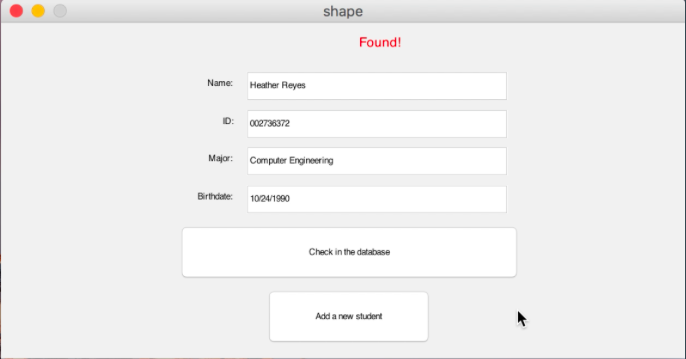
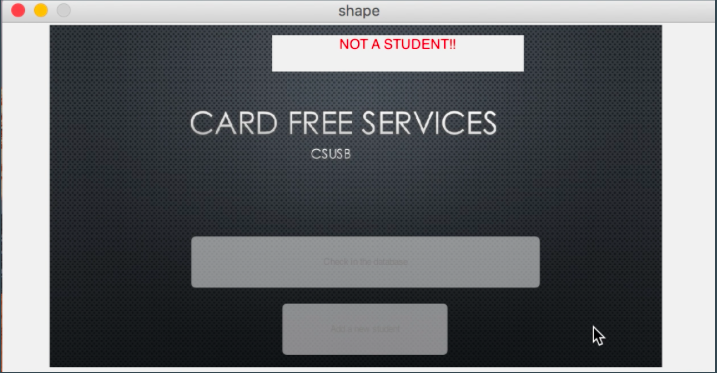
**III. Methodology Framework**

The software we went about in our project was using Matlab and the predefined AT&T face database. The database will contain 40 directories with each one belonging to a unique individual and a 10 sub directories containing different variations such as angle or face expressions. Our project consists of 3 files that will load the database, face recognition, and main program. In our load database file, we created a zero matrix with the rows to represent the resolution (112\* 92) and the columns as the number of photos. This will act as the matrix to contain our database with all our pictures. We implemented a nested loop to go through each directory and its sub directories to read/reshape. The first column will contain the 10 pictures of the first person. The face recognition file will load the database, a test image, and reshape the image to match the size of our database (112 x 92). The main file will consist of a graphical user interface that will allow the user to add a new student to the database, check if the student is already in the database, else output a “not found” message.

**IV. Results and Discussions**

We wrote a MatLab program using Non-negative Matrix Factorization to perform face recognition. As for a data set, we chose to download one of the AT&T preset data sets. We also implemented the ability to use a webcam to capture a live face image and return the same person from a database that can be altered as needed. Our user friendly interface consist of two buttons for two options - “Add a new student” and “Check in the database.” The process would be to add a student first by taking 10 photos, then entering the students name, student identification number, major, and date of birth, and photos and information are saved in a file.

The idea is that a virtual student identification card is created and saved in the database. So when a student is asked from a form of identification, no actual ID is required but simply a photo is taken to pull up a the virtual ID. That would be the second button on the friendly interface that checks if the student is in the database. When the student needs to be verified, a photo will be taken to identify the student as a student, if they were already in the database, or not a student, that has not yet been added. When a student is correctly identified, “Found” is appears on the screen along with the students information and if the volume is on the system will say “Student’s name is “ followed by the student’s name. When the student can not be identified, a notice is displayed and read out loud saying “NOT A STUDENT.”

We tested our data multiple times to get an accuracy rate of 87.5%. We added three of the four in our team to the database. With various different face positioning angles, the three off us took 10 photos switching it up each time with changing up glasses, hats, hair style, etc. We got a successful find of 25 out of 30 attempts. The fourth person took 10 photos also and was successfully not found 10 times since they were not saved in the database. So of the 40 attempts, 35 were successful. The rate is not that high but we need to take into consideration that the images tested were live and not that of a present data set. The environment is also a factor, ie lighting, but we did keep the background consistent to resolve the issue of too much noise in the background that the program will pick up and attempt to match as well. We wanted to change up our appearance a moderate amount, considering that the student’s appearance will slightly change from day to day, to test the strength of the program. We hope to improve the accuracy with improving the code where needed.

**V. Problems and Improvements**

First we will begin by saying this, the overall project (thought process and implementation process) was a bit time consuming, although an accomplishable task. With that being said, there are some “ups and downs” that we had to overcome in order to get a successful running project. The first issue being, “What is the goal of the project?” Everyone immediately began researching about Non-negative Matrix Factorization and how it can be applied. Time was consumed in researching what could be accomplished between NMF and SVD based PCA for Facial Recognition. For example, SVD factors contains both positive and negative entries while NMF factors are strictly positive. SVD also yields unique factors whereas NMF factors are non-unique. –Quora. Given guidelines of the project and the amount of time that is allotted to completing the project, based on research, we decided on Facial Recognition using NMF.

The next step was how to demonstrate Facial Recognition. Originally, we had decided on using a predetermined database with images with set values that were downloaded from the Web. Eventually, that transformed to using a database with images containing values from the members of the group. Using MATLAB, the ability to scan our very own faces and determine each user was implemented into a mock ID Scanning Service for CSUSB.

Although this is a very unique system, there are some compromises with the program. Imagine this Scenario, someone who takes their photograph for the software to scan in a certain background. When attempting to be recognized, it has a chance of being unsuccessful due to the fact that the variance in the intensity of the photograph as a whole. The program simply recognizes facial features using NMF, but it also picks up the actual background itself (this in computer scientific terms is called “noise”). Although the success rate is high utilizing by replicating the same conditions as the original photograph, the program fails to recognize the face alone. It outlines the whole entire image therefore skewing the results.

One way to possibly fix the scenario with a more allowed time period would be narrowing the area of the image window that is processed. By doing that, the program will have organized structure in which only the outline of an actual face is picked up rather than any superabundance when the image is processed. Another thing to improve on is the way the database is loaded. Reshaping occurs thus updating the database every time it executes. This causes severe drag in the process time. One possible way to alleviate the issue would be storing the database onto a high efficiency server therefore speeding up the process time. Overall, the program is successful in execution, although there are certain improvements that could be made in order to increase overall coherence of the project. With that being said, this provided for an excellent learning experience!

**VI. Benefits**

The overall project has taught the whole group about the application of Non-negative Matrix Factorization and how it can be applied in real world applications such as Facial Recognition. As far as knowledge wise, we learned that faces can be represented as a collection of distributed parts. Non-negative Matrix Factorization is utilized in order to demonstrate how these localized features can be represented. Information that were gathered during the project will be very beneficial with future application as not only have we learned about NMF, but how its application is different as compared to PCA or SVD (which were also other types of matrix factorization techniques offered for the project.) “NMF uses the decomposition of the matrix F into two matrices W of H of size nxr and rxm.” –Institute for Infocomm Research.

**F = WH**

Throughout building our knowledge base, we were able to expand our skill sets as well in the aspect of being able to apply NMF for Facial Recognition. A database with Images were produced and using MATLAB, three resulting programs were constructed in order to implement the data. The program that specifically enhanced our skillset was the faceRecognition.m as demoed in the final project. In this program, the application from the research of NMF is shown. For example, choosing an image to grab its vector form and locating it from the database.

**T = reshape(img,u(1)\*u(2),1);**

**[W,H] = nnmf(V,16);**

**W = double(W);**

**H = double(H);**

Technologically, our own databases (brains) became enlarged with a variety of uses for Facial Recognition. Our efficiency of MATLAB was substantially increased as well as our thought process upon the subject of Facial recognition. Facial recognition can improve real world services such as applying them to systems allowing an easier checkout process when shopping, in which McDonalds in Asia are experimenting with currently. They will allow Law Enforcement to operate a lot more effortless in identifying subjects who have violated the law and misconducts in public. Then again, it is a bit of alarming subject as well that hackers can gather data from databases and misuse the information provided. With that being said, Facial Recognition can have a lot of benefits presuming that more research will be continued in order to have an efficient working model that “safe, efficient, and reliable”.

**VII. References**

*Algorithms in Nature*. Carnegie Mellon, [www.cs.cmu.edu/~02317/slides/lec\_7.pdf.](http://www.cs.cmu.edu/~02317/slides/lec_7.pdf.)

Bharadwaja, Sameera H. *What Is the Difference between Non-Negative Matrix Factorization and Singular Value Decomposition?* 21 Dec. 2014, [www.quora.com/What-is-the-difference-between-non-negative-matrix-factorization-and-singular-value-decomposition](http://www.quora.com/What-is-the-difference-between-non-negative-matrix-factorization-and-singular-value-decomposition).

Guillamet, David. *Non-Negative Matrix Factorization for Face Recognition*. 2002,

<http://xrm.phys.northwestern.edu/research/pdf_papers/2002/guillamet_lectnotcompsci_2002.pdf>

Rajapakse, Menaka, and Lonce Wyse. “Face Recognition with Non-Negative Matrix

Factorization.” Institute of Infocomm Research. *Zhome.org*, 21 June 2016, [www.zwhome.org/~lonce/Publications/publications/1838\_1\_FaceRecognitionWithNMF.pdf.](http://www.zwhome.org/~lonce/Publications/publications/1838_1_FaceRecognitionWithNMF.pdf.)