

Facial Detection and Recognition Study for Security Applications

A Capstone Project by

Ali Arefi-Anbarani : oarefi@umich.edu

Ezgi Gumusbas: ezgi@umich.edu

Yujin Lee: yujinlee@umich.edu

Master of Applied Data Science



Table of Contents:

1. Introduction	3
1.1. Problem	3
1.2. Aim	4
2. Data	4
3. Methodology	5
3.1. Anti-spoofing	5
3.1.1. Preprocessing	5
3.1.2. Data Augmentation	5
3.1.3. Local Binary Patterns (LBP)	6
3.1.4. Histogram	7
3.1.5. Machine Learning	7
3.2. ArcFace Modeling	7
4. Results & Discussion	8
4.1. Local Binary Patterns & Histograms	8
4.2. Results of Lazy Prediction	11
4.3. t-distributed stochastic neighbor embedding (t-SNE)	12
4.4. LDA Model	13
4.5. ArcFace Model	14
4.5.1. ROC Curve	14
4.5.2. Face by Face Distance Matrix	15
4.6. Ethical and Security Considerations	16
4.6.1. General Security Cautions of Biometric Identification	16
4.6.2. Ethical Considerations	16
5. Conclusion & Future Improvements	17
6. Statement of Work	18
7. References	18
8. Appendix	19
Appendix I. Confusion Matrix of KNeighborsClassifier	19
Appendix II. Confusion Matrix of Ridge ClassifierCV	19

1.Introduction

Computer vision has many applications from a security standpoint. Facial recognition is one of the most common applications of computer vision. It is widely used for biometric identification. According to the U.S. Department of Homeland Security (2022) biometric identifiers (like face, voice, iris, fingerprint, etc.) are unique characteristics that can be used for automated recognition. Face recognition is commonly used to unlock phones, access smart phone applications which contain personal information like bank apps, health record apps, etc., access to buildings such as offices and high-security facilities, to compare the passport photo with the holder's face at the border checkpoints, to identify the criminal in the crowd, to give access to only authorized people in defense services as security precautions.

There are various advantages of using face recognition softwares. The first advantage is that facial recognition is a smooth and quick verification process. Compared to other biometric services like fingerprint, or compared to access with passwords, it has a faster process. Besides, it is widely used in built-in cameras of smartphones, tablets and computers, so it is easy to integrate to the applications. Like voice and fingerprint, everybody has a unique face that improves the security of these processes. Facial recognition analysis uses specific geometry features in the faces, such as the shape of chins, eyes and ears, the distance between the features of the face like nose, forehead, mouth, eyes, etc. As a result of using these specific geometric features and expressions, face recognition is accepted as one of the secure ways of identification. Live detection by cameras prevent the security systems from being hacked from photographs of the people.

1.1. Problem

Although it might be more difficult to hack biometric identification than passwords in most cases, face recognition is still a challenging technology because it can be hacked in different ways. For example, if there isn't live video detection, it is a challenge to differentiate the people's faces rather than photos of these people. Besides, sometimes it needs consistent lighting, positioning and resolution. For example, it might be challenging to detect a person from a side view with hat and beard from the front photo that has no hat and beard, or mask.

Facial spoofing is a popular risk of facial recognition systems. According to one of the very common user identification systems providers named eID,a Signicat Company(2022), "Facial spoofing is the act of using a person's face and simulating their facial biometrics with the use of a photo or video to steal their identity (becoming a digital identity theft)".

Attacking Methods

Most common attacking methods for face spoofing are presenting to the camera a photograph, a video, or a 3D mask of a targeted person and the photo attacks are the most critical type of attack because of some factors like ease and cheapness (Hernandez-Ortega et al.,2019). For example, to print a color image of the person's face is not difficult to do and it is a much cheaper option than generating a 3D mask. Besides, it is easy to find people's photos especially in social media accounts. If there is no mouth movement, eye movement and blinking detection in the camera, the system becomes more open to attacks. Similar to photo attacks, video attacks are also common, especially by using the live videos of the person in social media or they can be taken by using hidden video cameras in public places. The liveness feature makes the attack more difficult than photo spoofs, because in the video eye blinking and movements of facial features are considered. The 3D mask attack means that the attacker makes a 3D mask of the user's face and uses this mask to hack the system. It is a more complicated attack method than photo and video because it needs a special skill to build a good imitation of the user's face mask.

In this project, we focused on the photo spoofing method because it is widely used by the security systems and it is more easy to break the photo security than live methods like video. Most of the video and high-level live detection technologies are very expensive for the users. So, we want to focus on the cheapest and widely available one to find a solution to the spoofing problem and make a robust face detection system.

1.2. Aim

Our aim in this project is to make a robust facial recognition system by using state of the art models to achieve high performance and solve the problem of spoofing. Our goal is to answer the question of how we can authenticate a user with high accuracy and detect the spoofing activities.

2. Data

The dataset contains headshot photos of people in the plain background from [Adobe stock](#), and can be found [in this Drive folder](#). The 'zip' file of augmented images of this dataset can be found [in this Github repository](#). It can be downloaded by clicking 'view raw' colored text on the page.

For the ArcFace model, the test dataset contains headshot, half and [full body shot photos of Chinese people](#) in the various backgrounds from V3 dataset, and can be found [in this Drive folder](#). ArcFace is the pre-trained model, and the [celebrity dataset used](#) for training purposes.

3. Methodology

To solve the problem of face recognition, we divided this project into 3 main steps. The idea behind this decision is to differentiate the face detection part, anti-spoofing part and ArcFace modeling part.

3.1. Anti-spoofing

3.1.1. Preprocessing

Stock images of passport-like photos were collected from adobe stock. To improve the reliability of the system we aimed to use as uniform photos as possible. We used uniform photos like a passport or driver license style with a plain background. Also, we gave importance to using the images that are the representative of all racial ethnicities. We wanted to use diverse data from the ethnicity perspective. Additionally, our dataset has a balance of genders and age groups.

Some images from the dataset were printed using a black and white HP LaserJet P1006 printer. It has a 600 x 600 dpi and 1200 dpi effective output.

We used black and white prints because the LBP algorithm needs images to be in grayscale rather than color. Also, original color images were converted to greyscale.

Original photo images were labeled as “Photo” and black-white printouts of original images were labeled as “Print”. Also, these print images needed to be resized to match the original photo pixel dimensions, so we resized them down by a factor of approximately 1000. For example, the original dimensions of print photos were 4000 by 3200 pixels for each photo, at these dimensions, our LBP algorithm would be severely slowed down and thus pixels were resized to values between 300-400 by 300-400. The original Photo images were also in this range as well(300-400 by 300-400).

3.1.2. Data Augmentation

Data augmentation was performed to have a larger dataset, based on the principle that a larger dataset will better test the model. The techniques that we used:

- Auto-enhancement or brightening of the image
- +90 degree and -90 degree rotations
- Vertical and horizontal flips (flips image on x-axis and y-axis respectively)

For train dataset:

- Vertical and horizontal flips (flips image on x-axis and y-axis respectively)
- +90 degree and -90 degree rotations

These augmentation techniques helped us to have a larger dataset and give us a chance to test our algorithm in different ways like different angles and rotations.

After augmentation the train and test set had 1093 images and 248 images respectively. This is an approximately 80% to 20% split of the total 1341 images.

3.1.3. Local Binary Patterns (LBP)

We used local binary patterns to differentiate between original photos and printed images. Local binary patterns is a texture classifier based on a gray scale co-occurrence matrix that computes a local representation of texture in an image. It computes a local representation of texture by comparing each pixel with its surrounding neighborhood(Dayala,2020).

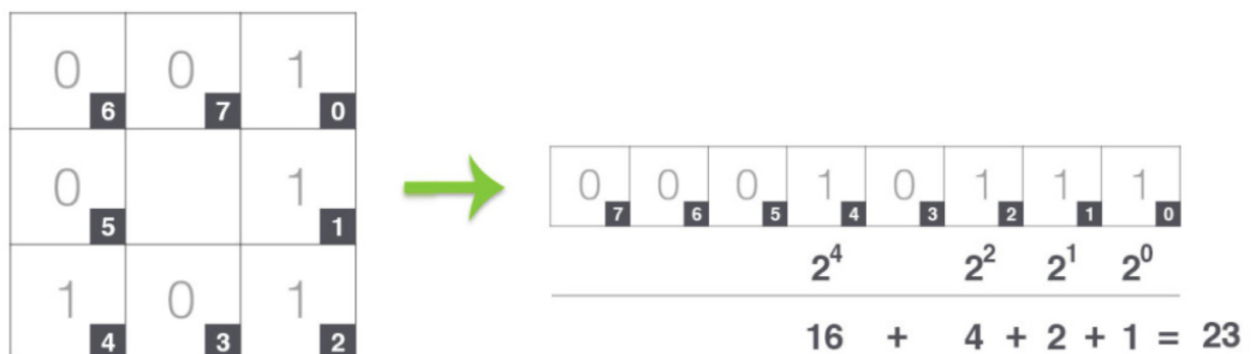
According to LBP, each pixel has a radius r and points p ;

- p number of points within that neighborhood
- r determines the size of the neighborhood
- The larger the p , the more patterns you encode
- The larger the r , the more texture details you capture

Unique rotation invariant binary patterns occur, and its amount is determined by r and p , which are multiplied to get the amount of patterns. The pixel of interest value acts as a threshold, if the neighboring pixels around it are less than the threshold, they are assigned a binary value, and if they are equal or greater than they are assigned another binary value. This creates a binary pattern around the pixel of interest. The positions of the neighbors determine a numerical placeholder (2^n) where n is the position around the pixel determined by the algorithm, if the pixel is a 1, that position is active and will be involved in the final summation that determines the value of the pixel of interest after applying LBP, if it is 0 it will not be in the sum. For example;

Figure 1

Taking the 8-bit binary neighborhood of the central pixel and converting it to a decimal representation.(Dayala,2020)



Adapted from Dayala, R. (2020, July 23). *10.7 local binary patterns*. Computer Vision. Retrieved December 11, 2022, from <https://cvexplained.wordpress.com/2020/07/22/10-7-local-binary-patterns/#:~:text=LBP%20compute%20a%20local%20representation,r%20surrounding%20the%20center%20pixel.>

3.1.4. Histogram

After local binary patterns, we create feature vectors to see if there is a difference between the digital photo and the printed(paper). We plot them on the histogram to notice the difference. The LBP algorithm histogram gives us the frequency of the normalized features and we can see distinct features for each category. These histograms can be found in Results & Discussion sessions with detailed explanations.

3.1.5. Machine Learning

We use lazy predict classification to do multiple classification models. We get 100% accuracy for certain models. So, to prevent overfitting, in this case we considered using K-fold cross validation to get a more accurate result. And, we performed k-fold cross validation to get a better understanding of the performance of our model.

One of the best performers was RidgeClassifiers, which is a classifier that uses ridge regression and the other was LinearDiscriminantAnalysis(LDA). Ridge regression uses a bias fit of line through the training data that produces less variance and the LDA model finds the direction of maximum class separation. Besides, LDA works to reduce variance like Ridge regression did. It is another reason that LDA works with Linear Binary Patterns well (Fratric & Ribaric, 2022).

3.2. ArcFace Modeling

Academic paper by Shi (2021) addresses different algorithms and models for face recognition and explains the limitations of models. We used this paper to prevent time and effort waste and narrowed down some choices depending on what tradeoffs we deem acceptable for our application. We found the ArcFace model useful for our case. ArcFace is a computer vision-related machine learning model. It compares the similarity of two faces. As an input, it takes two face images and the output of the model is the distance between these two faces. By this way, it returns how likely they are the same person.

“Without training any additional generator or discriminator, the pre-trained ArcFace model can generate identity-preserved face images for both subjects inside and outside the training data only by using the network gradient and Batch Normalization (BN) priors. Extensive experiments demonstrate that ArcFace can enhance the discriminative feature embedding as well as strengthen the generative face synthesis.”(Deng et al., 2022).

We used ArcFace modeling to find the similarity between two faces to give an access to the user or not. The certain threshold is defined to let the user access with the similarity value. ArcFace is a pre-trained model by celebrity faces(CelebA). We used our test set on Chinese nationality faces to test how the model is doing with different data because celebrity dataset

contains more American faces. Data sources used can be found in the data session of this report.

The ArcFace model extracts features from an object and creates a 512-dimensional embedding. The distance is calculated from the selected features to compare whether the two human features are the same. In this process, we put the Chinese dataset in, compared several threshold results, selected the threshold that produced the best result, and tested the final result.

4. Results & Discussion

4.1. Local Binary Patterns & Histograms

Figure 2

Feature Vector Distribution of Original Images

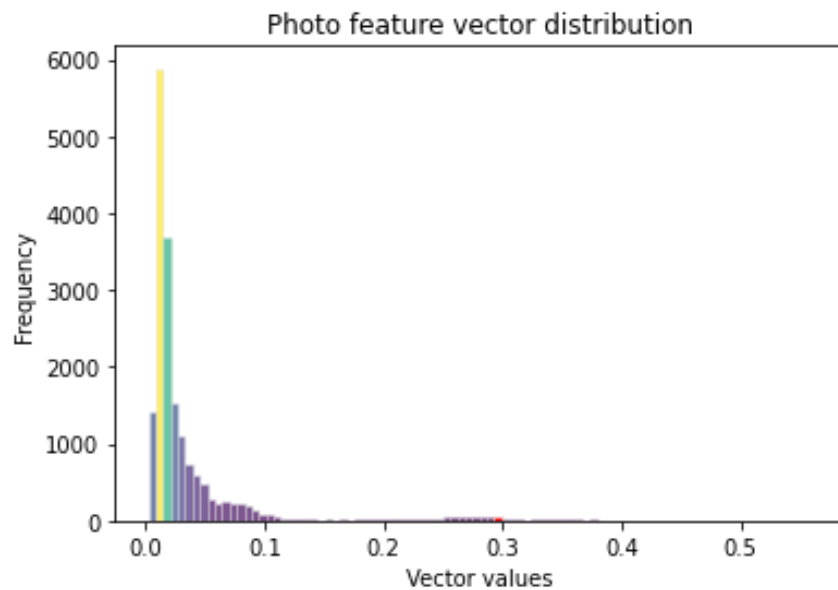
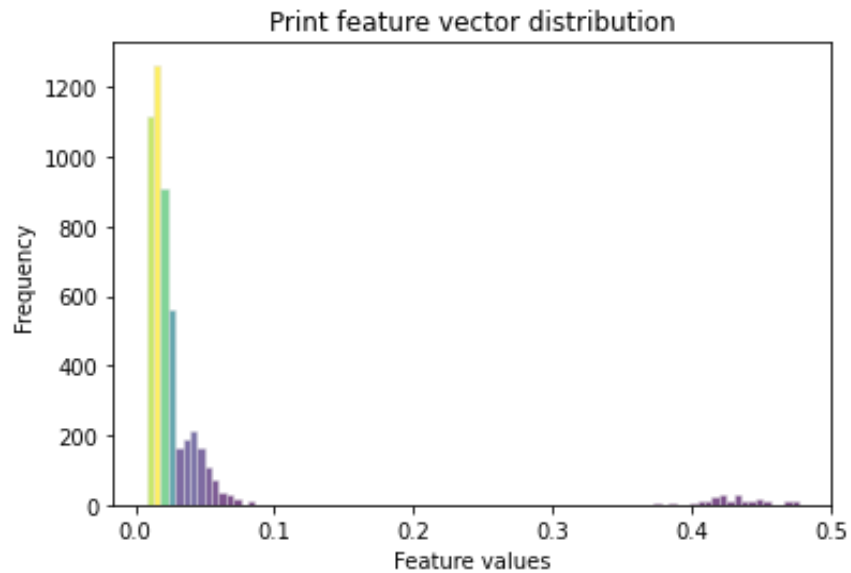


Figure 3

Feature Vector Distribution of Printed Photos

































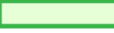
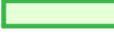




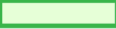
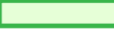




These figures show how the vector values of the images are distributed by frequency. X-axis represents the values of features of the vector and y-axis represents the frequency of them.

As it is seen from Figure 2 and Figure 3, the original images vector distribution is very different from printed images' vector distribution. It proves that LBP did a good job on texture differences. Although the distribution of feature values between 0.0 and 0.1 looks similar, the right side of the distribution is totally different. In real images, the right side tail is distributed between 0.2-0.3, but in the printed photos, the right tail is distributed between 0.4-0.5 mostly. This gives a distinct separation of original images and printed photos. Photos vs print observations are well separated in the feature space.

Figure 4

Overview of Anti-Spoofing Solutions

	Static PAI	Dynamic PAI	Image input	User involvement	Generalized	Environment- independent	Cost
LBP							
Eye blink detection							
CNN							
Active flash							
Challenge- response							
3D camera							

mobidev

Adapted from <https://mobidev.biz/blog/face-anti-spoofing-prevent-fake-biometric-detection>.

Maksymenko (2019) compared the state-of-art techniques that are used for face recognition and made this figure that we inserted above. There are reasons that led us to choose LBP for this project. The first reason is that local binary pattern has a better determination on different textures and than eye blink detection. And, LBP can do 2D Dynamic classification where eye blink detection cannot. Another reason for choosing LBP is that although LBP requires average knowledge in the subject area, CNN (Convolutional Neural Nets) requires more expert knowledge about neural nets. Besides it requires significant time to develop and there is no clear and straightforward way to explain how the specific model works. Although the implementation of blink detection is easy and quick, another disadvantage of blink detection is that it can be tricked, if holes are cut for the eyes in the picture and blink frequency might change with health conditions. Although active flash technology separates and classifies face features well, the main disadvantage of it is a brightness that screen brightness matters for active flash. Bright light and daylight might level out the flash. Challenge - Response method can't be used on photos so we eliminated it easily. When we came to the 3D cameras, it looked more accurate than all other methods but the main disadvantage is the cost of the camera and the system.

In this project, we aimed to build a cheap and easily implemented solution for face recognition problems of the users. It might be easily implemented but work well with different patterns like printed versions of the person's photo. So, as a last decision we worked with LBP in this project. LBP also has some drawbacks like low robustness in some cases, not doing well in screen attacks and it is sensitive to noise. But, when we compared the pros and cons that fit our aim, we found LBP is the correct solution for us in this project.

4.2. Results of Lazy Prediction

Table 1

Lazy Prediction Results Table

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	Time Taken
RidgeClassifierCV	1.00	1.00	1.00	1.00	0.02
RidgeClassifier	1.00	1.00	1.00	1.00	0.01
LinearDiscriminantAnalysis	1.00	1.00	1.00	1.00	0.03
Perceptron	0.99	0.99	0.99	0.99	0.01
LabelPropagation	0.96	0.97	0.97	0.96	0.08
LabelSpreading	0.96	0.97	0.97	0.96	0.11
KNeighborsClassifier	0.96	0.97	0.97	0.96	0.03
AdaBoostClassifier	0.95	0.96	0.96	0.95	0.31
LogisticRegression	0.94	0.95	0.95	0.94	0.03
PassiveAggressiveClassifier	0.95	0.95	0.95	0.95	0.02
LinearSVC	0.95	0.95	0.95	0.95	0.02
CalibratedClassifierCV	0.95	0.95	0.95	0.95	0.10
DecisionTreeClassifier	0.91	0.92	0.92	0.91	0.02
XGBClassifier	0.91	0.92	0.92	0.91	0.74
SGDClassifier	0.90	0.91	0.91	0.90	0.02
ExtraTreeClassifier	0.89	0.90	0.90	0.89	0.02
BernoulliNB	0.90	0.90	0.90	0.90	0.02
GaussianNB	0.88	0.89	0.89	0.88	0.01
BaggingClassifier	0.87	0.88	0.88	0.87	0.07
RandomForestClassifier	0.87	0.88	0.88	0.87	0.24
SVC	0.87	0.88	0.88	0.87	0.02
NearestCentroid	0.89	0.88	0.88	0.88	0.01
ExtraTreesClassifier	0.84	0.86	0.86	0.84	0.17
LGBMClassifier	0.83	0.85	0.85	0.83	0.20
QuadraticDiscriminantAnalysis	0.61	0.65	0.65	0.56	0.02
DummyClassifier	0.44	0.50	0.50	0.27	0.01

According to lazy prediction results, Ridge Classifier and LDA are the best performers. To help to detect overfitting, we used K-fold cross validation. In this method, the data is splitted into k equal subsets that are called folds. In each turn, one subset is used as a test set where the remaining folds are used for training.

4.3. t-distributed stochastic neighbor embedding (t-SNE)

When we tried the K-Nearest Neighbors classifier, we saw that optimal neighbor hyperparameter is 1. So, this makes us think about t-SNE to reduce the dimension and plot our observations. This algorithm shows the probability of neighbors around each point. Neighbors mean the set of points that are closest to each other. This makes us observe the distinction between the clusters that contain photos and prints of these photos.

Figure 5

T-SNE projection of photos and prints of these photos in training set

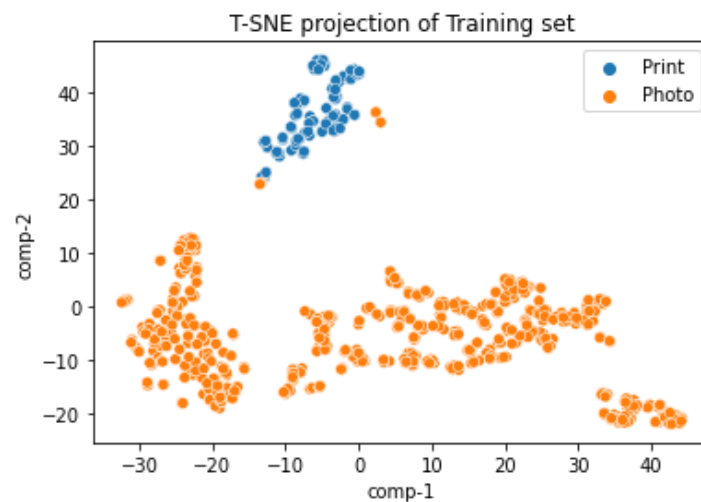
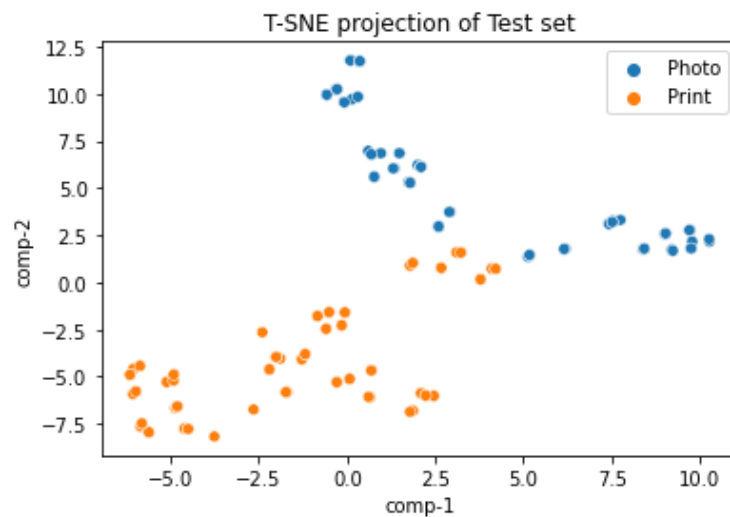


Figure 6

T-SNE projection of photos and prints of these photos in test set



“In t-SNE, the main parameter controlling the fitting is called *perplexity*. Perplexity is roughly equivalent to the number of nearest neighbors considered when matching the original and fitted distributions for each point. A low perplexity means we care about local scale and focus on the closest other points. High perplexity takes more of a "big picture" approach.” (Hoare,2021)

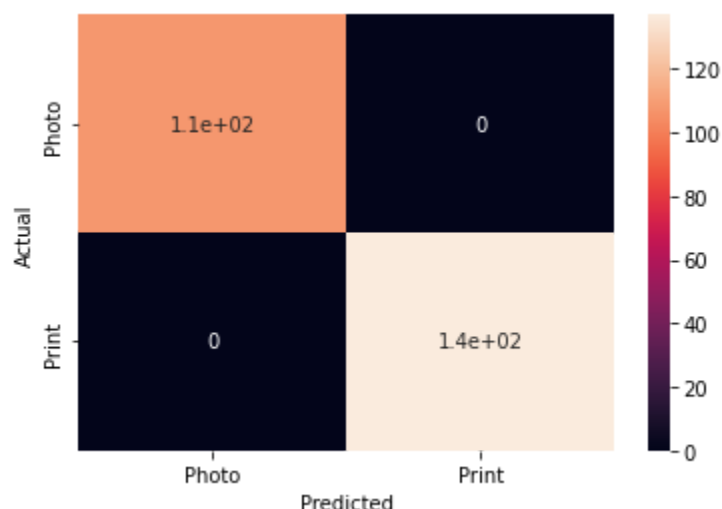
From these observations, we can see the clear clustering as expected, especially in the training set, this is with maximum perplexity and we got similar results from lower perplexity values. In this chart, blue points represent the print photos and orange points represent the original photos and except a few points the distinction between the oranges and blues are very clear in training data. X and y axis show the dimensions 1 and 2, respectively. In test data, although it is not as clear as the training data, there are still distinct points that blues are in the upper portion and oranges are in the lower portion of the invisible separation line.

4.4. LDA Model

Based on the Lazy Predictor values, it seems that the models like Ridge Classifiers and LDA might be better because they won't need much tuning. They already gave good accuracy in the lazy prediction calculations.

The accuracy of the Ridge prediction on the test set is 92.65%. It means that the model predicts the photo or print correctly 92.65% of the time. And, the accuracy of LDA on the train set is 99.1% where the accuracy of the LDA on the test set is 100% after K-fold. This is interesting to see that running the K-fold cross validation we get a less accurate model on the training data then the test data. This might be that the test data is considerably smaller than the train data and it might be solved by looking at the training and testing splits again. Randomization of splitting might be the problem and we can address this issue by changing the splitting ratio and randomization between test and train data.

Figure 7
Confusion matrix of LDA



Confusion matrix shows the actual and predicted results of the model. False positives and false negatives can be seen from the confusion matrix easily. In our results, it is seen that LDA on the test set predicts all photos as photo and predicts all prints as print in the test.

Normally, when we get a 100% accuracy on a train set, it might be considered to be overfitting, but in this project it can be 100% accuracy on a test set, because we checked the projection of our data and there was already a distinct separation between feature vectors of photos and prints. So, the prints and photos can be predicted perfectly. Also, LDA projections maximize the distances between different classes, it does this by maximizing the component class axes for class separation. In other words, it gets a more separate distribution between the classes on that axis. So, it performed well in this project.

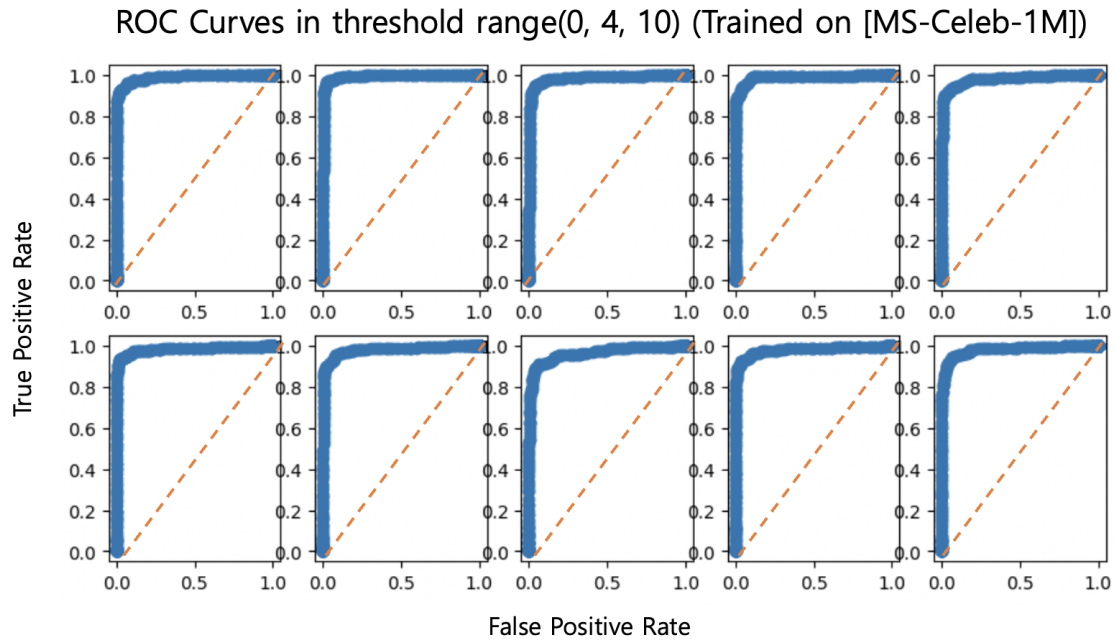
4.5. ArcFace Model

Our model's accuracy on the Chinese people test set is 77%. In the training data set mostly American faces were used and in the test set, we used the Asian faces to see how the model is doing on the new data. 77% accuracy might not be bad in general face recognition cases but if the system will be used in security systems, we want to get more accurate results. So, the fine-tuning can be done, especially for Asian faces, as a future improvement to increase the accuracy. By this way, it makes the system more reliable on faces from different nationalities.

4.5.1. ROC Curve

The Roc curve will help us to measure how successful the model is because it enables us to analyze the ratio of true positives and false positives. As variables of the Roc curve line graph, the x-axis shows the false positive rate (FPR) and y-axis represents the true positive rate (TPR). The graph shows the change in FPR and TPR when measured by continuously changing the criterion. And, through this, we can view if the accuracy of the model increases in a balanced way. If one axis drops sharply as you increase the value on one axis, the area under the curve decreases. This is a sign that we should reconsider the learning method. To do a fine tuning on a pre-trained model, we need to see if TPR or FPR is decreasing in our dataset. We experimented with various thresholds, and as a result, the best result was obtained at 0.4, and 0.4 was selected as the best threshold to proceed with the next face recognition process for Chinese people data.








Figure 8
Roc Curve



4.5.2. Face by Face Distance Matrix

This matrix shows the similarity distance between two faces. By looking at this matrix, the threshold can be defined and users can be given access based on this threshold.

Figure 9
Face Similarities Matrix

							
test_0.png	0.000000	0.942101	2.349689	1.964111	2.434061	1.742653	1.959101
test_1.png	0.942101	0.000000	2.081174	1.760810	2.193204	2.051017	2.109672
test_420.png	2.349689	2.081174	0.000000	1.036252	2.101191	2.178593	2.250163
test_421.png	1.964111	1.760810	1.036252	0.000000	2.091505	2.250613	2.375853
test_00001.jpeg	2.434061	2.193204	2.101191	2.091505	0.000000	1.423741	1.237057
test_00000.jpg	1.742653	2.051017	2.178593	2.250613	1.423741	0.000000	0.704062
test_00002.jpg	1.959101	2.109672	2.250163	2.375853	1.237057	0.704062	0.000000

In this photo matrix, the green box shows the correct person that matches. 0 (zero) values show the same photo, and as expected in the matrix it is diagonal. The important part is that lower than a certain threshold means it is the same person. So, when the value is decreasing, it means that the similarity between two faces are increasing. For example, the first two photos belong to the same person. The 3rd and 4th photos are of the same person. The 5th and 7th photos are of the same person. Model catches the similarities below a certain threshold. In this case, it might be 1.3. There is one exception here that models find 6th and 7th photos similar also, although they are not the same person. Other than that, when we look at the results row by row, we can see that the lower results point to the same person and model did a good job on this.

4.6. Ethical and Security Considerations

4.6.1. General Security Cautions of Biometric Identification

Although face recognition is a useful and important technology from the security standpoint, it should be used in a very careful way. The leakage in the biometric identifiers such as fingerprint, iris, voice, face, etc. can threaten the privacy of people and can violate personal rights. The misuse of these systems and this data can cause significant damages so the usage of facial recognition systems should be very sensitive and accurate. It is very important to have a strong anti-spoofing check in the face recognition systems.

There are many high-security facilities that use biometric identification. If the system can be hacked easily, it can give dangerous access to the wrong person. Or, in the border checkpoints, if the passport photo and the holder's face can not be compared accurately, it can lead to wrong access to criminals that threatens the security of the countries. As a result, biometric identification is a very useful technology for security, but it also has to be used very carefully to prevent crimes, data vulnerabilities, violation of personal rights and privacy.

To address this issue, we highlighted the anti-spoofing part of this project.

4.6.2. Ethical Considerations

The main concern of the facial dataset is confidentiality. Because, the facial, retinal, or fingerprint data is very sensitive information for privacy. To address this concern, we used a public pre-trained model and publicly open dataset. Another important concern is that if there will be a web app in future for this facial recognition system, the personal data shouldn't be saved in the server, it can be saved in the session and it has to be deleted after the session will be ended.

Another ethical consideration of facial recognition is transparency and consent. To address this issue, we used a publicly open dataset, but in daily applications, these systems are sometimes used without consent and notification. For example, if the cameras will be used by the general public for recognition purposes, it is not good to use this data without informing the people.

Racial bias is another ethical issue that should be considered in face recognition systems. Because, it can cause to racial discrimination when it is used by law enforcement if the models don't work well on diverse faces. So, the face recognition systems should be used very carefully, sensitively and privately.

5. Conclusion & Future Improvements

In conclusion, although there are many different approaches in face recognition systems, in this project we chose the LBP and LDA model for the anti-spoofing part and for the face matching part, we used the ArcFace model that is one of the state-of-art models.

For the future improvements, to improve the reliability of the model, video detection can be used. Another option that might work very well is the combination of anti-spoofing solutions such that eye blink detection can be added to LBP, and the combination of these two methods will be used. Active flash also might be added to this combination to make it more strong.

Although we used as uniform photos as possible, the unnecessary artifacts in the images can also cause overfitting and reduce the reliability of the system. To improve the reliability of the system, we can improve the uniformity of the data sets. We can work on the balance of the detailed artifacts in train and test sets. Besides, we can work to improve the randomization of the splitting to address that the accuracy of the test set is higher than the accuracy of the train set. Another future work might be the demo web app to represent the project in application and to show the connection between anti-spoofing and ArcFace modeling output. For the ArcFace model, fine-tuning on more data can be done for future work, specifically on Asian faces. Because, pre-trained ArcFace is trained on celebrities data and the majority of the data contains American celebrity faces. So, if the model will be fine-tuned on more Asian faces, the accuracy of the results might increase.

Also future work should include use of inkjet printers as inkjets can have dpi of up to 5000 dpi as of date, this will better test the performance of the model. More greyscale data augmentation techniques like 45 degree rotations or random degree rotations, as well as shifting augmentation should be applied to increase the dataset.

6. Statement of Work

Ali Arefi	Background research, Preprocessing, Anti-spoofing techniques, Visuals.
Ezgi Gumusbas	Background research, Weekly deliverables, Final Report, Poster, Github.
Yujin Lee	Background research, Data search, ArcFace model, Visuals.

7. References

Biometrics. Biometrics | Homeland Security. (n.d.). Retrieved December 8, 2022, from <https://www.dhs.gov/biometrics>

Dayala, R. (2020, July 23). *10.7 local binary patterns*. Computer Vision. Retrieved December 11, 2022, from <https://cvexplained.wordpress.com/2020/07/22/10-7-local-binary-patterns/#:~:text=LBP%20compute%20a%20local%20representation,r%20surrounding%20the%20center%20pixel.>

Deng, J., Guo, J., Yang, J., Xue, N., Kotsia, I., & Zafeiriou, S. (2022, September 4). *Arcface: Additive angular margin loss for deep face recognition*. arXiv.org. Retrieved December 11, 2022, from <https://arxiv.org/abs/1801.07698>

Facial spoofing: Meaning and how to prevent it. (2022, May 6). Retrieved December 9, 2022, from <https://www.electronicid.eu/en/blog/post/facial-spoofing-what-it-is-how-to-prevent-it-and-spoofing-detection-solutions/en>

Fratric, I., & Ribaric, S. (n.d.). *Local binary LDA for face recognition*. Retrieved December 11, 2022, from <https://bib.irb.hr/datoteka/511724.lbllda.pdf>

Hernandez-Ortega, J., Fierrez, J., Morales, A., & Galbally, J. (2019). Introduction to Face Presentation Attack Detection. In *Handbook of Biometric Anti-Spoofing* (Second, pp. 190–194). essay, Springer.

Hoare, J. (2021, June 9). *How t-SNE works and Dimensionality Reduction*. Displayr. Retrieved December 11, 2022, from <https://www.displayr.com/using-t-sne-to-visualize-data-before-prediction/>

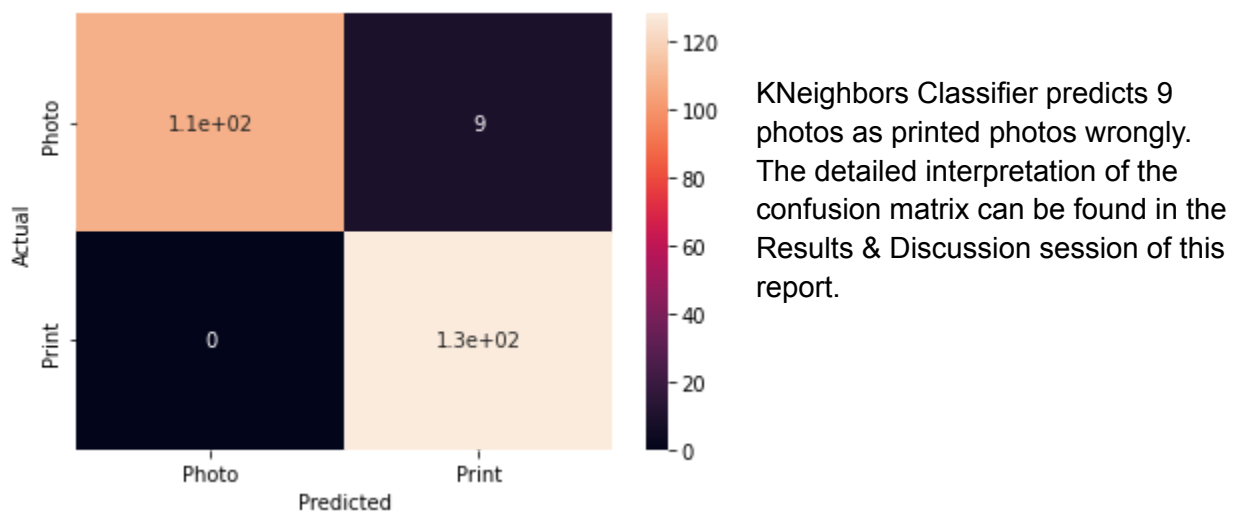
Maksymenko, S. (2019, August 23). *Face anti-spoofing techniques for liveness detection in security systems*. MobiDev. Retrieved December 11, 2022, from <https://mobidev.biz/blog/face-anti-spoofing-prevent-fake-biometric-detection>

Shi, Y. (2021). *TOWARDS A ROBUST UNCONSTRAINED FACE RECOGNITION PIPELINE WITH DEEP NEURAL NETWORKS*. Michigan State University. Retrieved December 11, 2022, from <https://biometrics.cse.msu.edu/>

8. Appendix

Appendix I. Confusion Matrix of KNeighborsClassifier

Accuracy: 96.33%



Appendix II. Confusion Matrix of Ridge ClassifierCV

Accuracy: 92.65%

