KLE Society's
KLE Technological University

A Mini Project Report
On

Fingerprint Liveness Detection

submitted in partial fulfillment of the requirement for the degree of

Bachelor of Engineering
In
Computer Science and Engineering

Submitted By

Priyanka C Pardeshi	01FE19BCS014	111
Manisha Belagal	01FE19BCS207	407
Mehar Anjum	01FE19BCS208	408
Devyansh Agrawal	01FE19BCS245	443

Under the guidance of
Ms. Neha Tarannum

.

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

HUBLI–580 031 (India)

Academic year 2021-22

KLE Society's
KLE Technological University

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that Mini Project entitled **"Fingerprint Liveness Detection"** is a bonafied work carried out by the student team Ms. Priyanka C Pardeshi - 01FE19BCS014, Ms.Manisha Belgal – 01FE19BCS207, Ms. Mehar Anjum – 01FE19BCS208, Mr. Devyansh Agrawal – 01FE19BCS245, in partial fulfilment of completion of Fifth semester B. E. in Computer Science and Engineering during the year 2021 – 2022. The project report has been approved as it satisfies the academic requirement with respect to the project work prescribed for the above said programme.

Guide **Head, SoCSE**

Ms. Neha Tarannum **Dr. Meena S. M**

External Viva:

Name of the Examiners	**Signature with date**
1.	
2.	

ABSTRACT

Fingerprints are unique way of identifying an individual since no two individuals have the same fingerprint. Under any conditions like burns or tearing of the skin the fingerprint doesn't lose its pattern. Now a days fingerprints are widely used in authentication systems. Fingerprint scanners can be spoofed by artificial fingers using mouldable plastic, clay, play-doh , gelatine, silicone, rubber and other materials. Thus, developing an efficient method to protect fingerprint systems from imposter access is urgent. Sometimes the fingerprint scanner might have some dust particles on it or the sensor might be moist due to humidity, therefore the images generated might have some distortion and noise, this noise has to be removed before classification, therefore we are trying to remove the noise and distortion using GAN and identify the liveness of the fingerprint by using various models like CNN, VGG 16.

Keywords: Fingerprint, GAN, CNN, VGG 16, Authentication, Spoofed, Liveness, Distortion

ACKNOWLEDGEMENT

We would to express our gratitude to a number of individuals whose professional guidance and encouragement helped us in the successful completion of this project.

We take this opportunity to thank Dr. Ashok Shettar, Vice-chancellor, KLE Technological University and to Dr. Prakash Tewari, Principal, KLE Technological University, Hubballi.

We also take this opportunity to thank Dr. Meena S M, Head and Coordinator, School of Computer Science and Engineering for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

We sincerely thank our guide Ms. Neha Tarannum , School of Computer Science and Engineering for her guidance, inspiration and whole hearted co-operation during the period of completion.

Our gratitude will not be complete without thanking the Almighty God, our beloved parents, our seniors and our friends who have been a constant source of blessings and aspirations.

Priyanka C Pardeshi	01FE19BCS014	111
Manisha Belagal	01FE19BCS207	407
Mehar Anjum	01FE19BCS208	408
Devyansh Agrawal	01FE19BCS245	443

Chapter No.	TABLE OF CONTENTS				Page No.
1.	**INTRODUCTION**				**3**
	1.1	Preamble			3
	1.2	Motivation			3
	1.3	Objectives of the project			3
	1.4	Literature Survey			4
	1.5	Problem Statement			6
2.	**PROPOSED SYSTEM**				**7**
	2.1	Description of Proposed System.			8
	2.2	Description of Target Users			8
	2.3	Advantages/Applications of Proposed System			8
	2.4	Scope			8
3.	**SOFTWARE REQUIREMENT SPECIFICATION**				8
	3.1	Overview of SRS			8
	3.2	Requirement Specifications			9
		3.2.1	Functional Requirements		9
		3.2.2	Use case diagrams		10
		3.2.3	Use Case descriptions using scenarios		11
		3.2.4	Non functional Requirements		13
			3.2.4.1	Performance requirements	13
			3.2.4.2	Usability	13
			3.2.4.3	Software and hardware requirement	13
	3.3	Software and Hardware requirement specifications			13
4	**SYSTEM DESIGN**				14
	4.1	Architecture of the system			15
	4.2	Activity Diagram			16
5	**IMPLEMENTATION**				17
	5.1	Proposed Methodology			17
	5.2	Description of Modules			17
6	**TESTING**				18
	6.1	Test Plan and Test Cases			18

1.INTRODUCTION

1.1 PREAMBLE

Fingerprints have played a major role in identification of an individual. In many real time scenarios like employment, defence security clearance, fraud detection, and so on, fingerprints play an effective role in identifying the person and validating.
Since the fingerprint biometrics is widely used these days, there are many threats to this fingerprint system. One of major threats is the usage of spoofed fingerprints. Artificial fingerprint can made of rubber, plastic, wax, and gelatin which resembles the original fingerprint.
In current scenario our proposed system can be used in various sectors like forensics, military, banks and other departments where fingerprint is used for authentication. Our system will first denoise the fingerprint image and then identify whether the fingerprint is fake or live and if it is found to be fake the material used for spoofing is also identified which in turn increases the authenticity.

1.2 MOTIVATION

The use of biometric systems is widespread today, it is rapidly replacing the old system of using a PIN, password or patterns that must always be remembered. Biometric characteristics allow people to successfully identify and authenticate the various security systems. Biometric fingerprint detection is one of the easiest ways to improve system security. Fingerprints are unique for every individual and even if a person has burnt his fingers the pattern of fingerprint is not lost. But there are some disadvantages too, there are many threats to this system. One of the threats is fake fingerprints. Artificial fingerprint can be made of rubber, plastic, wax, wood glue and gelatin which resembles the original fingerprint. So to prevent spoofing and illegal access to various confidential and important security systems such as banking applications we go for anti-spoofing techniques like Liveness Detection.

1.3 OBJECTIVES OF THE PROJECT

- To Deblur the given fingerprint sample if required.
- To extract the features of the fingerprint.
- To detect if the given fingerprint sample is Fake or Real.
- To classify the spoofing material when the detected sample is fake.
- To design graphical user interface for the model.

1.4 LITERATURE SURVEY

1. **Paper Title:** Detection of Fingerprint Authenticity Based on Deep Learning Using Image Pixel Value .
   Authors: Hariyanto, Sunny Arief Sudiro, Tubagus Maulana Kusuma, Sarifuddin Madenda, Laode M. Rasdi Rere
   Date Published: 24 December 2020
   Learning: This research proposes a method based on deep learning with the CNN algorithm for the implementation of the authenticity of fingerprint detection. Experiments were performed on a fingerprint dataset with a variety of different parameters to see the best accuracy results.

2. **Paper Title:** Convolutional Neural Networks for Fingerprint Liveness Detection System
   Authors: Arun Kumar T K , Vinayakumar R , Sajith Variyar V V , Sowmya V , Soman K P
   Date Published: 16 April 2020
   Learning: This paper uses CNN for feature extraction and classification of fingerprints into live or fake. A comparison study was made where the features extracted from CNN are applied to various machine learning algorithms such as SVM, KNN etc .From the comparison study of deep learning based and machine learning based classifiers, it was observed that both models gave almost similar performances with a maximum test accuracy of 95%.

3. **Paper Title**: Overview of Fingerprint Recognition System
   Authors: Mouad M.H. Ali, Pravin Yannawar, Vivek Hilal Mahale,Ashok Gaikwad
   Date Published: March 2016
   Learning: This paper gives us an overview on the main steps involved in fingerprint authentication that are: Image Capture or Image Acquisition stage, Image Pre-processing stage, Feature extraction stage, Matching stage. It also talks about the previous processes and algorithms used for each step.

4. **Paper Title**: Fingerprint Liveness Detection by a Testing and Trained Data using Convolutional Neural Network
   Authors: Vishwas Prakash. R, Dr. Mohan Kumar H. P
   Date published: 2020
   Learning: This paper states about the precision of the liveness model by admitting the dataset (i.e. fingerprint) to be live or fake. The proposed method uses the fingerprints referring to the trained and test fingerprints using convolution neural networks to render a judgment on liveliness. The liveness is checked by feature extraction, testing and training phases. Using CNN and various algorithms for classification like decision tree, Randomforest, Logistic Regression, Neural Net, Gradient Boosting and Naive Bayes the model has been implemented.

5. **Paper Title:** Evaluating software-based fingerprint liveness detection using Convolutional Networks and Local Binary Patterns.
   Authors: Rodrigo Frassetto Nogueira, Roberto de Alencar
   Date Published: 10 October 2015
   Learning: This research proposes a method based on deep learning with the CNN algorithm and Local Binary Pattern. Both techniques were used in conjunction with a Support Vector Machine (SVM) classifier. Dataset Augmentation was used to increase classifier's performance and a variety of preprocessing operations were tested for the implementation of the authenticity of fingerprint detection.

6. **Paper Title:** FDeblur-GAN: Fingerprint Deblurring using Generative Adversarial Network.
   Authors: Amol S. Joshi,Ali Dabouei,Jeremy Dawso, Nasser M. Nasrabadi
   Date Published: 21 JUNE 2021
   Learning: This research proposes a method to De-Blur the fingerprint samples acquired from crime scenes, mobile cameras, or low-quality sensors. This paper uses FDeblur-GAN, based on the cGANs and multi-stage framework of the stack GAN used. The first sub-network is a ridge extractor model. It is added to generate ridge feature maps to ensure that fingerprint information are preserved in the deblurring process. The second sub-network is a verifier that helps the generator to preserve the fingerprint information during the generation process. Using a database of blurred fingerprints and corresponding ridge maps, the deep network learns to deblur from the input samples. The discriminators, ridge extractor, and verifier, work simultaneously to enhance the deblurring process.

7. **Paper Title:** Transformers and Generative Adversarial Networks for Liveness Detection in Multitarget Fingerprint Sensors
   Authors: Soha B. Sandouka, Yakoub Bazi and Naif Alajlan
   Date Published: 20 January 2021
   Learning: The aim of this paper is to increase the accuracy for cross sensor and cross materials. CycleGAN is used to generate augmented versions of fingerprint.Then a pre trained CNN is used together with a transformer to form a hybrid network. It contains three main components, which are: a CNN backbone to extract a compact feature representation of a fingerprint, a transformer, and cycleGAN.

1.5 PROBLEM STATEMENT

To develop a system that classifies the given sample of fingerprint into real or fake and also categorize the spoofing material used for fake fingerprint, using deep learning techniques.

PROBLEM DESCRIPTION: Given a fingerprint sample the task is to first deblur the fingerprint image using DeBlur-GAN and then to classify the sample fingerprint into live or fake using CNN or VGG16 as the classifiers. If the sample is found to be fake, the spoofing material is identified.

2. PROPOSED SYSTEM

2.1 Description of proposed system with simple block diagram

Fig 2.1 represents the block diagram which describes the flow/process of the entire system. FDeblur-GAN framework consists of a cGAN deblurring network and the output of the FDeblur-GAN is given as the input to the deep-learning model.These images are fed into deep learning models like CNN or VGG-16. After processing the image, it is then fed into a classifier which classifies the input image as fake or real. If the image is fake, then it is further classified based on the type of material the fake fingerprint is made from.

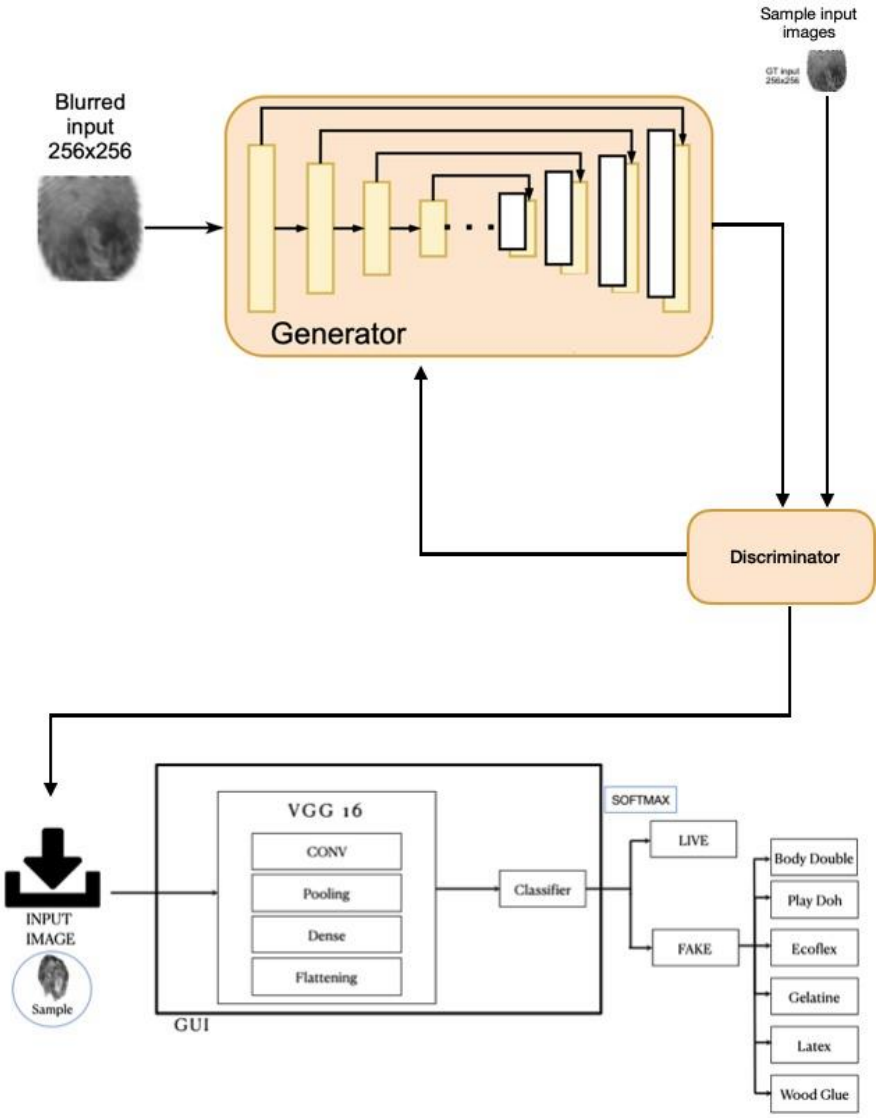

Fig 2.1 Block Diagram of the proposed system

2.2 Description of Target users

Our target users are Forensic personnel's, researchers, organizations using fingerprint authentication.

2.3 Advantages/Applications of Proposed System

- It will help to decrease the crime rate.
- It is used to de-Blur the fingerprints .
- It will help to identify the spoofing material.
- It is used to increase the accuracy of liveness detection.
- It will avoid illegal or unauthorised access to confidential information.
- It can be used in banking applications and to secure confidential records.

2.4 Scope (Boundary of Proposed System)

The scope of this project is restricted to detect if the fingerprint sample is real or fake and not in identification of individual.

3. SOFTWARE REQUIREMENT SPECIFICATION

3.1 Overview of SRS

SRS is the basis for the entire project. It lays the framework that should be followed. Software requirements specification assesses the requirements before entering into the further stages, so that redesigning reduces to a large extent. This helps in reducing the time, efforts and also the cost. SRS establishes an agreement between clients and programmers on how the software product should function. SRS document contains the necessary requirements.

It helps in estimating product costs, risks. Project failures can be prevented if used properly. The developer must be in constant communication with the client to be able to understand the requirements of the clients clearly so that he/she can incorporate those changes in the software.

3.2 Requirement Specifications

3.2.1 Functional Requirements

- User:

1. The user shall be able to successfully login into the system.

- System:

1.1 The system shall be able to authenticate the user.

1.2 The system shall display a warning for invalid credentials.

- User:

2. User shall be able to upload the input image.

- System:

2.1 System shall provide the option for uploading the input image.

2.2 System shall be able to extract the features of the input image.

2.3. System shall be able to detect the image as real or fake.

- User:

3. User shall be able to view the result.

- System:

3.1 System shall be able to display if the input image is live or fake

3.2 If the image is found to be fake the system shall classify the image as Body Double , Ecoflex , Gelatin , Oomoo , Playdoh and display the same.

3.2.2 Use Case Diagram:

Fig 3.1 describes the functionality and requirements of the system.

The use cases in the figure are : Login into the system, Upload the input image, View Live/fake and Develop a GUI.

Login into the system includes: entering the user ID and the password.

View Live/Fake use case has an extend relationship with the view spoofing material for fake image use case as shown in the figure 3.1

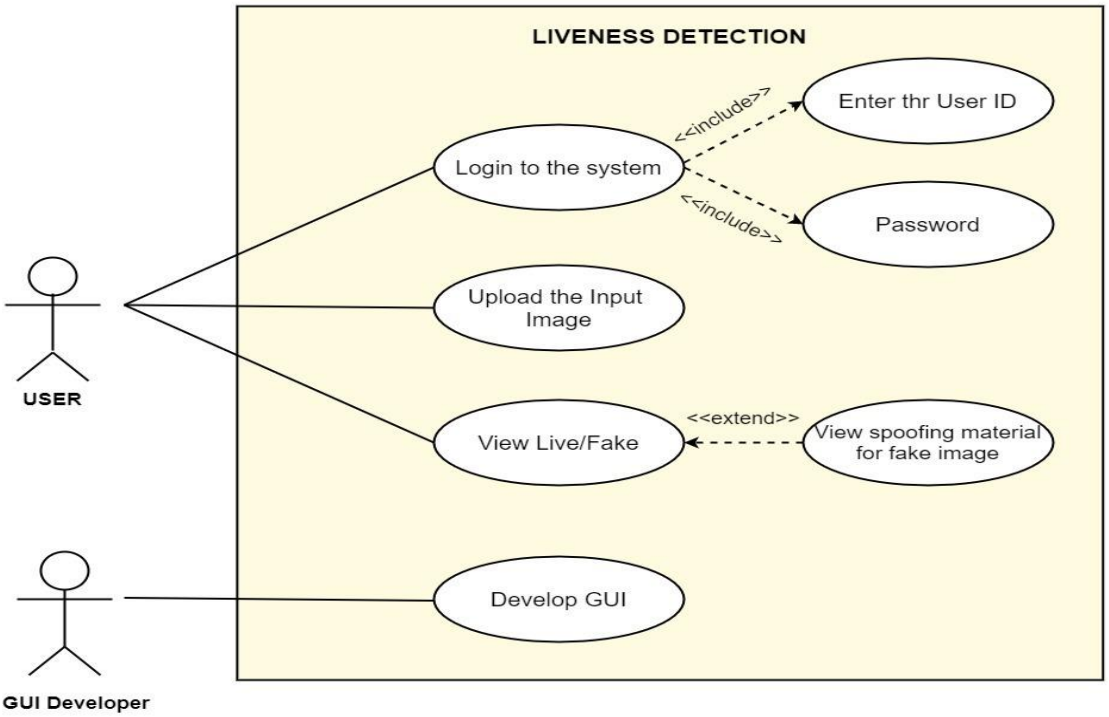

Fig 3.1 – Use case diagram to capture the functionality and requirements of the system

3.2.3 Use Case Descriptions Using Scenarios

Use case: Login to the system.

Actor: User.

Pre-Condition: The use must be registered to the system.

Post - Condition: The user-Id & the password entered by the user should be valid.

Main Success Scenario:

1. User enters the User-id.

2. User enters the password.

3. User will be able to login successfully if the entered Id & password are valid.

4. Home page will be displayed after logging in.

Exception Scenario:.

1. The entered User-Id & password is invalid.
2. Alert message will be displayed.

2. Use Case: Upload Input Image.

Actor: User.

Pre-Condition: The input image should be in .jpeg or .png format.

Post- Condition: The input image should get uploaded successfully!

Main Success scenario.

1. The user must click on the upload button to upload the input image.

2. The user shall wait for few seconds for the completion of uploading the image.

Exception scenario:

If the user uploads image of format, other than .jpeg and .png then the image won't be supported by the system.

3. View live or fake.

Actor: User

Pre-Condition: The input image should be processed Successfully.

Post-Condition: The system should be able to fetch the result from the model.

Main Success scenario:

1. After the uploading of image, the system will display the analysed result.

FAKE: If the input image processed was fake, it displays the result as fake.

LIVE (Real): System displays real for the real image.

Exception scenario:

1: If the image was not processed, then an alert message will be displayed as process failed.

4. Use Case: Develop GUI

Actor: GUI developer.

Pre-Condition: The developer should have the fresher knowledge of the GUI development.

Post-Condition: The developer is able to develop GUI successfully.

Main Success scenario:

1. Login Page will be developed with all the necessary functionality.

2. The Login page consists of Userld, password and a login button.

3. Once the user is logged in successfully, the login page is redirected to home page.

4. The Home page consists of the following features:

(i) Uploading the image

(ii) Check button.

(iii) Result button.

5. The model will be able to process the input image and display the desired result.

Exception scenario:

1. The entered user-id & password by the user is invalid.

2. The format of the input image is not supported by the system.

3.2.4 NonFunctional Requirements

3.2.4.1 Performance requirements

- To attain the accuracy of atleast 90% for determining the authenticity of the fingerprint.
- The desired output should be displayed within 5 seconds.
- The user should be redirected to the homepage after login within 5 seconds.

3.2.4.2 Usability

- The format of the input image should be either .jpeg or .png.

3.3 Software and Hardware requirement specifications

The minimum requirement of this system is 8 GB RAM ,8 core GPU, 50 GB ROM, Linux operating system.

4. SYSTEM DESIGN

4.1 Architecture of the system

Fig 4.1 describes the process of Pipe and Filter architecture of the system. The architecture consists of pipe and filter.Filter is a processing component which is discrete and carries out data transformation.This data moves from one component to another as in a pipe.In the below figure, the input image is pre-processed and feature extraction of the input image is done. It is further fed to the classifier which classifies the fingerprint into fake and real. If the fingerprint is found to be fake then the classification of spoofing material takes place and displays the desired output as shown in the above figure.

Pipe And Filter Architecture

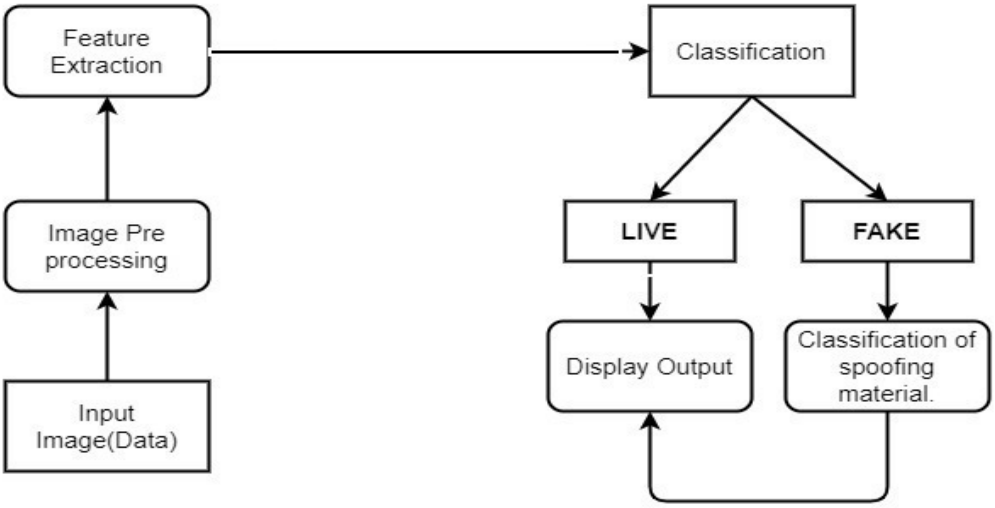

Fig 4.1 – Pipe and Filter Architecture of the system

Model View Controller Architecture For GUI

Fig 4.2 describes the process of MVC architecture. Here there are three components which are model , view and controller . The model component is responsible of the operation and management of the data.The view component is responsible for the presentation of the data to the user . The controller component is responsible for user interaction.

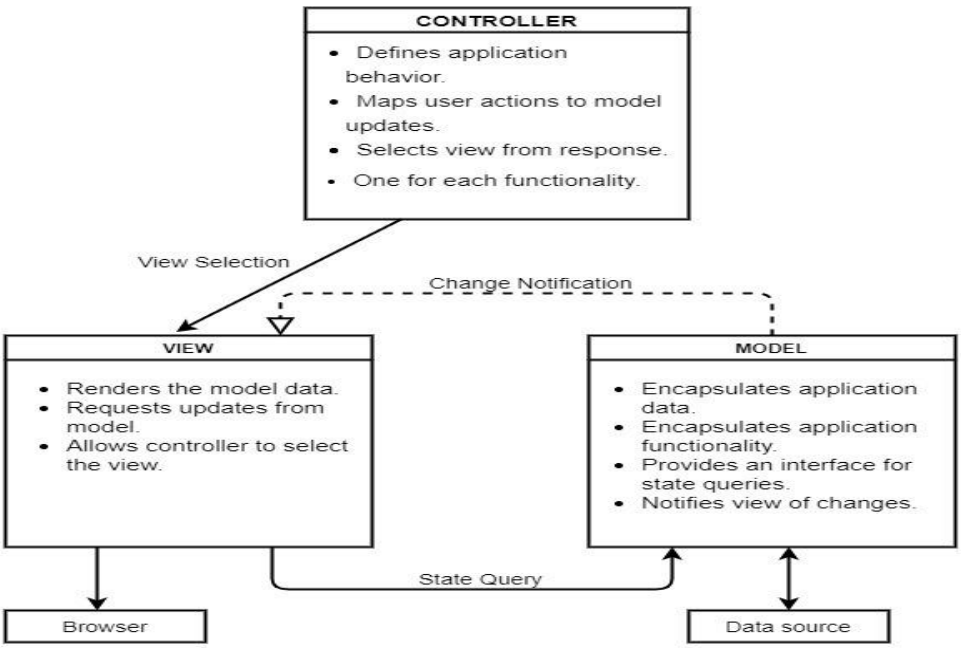

Fig 4.2 – Model View Controller Architecture of the system

4.2 ACTIVITY DIAGRAM

User Login

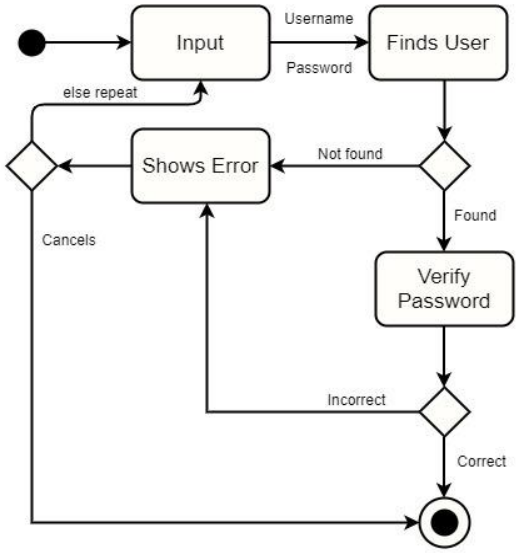

Fig 4.3 – Activity Diagram for Login System

Upload Image

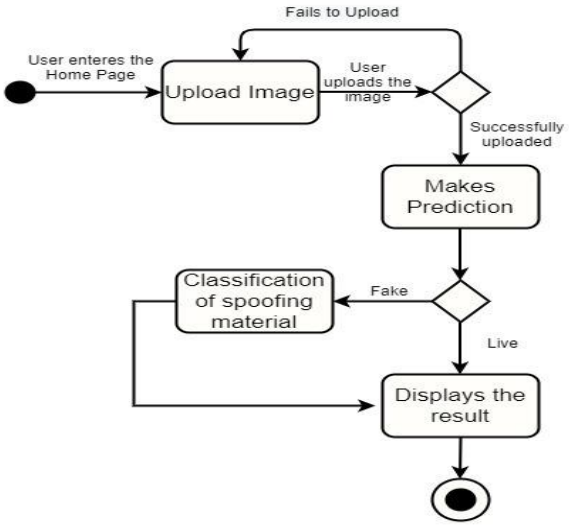

Fig 4.4 – Activity Diagram for Uploading Image

The activity diagram for login system and upload image is shown in Fig 4.3 and Fig 4.4 respectively. The diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

5. IMPLEMENTATION

5.1 PROPOSED METHODOLOGY

5.1.1. Dataset Analysis

- The dataset which we acquired consists of real and fake fingerprint samples with different sensors.
- The fake fingerprint sample consists of different spoofing materials i.e Body Double , Ecoflex , Playdoh ,Gelatine , Latex , Wood glue.
- Total training images – 8983 (4473 fake fingerprints ,4510 live fingerprints).
- Total testing images - 6551 (2051 fake fingerprints ,4500 live fingerprints).

5.1.2 Data Pre-processing

- Re-scaling : Helps to get the pixel value between 0 and 1.
- Shear range : image will be distorted along an axis, mostly to create or rectify the perception angles.
- Zoom range :This method randomly zooms the image either by zooming in or it adds some pixels around the image to enlarge the image.
- Horizontal flipping : Reversing the entire rows and columns of an image pixels horizontally

Model Used : Convolution Neural Network, VGG16 ,DeBlur-GAN

6.TESTING

6.1 TEST PLAN AND TEST CASES

Requirement ID	Test ID	Input	Expected Output	Test Status
1.	1.1	Correct Username and Password	Navigates to next page	Pass
	1.2	Wrong username	Fails to direct to next page	Pass
	1.2	Password not correct	Fails to direct to next page	Pass
2.	2.1	Uploading image format is .jpeg	Processes and displays output	Pass
	2.2	Uploading image format is .png	Processes and displays output	Pass
	2.3	Uploading image format in other formats	Input image is not accepted.	Pass

7.RESULTS AND DISCUSSIONS

Results obtained from CNN and VGG16 model

CNN	VGG 16
• 4 convopool layers are used • Contains 2 modules feature extraction module and classifier module	13 layers of CNN with 5 layers of pooling and 3 layer dense
Transformations 1.shear_range = 0.2 2.zoom_range = 0.2 3.horizontal_flip = True	Transformations 1.shear_range = 0.2 2.zoom_range = 0.2 3.horizontal_flip = True
Accuracy -91.91 %	Accuracy – 94.23%
Loss -0.2880	Loss – 0.2423

Graph of Loss vs Epoch of CNN model

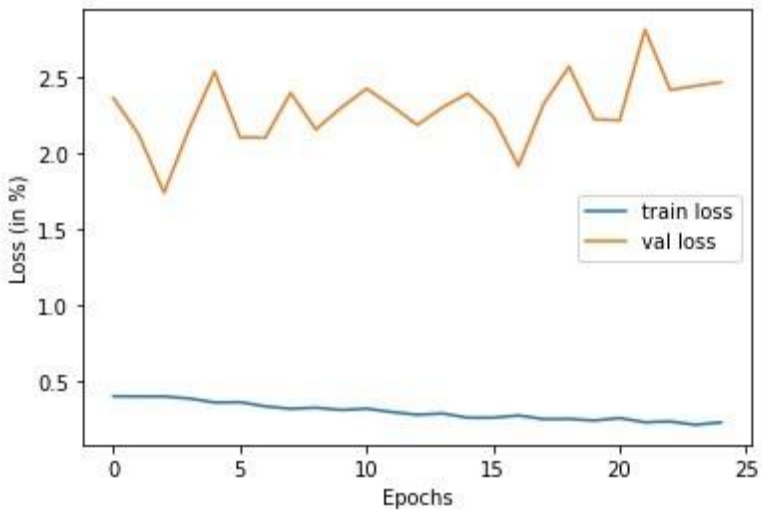

Fig 7.1 – Graph of Loss vs Epoch of CNN model

The graph obtained for loss vs epoch using CNN model is depicted in Fig 7.1

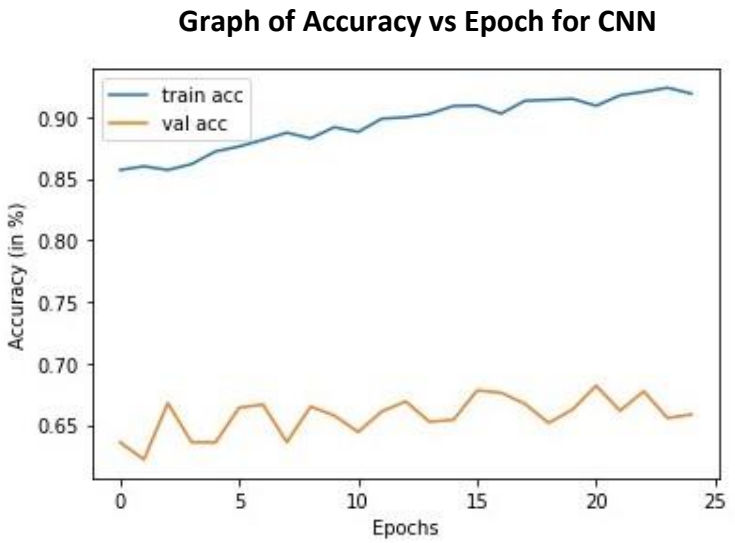

Fig 7.2 – Graph of Accuracy vs Epoch for CNN model

The graph obtained for accuracy vs epoch using CNN model is depicted in Fig 7.2

GUI developed using Django Framework

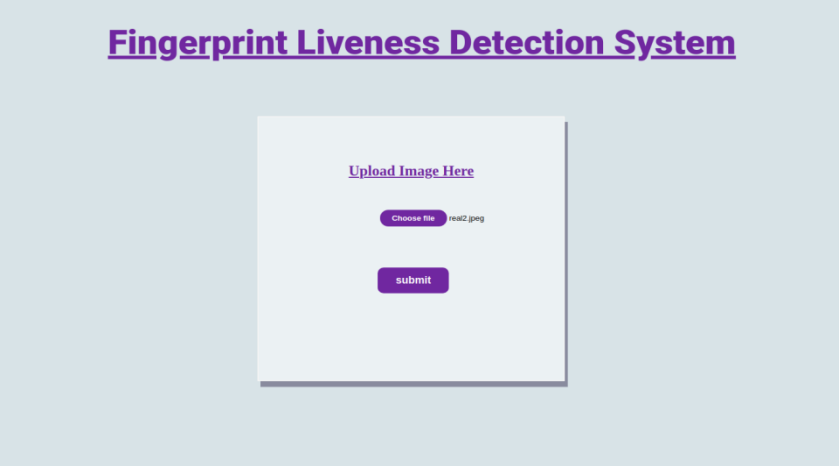

Fig 7.3 – Home Page

Fig 7.3 shows the home page of the GUI developed using Django Framework. Here the user is provided with an option to upload the input image of .jpeg or .png type. Once the user uploads the image, the user should click on the submit button to view the result.

Fig7.4- Output for Live Fingerprint

Fig 7.4 displays the result when the image is uploaded successfully i.e. it predicts whether the uploaded sample/input image was live or fake. In this figure it displays the result as Live. It also displays the image uploaded by the user.

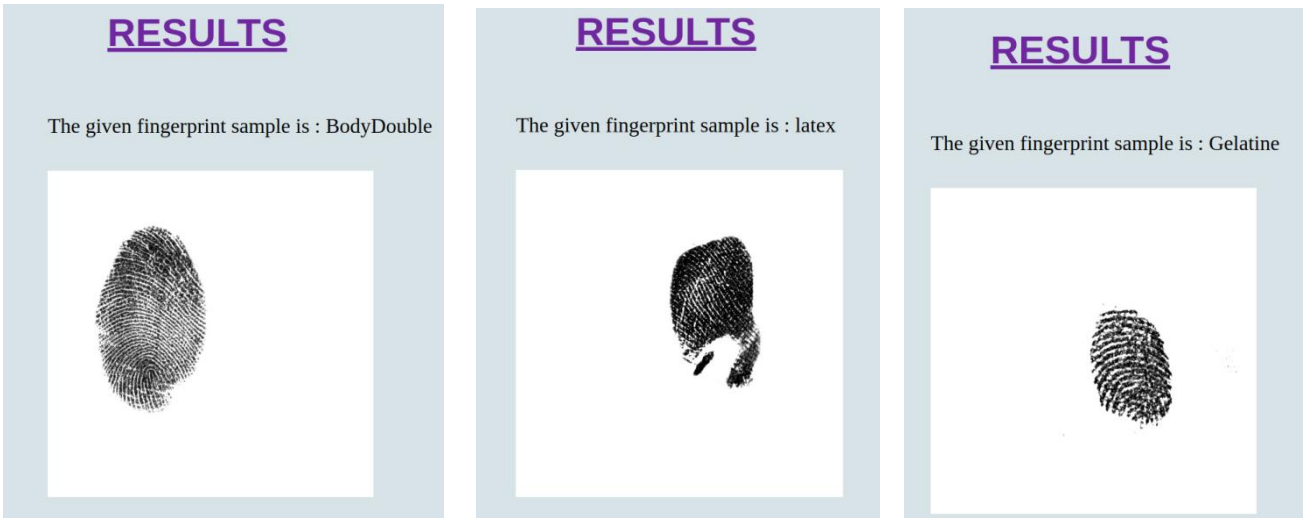

Fig 7.5 - Output for Fake Fingerprint

Fig 7.5 depicts the various results generated for different types of fake fingerprint sample when the fingerprint image uploaded was predicted fake. It displays the spoofing material as shown in the figure. The different types of spoofing materials are shown in the figure i.e. BodyDouble, Latex and Gelatine.

8.CONCLUSION AND FUTURE SCOPE

CONCLUSION

We tried with the implementations of models like CNN, VGG16 and GAN to rectify the distortion (Deblurring) of the fingerprint images. We obtained an Accuracy of 91.91% for CNN and an accuracy of 94.23% for VGG16. In real life scenarios we come across fingerprints which may be blurred so we have to Deblur it and then identify if the fingerprint is live or fake, to conclude we can use GAN to remove blurriness from the image and then classify it as real or fake and if it is found to be fake then the spoofing material will be identified. We also have created an user interface for the same using Django framework. Here the user can upload an input image and the predictions will be shown along with the input image which the user had given as an input. If it's a real fingerprint LIVE is displayed or else if its a fake one then the spoofing material is displayed as the output.

FUTURE SCOPE

Fingerprints are unique way of identifying an individual since no two individuals have the same fingerprint, not even twins. So, it is a reliable technique which is widely used and will be very beneficial to us in the future as well if we overcome the threats that it currently has. Since some of the very confidential information is secured by the fingerprint systems, it is essential that we overcome the threats. It can be used in various applications like banking applications, online payment, mobile banking. It can be used in business applications, for secure data storage, immigration services and many more applications.

9.REFERENCES

1. Free-form image inpainting with gated convolution. Yu et al. In ICCV 2019
2. Region Normalization for Image Inpainting. Tao et al. In AAAI 2020
3. Learning to Incorporate Structure Knowledge for Image Inpainting, Yang et al. In AAAI 2020
4. Prior Guided GAN Based Semantic Inpainting, Lahiri1 et al. In CVPR 2020
5. UCTGAN: Diverse Image Inpainting based on Unsupervised Cross-Space Translation, Zhao et al. In CVPR 2020
6. Recurrent Feature Reasoning for Image Inpainting, Li et al. In CVPR 2020
7. Contextual Residual Aggregation for Ultra High-Resolution Image Inpainting, Yi et al., In CVPR 2020
8. 3D Photography using Context-aware Layered Depth Inpainting, Shih et al. In CVPR 2020
9. Rethinking Image Inpainting via a Mutual Encoder-Decoder with Feature Equalizations, Liu et al. In ECCV 2020
10. Guidance and Evaluation: Semantic-Aware Image Inpainting for Mixed Scenes, Liao et al. In ECCV 2020
11. VCNet: A Robust Approach to Blind Image Inpainting, Wang et al. In ECCV2020
12. High-Resolution Image Inpainting with Iterative Confidence Feedback and Guided Upsampling, Zeng et al. In ECCV 2020
13. Image Inpainting with Onion Convolution, Shant et al., In ACCV2020
14. Hyperrealistic Image Inpainting with Hypergraphs, Wadhwa et al., In WACV 2021
15. Image Inpainting with External-internal Learning and Monochromic Bottleneck, Wang et al., In CVPR 2021
16. Generating Diverse Structure for Image Inpainting With Hierarchical VQ-VAE, Peng et al., In CVPR 2021
17. PD-GAN: Probabilistic Diverse GAN for Image Inpainting, Liu et al., In CVPR 2021
18. Image Inpainting Guided by Coherence Priors of Semantics and Textures, In CVPR 2021
19. TransFill: Reference-guided Image Inpainting by Merging Multiple Color and Spatial Transformations, Zhou et al., In CVPR 2021
20. FaceInpainter: High Fidelity Face Adaptation to Heterogeneous Domains, Li et al., In CVPR 2021
21. Large Scale Image Completion via Co-Modulated Generative Adversarial Networks, Zhao et al., In ICLR 2021
22. Image Inpainting via Conditional Texture and Structure Dual Generation, Guo et al., In ICCV 2021
23. CR-Fill: Generative Image Inpainting With Auxiliary Contextual Reconstruction, Zeng et al., In ICCV 2021
24. Parallel Multi-Resolution Fusion Network for Image Inpainting, Wang et al., In ICCV 2021

25. WaveFill: A Wavelet-Based Generation Network for Image Inpainting, Yu et al., In ICCV 2021
26. SLIDE: Single Image 3D Photography With Soft Layering and Depth-Aware Inpainting, Kaeser et al., In ICCV 2021
27. Distillation-Guided Image Inpainting, Suin et al., In ICCV 2021
28. High-Fidelity Pluralistic Image Completion With Transformers, Wan et al., In ICCV 2021
29. Learning a Sketch Tensor Space for Image Inpainting of Man-Made Scenes, Cao et al., In ICCV 2021
30. Learning High-Fidelity Face Texture Completion Without Complete Face Texture, Kim et al., In ICCV 2021