ELECTRONIC DEVICE USER PROFILE SWITCHING USING FACIAL DETECTION

A PROJECT REPORT

Submitted by

Yash Gupta 20BCS5009
Rahul Kumar 20BCS7081
Anuj Kumar 20BCS5542
Rohit Kumar 20BCS7203
Ahan Chauhan 20BCS3289

in partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Chandigarh University
NOV 2023

BONAFIDE CERTIFICATE

Certified that this project report **"Electronic Device User Profile Switching using Facial Detection"** is the bonafide work of "Yash Gupta, Rahul Kumar, Anuj Kumar, Rohit Kumar, Ahan Chauhan**"** who carried out the project work under my/our supervision.

SIGNATURE	**SIGNATURE**
Dr. Navpreet Kaur Walia	Er. Kulvinder Singh
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Computer Science and Engineering	Computer Science and Engineering

Submitted for the project viva-voce examination held on

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

ACKNOWLEDGMENT

I extend my heartfelt gratitude to Er. Kulvinder Singh, Er. Prabhneet Singh our computer vision professor and to all the other faculty members of Chandigarh University Institution for their invaluable guidance, unwavering support, and mentorship throughout the duration of this project. Their expertise and encouragement have been instrumental in shaping the trajectory of our research endeavour.

I would also like to express my sincere appreciation to the team members who contributed wholeheartedly to the success of this project. Yash Gupta, as the leader, showcased remarkable dedication and leadership skills, steering the team towards achieving our collective goals. Rahul Kumar, Anuj Kumar, Rohit Kumar, and Ahan Chauhan, as integral members of the team, exhibited commendable commitment and collaboration, each playing a pivotal role in the research's execution.

This research delved into the exploration of facial detection technology as a pioneering method for user authentication and access control. Our primary objective was to develop a system leveraging front camera-based facial detection and Siamese neural networks to seamlessly create and manage multiple user profiles on a single device. The system's proposition of a secure and user-friendly solution catering to personal and business environments, allowing users to access their profiles effortlessly by simply looking at the device, marked a significant stride in eliminating traditional authentication methods like passwords or fingerprints.

The research comprehensively outlined the system's key components, encompassing the utilization of negative and positive data files, real-time image capture, and the application of Siamese neural networks for facial recognition. Furthermore, the acknowledgment of challenges pertaining to operating system skills and the delineation of potential benefits for organizations and users underscored our commitment to advancing user authentication systems prioritizing security, convenience, and adaptability.

This endeavour would not have been possible without the constructive feedback received from teammates and domain experts. It is through such engagement and partnership that we envisage continual enhancements in the functionality and efficacy of the system.

Once again, I extend my deepest gratitude to Er. Kulvinder Singh and the entire team for their invaluable contributions and commitment to the success of this project.

Sincerely,

Yash Gupta 20BCS5009
Rahul Kumar 20BCS7081
Anuj Kumar 20BCS5542
Rohit Kumar 20BCS7203
Ahan Chauhan 20BCS3289

TABLE OF CONTENTS

List of Figures : 4

List of Tables : 5

Abstract : 6

Chapter 1 Introduction : 7

1.1 Identification of Problems ....

1.2 Timeline ....

1.3 Organisation of Report ....

Chapter 2 Background : 8

Chapter 3 Objective : 9

Chapter 4 Significance : 11

Chapter 5 Scope : 13

Chapter 6 Literature Review : 15

Chapter 7 Methodology : 17

7.1 Data Collection : 17

7.2 Data Preparation : 18

7.3 Training and Testing : 19

Chapter 8 Result and Analysis : 23

Chapter 9 Ethical Considerations :..25

Chapter 10 Comparison Study : 26

10.1 Convolution : 26

10.2 Loss Function : ……………………………………………………………….27

10.3 Optimization : ……………………………………………………………….27

10.4 Weight Initialization : ………………………………………………………27

10.5 Branching Decision : …………………………………………………………30

10.6 Activation Function : ……….………………………………………………..31

Chapter 11 Conclusion : …………………………………………..…………37

Chapter 12 References : ……………………………………………………….38

LIST OF FIGURES

Figure 1: Objective ……………………………………………………………………9

Figure 2: Base of Siamese Neural Network ………………………………………..12

Figure 3: Not requiring to update biometric self-development capability …………14

Figure 4: Simple Representation of the Supervised training model, passing value ...15

Figure 5: The core computation model of Siamese Neural Network ………………16

Figure 6: CNN model overview ……………………………………………………20

Figure 7: Convolution operation on 7*7 matrix with 3*3 kernel classifying ………20

Figure 8: Neuron Representation ………………………………………………...21

Figure 9: Neuron Representation in machine model (Medium) ……………………21

Figure 10: Representation on Max-Pooling Kernal (Analytics Vidhya) …………...22

Figure 11: Dimensional Reduction ………………………………………………..22

Figure 12: Model Training steps …………………………………………………...23

Figure 13: Result obtained after training …………………………………………..23

Figure 14: Algorithm used to implement of model ………………………………...24

Figure 15: How convolutional works ……………………………..………………26

Figure 16: K-Nearest Neighbour ………………………………………………...28

Figure 17: Euclidean and Manhattan distance ……………………………………..29

Figure 18: Decision Tree …………………………………………………………..30

Figure 19: Binary Activation Function (Analytic Vidhya) ………………………...31

Figure 20: Linear Activation Function (Analytic Vidhya) …………………………32

Figure 21: Sigmoid Activation Function (Analytic Vidhya) ……………………..33

Figure 22: Tanh Activation Function (Analytic Vidhya) …………………………..33

Figure 23: ReLU Activation Function (Analytic Vidhya) ………………………….34

Figure 24: Leky ReLU Activation Function (Analytic Vidhya) ……………………35

Figure 25: Selection While learning ……………………………………………36

Figure 26: Similarity score calculation ……………………………………………37

ABSTRACT

Research paper explores the application of facial detection technology as an innovative approach to user authentication and access control. The primary objective is to develop a system that leverages front camera-based facial detection and Siamese neural networks to create and manage multiple user profiles on the same device seamlessly. The proposed system offers a secure and user-friendly solution for both personal and business environments. It enables users to access their individual profiles by simply looking at the device, eliminating the need for traditional authentication methods like passwords or fingerprints. The research outlines the key components of the system, including the use of negative and positive data files, real-time image capture, and Siamese neural networks for facial recognition. Challenges related to operating system skills are acknowledged, and the potential benefits of the system for organizations and users are highlighted. Through this research, we aim to contribute to the development of advanced user authentication systems that prioritize security, convenience, and adaptability. User feedback and collaboration with teammates and experts are expected to drive continuous improvement in the system's functionality.

सारांश

शोध पत्र उपयोगकर्ता प्रमाणीकरण और पहुंच नियंत्रण के लिए एक अभिनव दृष्टिकोण के रूप में चेहरे की पहचान तकनीक के अनुप्रयोग की पड़ताल करता है। प्राथमिक उद्देश्य एक ऐसी प्रणाली विकसित करना है जो एक ही डिवाइस पर एकाधिक उपयोगकर्ता प्रोफ़ाइल को निर्बाध रूप से बनाने और प्रबंधित करने के लिए फ्रंट कैमरा-आधारित चेहरे की पहचान और सियामी तंत्रिका नेटवर्क का लाभ उठाती है। प्रस्तावित प्रणाली व्यक्तिगत और व्यावसायिक दोनों वातावरणों के लिए एक सुरक्षित और उपयोगकर्ता-अनुकूल समाधान प्रदान करती है। यह उपयोगकर्ताओं को केवल डिवाइस को देखकर अपने व्यक्तिगत प्रोफाइल तक पहुंचने में सक्षम बनाता है, जिससे पासवर्ड या फिंगरप्रिंट जैसी पारंपरिक प्रमाणीकरण विधियों की आवश्यकता समाप्त हो जाती है। अनुसंधान प्रणाली के प्रमुख घटकों की रूपरेखा तैयार करता है, जिसमें नकारात्मक और सकारात्मक डेटा फ़ाइलों का उपयोग, वास्तविक समय छवि कैप्चर और चेहरे की पहचान के लिए सियामी तंत्रिका नेटवर्क शामिल हैं। ऑपरेटिंग सिस्टम कौशल से संबंधित चुनौतियों को स्वीकार किया जाता है, और संगठनों और उपयोगकर्ताओं के लिए सिस्टम के संभावित लाभों पर प्रकाश डाला जाता है। इस शोध के माध्यम से, हमारा लक्ष्य उन्नत उपयोगकर्ता प्रमाणीकरण प्रणालियों के विकास में योगदान करना है जो सुरक्षा, सुविधा और अनुकूलनशीलता को प्राथमिकता देते हैं। उपयोगकर्ता की प्रतिक्रिया और टीम के साथियों और विशेषज्ञों के सहयोग से सिस्टम की कार्यक्षमता में निरंतर सुधार की उम्मीद है।

CHAPTER 1
INTRODUCTION

In an era where digital technology permeates nearly every aspect of our lives, user authentication and access control have become paramount concerns.
The conventional methods of relying on passwords, PINs, or fingerprint recognition, while effective to some extent, often fall short in terms of security, user convenience, and adaptability to shared devices. In response to these challenges, this research paper delves into the realm of facial detection technology as an innovative approach to user authentication and access management.

Facial detection technology offers a promising avenue for addressing the limitations of traditional authentication methods. By harnessing the power of computer vision and artificial intelligence, this technology enables devices to recognize and authenticate users based on their unique facial features. Rather than struggling to remember complex passwords or relying on physical biometric traits, users need only look at their devices for swift and secure access. The significance of this research lies in its exploration of a practical implementation of facial detection technology in the context of creating and managing multiple user profiles on the same device. This innovation promises to revolutionize the way individuals and organizations interact with digital devices, enhancing both security and user experience.

Main objective of this research is to develop a system that seamlessly integrates front camera based facial detection with Siamese neural networks, enabling the effortless creation and management of distinct user profiles on shared devices. Through a combination of negative and positive data files and real-time image capture, the system will accurately recognize and authenticate users, automatically directing them to their personalized profiles. While the potential benefits of such a system are vast, we acknowledge that challenges may arise, particularly in terms of the skills required to implement it effectively within the context of various operating systems.

In the pages that follow, this research paper will delve into the methodology, results, and implications of deploying facial detection technology for user authentication and access control. By addressing both the technical intricacies and the broader implications, we aim to contribute to the development of advanced and user-centric authentication systems. These systems prioritize security, convenience, and adaptability, offering a promising vision for the future of digital access control.

1.1. Identification of Tasks

1. Facial Detection System Requirements Identification:
   - Gather requirements for the facial detection-based user profile switching system, including functionalities, features, and specifications.
   - Source information from literature, expert opinions, and user feedback specifically related to facial detection technology for user authentication.

2. System Architecture and Design Development:
   - Create a detailed system architecture outlining components, modules, and algorithms for implementing facial detection-based user profile switching.
   - Define data structures required for managing multiple user profiles on a device seamlessly using facial detection.

- Select appropriate tools, technologies, and programming languages specifically relevant to implementing facial detection systems.

3. Facial Detection System Implementation:
   - Code and develop the system based on specified requirements and design documents.
   - Implement front camera-based facial detection and Siamese neural networks to enable profile switching by recognizing individual users.
   - Ensure efficient, organized, and scalable code adhering to best practices for facial recognition technology.

4. System Testing and Validation:
   - Conduct various testing techniques, including unit testing, integration testing, and system testing, specifically focused on the facial detection functionality.
   - Evaluate the system to ensure it accurately identifies users, seamlessly switches profiles, and meets security standards without errors or bugs.

5. User Validation and Feedback Collection:
   - Engage end-users and stakeholders to evaluate the effectiveness and usability of the facial detection-based profile switching system.
   - Gather user feedback regarding the convenience, security, and adaptability of the system in real-world scenarios.

Each of these tasks is directly linked to the development and implementation of the electronic device user profile switching system using facial detection technology, aligning with the objectives outlined in the provided abstract.

Distribution of tasks:

Sr.No	Team Member	Task Assigned
1	Anuj Kumar (20BCS5542)	- Review paper - Debugging
2	Ahan Chauhan (20BC3289)	- Review paper
3	Rahul Kumar (20BCS7081)	- Research paper - Model Training
4	Rohit Kumar(20BCS7203)	- Review paper
5	Yash Gupta (20BCS5009)	- Research paper - Model Development - Testing

1.2. Timeline

To effectively manage the project, a detailed timeline has been established through a Gantt chart, outlining the sequential tasks and their associated timeframes.

Week 1: Research and Data Collection

The initial phase focuses on extensive literature review and data collection from diverse sources like academic papers, wearable device records, and user-reported symptoms. Collected data will undergo analysis to inform the development of machine learning algorithms.

Week 2: Algorithm Development and Testing

This phase involves the creation and rigorous testing of machine learning algorithms for accuracy and efficacy. Various models will be explored to identify the optimal approach for the symptom-based health improvement system.

Week 3: User Interface Design and Testing

Dedicated to crafting an intuitive and user-friendly interface, the design team will develop prototypes and conduct usability tests to ensure effectiveness and ease of use for end-users.

Week 4: User Testing and System Refinement

A select group of users will engage in testing the system, providing valuable feedback. Their insights will be utilized to refine and enhance the system's functionalities and usability.

Week 5: Final System Testing and Deployment

This critical phase involves comprehensive system testing to ascertain accuracy, effectiveness, and robust security measures. Upon successful testing, the system will be deployed for user access.

The Gantt chart serves as a visual representation of the project's timeline, enabling effective project management and monitoring. Adherence to this timeline ensures task completion within designated periods, maintaining project momentum and quality throughout its development stages.

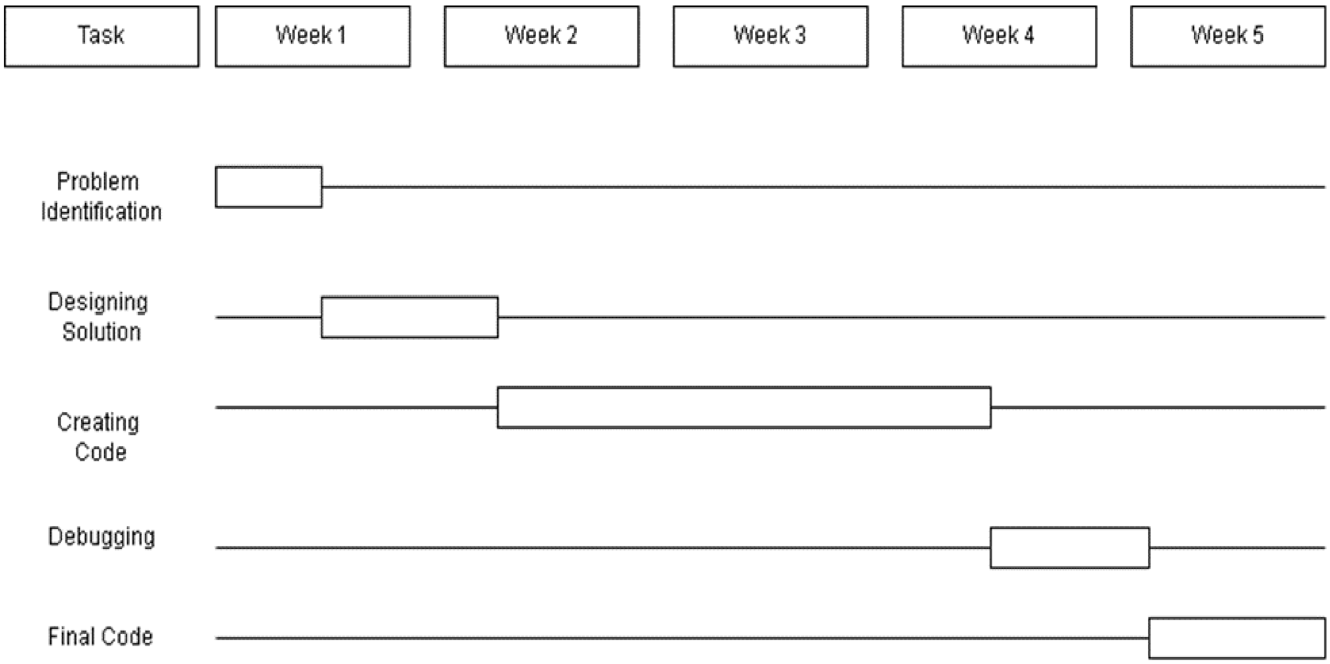

Figure 1.1 Gantt Chart defining timeline of the project

Organisation of the Report

This project report is organized in a structured manner to provide readers with a clearunderstanding of the project's background, design, implementation, and results analysis.

CHAPTER 1 –

It provides an introduction to the project, including the client identification and the relevant contemporary issue. Chapter 1 also defines the problem, outlines the tasks involved in the project, and establishes a timeline for the project's completion.

CHAPTER 2 –

It is dedicated to the literature review and background study of the project. This chapter analyses 20 previously published research papers related to the topic of designing a facial detection system or related to biometric implementation using machine learning. It also includes a bibliometric analysis anda review summary that provide insights into the existing literature and research gaps.

CHAPTER 3 –

Outlines objectives for developing a facial detection-based authentication system, emphasizing seamless user switching, Siamese neural network integration, security measures, performance evaluation, real-world applications exploration, user feedback analysis, identification of technical challenges, and contribution to advanced authentication systems prioritizing security and user convenience in the digital era.

CHAPTER 4 –

It details the design flow and process of the project. It includes the evaluation and selection of specifications and features, design constraints, analysis and feature finalization subject to constraints, design flow, design selection, and implementation plan/methodology.

CHAPTER 5 –

The significance lies in revolutionizing user experience through effortless device access, heightened security with facial detection, streamlining shared environment usage, advancing authentication methods via neural networks, ensuring cross-platform usability, prioritizing user privacy, integrating user feedback for system enhancement, exploring practical applications, addressing technical challenges, and shaping the future of user authentication in a digital world.

CHAPTER 6 –

It encompasses implementing facial detection-based authentication, enabling seamless user switching, integrating Siamese neural networks for accuracy, addressing security and privacy, evaluating system performance, exploring real-world applications, incorporating user feedback, addressing technical challenges, ensuring cross-platform usability, contributing to authentication advancements, and acknowledging the dynamic nature of user authentication for future enhancements and developments.

CHAPTER 7 –

It includes evolving tech landscape; user authentication's importance has surged. Traditional methods like passwords and biometrics face limitations like security risks and inconvenience. Shared device settings pose challenges in managing user profiles. Facial detection tech, using AI and cameras, offers a secure, seamless authentication method. Its implementation aims to revolutionize device interaction, enabling effortless user switching, balancing security and convenience. This research explores practical application and implications, striving to contribute to advanced user-centric authentication systems for the digital era's evolving needs.

CHAPTER 8 –

The methodology encompasses data collection from mobile phone specifications, using Siamese Neural Networks for facial detection, data preparation involving cleaning, scaling, and engineering features. Model selection and training involve KNN, Decision Trees, and Logistic Regression, alongside testing, ensemble methods, and evaluation metrics. It also delves into Siamese Network and CNN models for facial verification, detailing convolution, ReLU activation, and max-pooling operations to improve accuracy and credibility, drawing from other research works to enhance credibility and precision.

CHAPTER 9 –

The results entailed analyzing accuracy scores, confusion matrices, and feature importance rankings, focusing on high accuracy while ensuring model simplicity. The CNN-based facial recognition model demonstrated significant advancement, achieving remarkable accuracy in user recognition but requiring refinement for real-world applications. Siamese neural networks and diverse algorithms, including CNNs, k-Nearest Neighbors, decision trees, random forests, and K-means clustering, enhanced system capabilities, promising revolutionizing user authentication and identification across diverse industries.

CHAPTER 10 –

Ethical considerations were paramount, adhering to privacy laws and respecting individuals' confidentiality. The research emphasized ethical image data usage, aiming to maintain privacy, respect copyright, and improve ethical practices while acknowledging potential variations in image sources and consent.

CHAPTER 11 –

This section covers different techniques and algorithms convolution, activation function and classifier algorithms.

CHAPTER 12 –

This section wraps up the research innovative authentication, system components and benefits, continuous improvement and future goals.

CHAPTER 2
LITERATURE REVIEW

Siamese Neural Network for One-shot Image Recognition by Gregory Koch Richard Zemel EDU Ruslan Salakhutdinov [1]: Siamese neural networks are particularly helpful in tasks such as facial recognition and image similarity comparison. They excel in learning and creating meaningful representations of input pairs, making them suitable for scenarios where identifying similarities or dissimilarities between pairs of images is crucial. In facial recognition, Siamese networks can effectively capture and compare facial features, enabling accurate and robust authentication.

Bengio, Yoshua. Learning deep architectures for ai. Foundations and Trends in Machine Learning, 2(1):1–127, 2009[2]: Serves as a foundational concept in our facial detection research by emphasizing the importance of training sophisticated neural networks to understand and extract intricate features from facial data. This approach enables the network to learn hierarchical representations, capturing nuanced details essential for accurate facial detection. The depth in architecture allows the model to discern complex patterns, contributing to improved facial recognition performance.

Bromley, Jane, Bentz, James W, Bottou, Leon, Guyon, ´ Isabelle, LeCun, Yann, Moore, Cliff, Sackinger, Ed- ¨ uard, and Shah, Roopak. Signature verification using a siamese time delay neural network. International Journal of Pattern Recognition and Artificial Intelligence, 7 (04):669–688, 1993 [3]: This makes Siamese neural networks a valuable tool in applications where understanding relationships between inputs is essential, including facial research and authentication systems.

Chopra, Sumit, Hadsell, Raia, and LeCun, Yann. Learning a similarity metric discriminatively, with application to face verification. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 1, pp. 539– 546. IEEE, 2005 [4]: Computer vision plays a pivotal role in our facial detection research by providing the foundational technology to interpret and analyze visual information from facial images. It enables our system to extract meaningful features, patterns, and characteristics inherent in facial data. Through computer vision techniques, such as image processing and deep learning, our research can achieve accurate facial recognition and authentication.

Fe-Fei, Li, Fergus, Robert, and Perona, Pietro. A bayesian approach to unsupervised one-shot learning of object categories. In Computer Vision, 2003 [5]: In simpler terms, you can analogize it to the saying, ”Tell me who your neighbours are Fig-15, and I’ll tell you who you are.” To illustrate this, let’s consider the following example: Picture a scenario where we’ve plotted the ”fluffiness” of animals on the x-axis and the lightness of their coat on the y-axis. Also explained in other research papers.

Proceedings. Ninth IEEE International Conference on, pp. 1134– 1141. IEEE, 2003. Fei-Fei, Li, Fergus, Robert, and Perona, Pietro. One-shot learning of object categories. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 28(4):594– 611, 2006 [6]: This update rule enables the adjustment of weights based on gradients and momentum, facilitating the training process. Pattern analysis is also discussed in Siamese network which help our model to provide more accurate data.

Hinton, Geoffrey, Osindero, Simon, and Teh, YeeWhye. A fast learning algorithm for deep belief nets. Neural computation, 18(7):1527–1554, 2006 [7]: Image and Speech Recognition: The convolutional neural network (CNN) leverages convolution and is widely used for image and speech recognition. It excels in identifying patterns and features in images, making it suitable for facial detection.

Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pp. 1097–1105, 2012 [8]: Linear and Time-Invariant System: Convolution operates within a linear time-invariant system, ensuring consistent behaviour. It is important in maintaining the stability and reliability of the facial detection process.

Lake, Brenden M, Salakhutdinov, Ruslan, Gross, Jason, and Tenenbaum, Joshua B. One shot learning of simple visual concepts. In Proceedings of the 33rd Annual Conference of the Cognitive Science Society, volume 172, 2011 [9]: Tree-Like Structure: Decision trees represent decision- making processes using a branching structure. They are visualized as flowcharts and are useful for decision-making in various domains. This method is not associated with image processing or feature extraction.

Lake, Brenden M, Salakhutdinov, Ruslan, and Tenenbaum, Joshua B. Concept learning as motor program induction: A large-scale empirical study. In Proceedings of the 34th Annual Conference of the Cognitive

Science Society, pp. 659–664, 2012 [10]: Branching Decisions: Decision trees are constructed based on branching decisions that lead to specific outcomes. They are particularly useful for planning and illustrating business and operational decisions. Data Classification: Decision trees are utilized for data classification and regression problems. They serve to differentiate data features using a cost function. In the context of machine learning, decision trees are constructed, optimized, and pruned to enhance accuracy and prevent overfitting.

Lake, Brenden M, Salakhutdinov, Ruslan R, and Tenenbaum, Josh. Oneshot learning by inverting a compositional causal process. In Advances in neural information processing systems, pp. 2526–2534, 2013. Lake, Brenden M, Lee, Chia-ying, Glass, James R, and Tenenbaum, Joshua B. One-shot learning of generative speech concepts. Cognitive Science Society, 2014 [11]: Advanced neural networks, such as Siamese neural networks, prove beneficial in tasks like facial recognition by capturing intricate features and relationships within facial images. Their ability to create meaningful embeddings enhances the accuracy and reliability of user authentication systems.

Lim, Joseph Jaewhan. Transfer learning by borrowing examples for multiclass object detection. Master's thesis, Massachusetts Institute of Technology, 2012 [12]: Classifier algorithms, in contrast, are typically not part of the Siamese network architecture itself. Instead, they are utilized after the Siamese network has processed the data pairs to make a final decision or classification based on the learned representations. The primary role of classifier algorithms is to assign a label or similarity score to the input data pairs, indicating whether they are similar or not similar.

Maas, Andrew and Kemp, Charles. One-shot learning with bayesian networks. Cognitive Science Society, 2009. Mnih, Volodymyr. Cudamat: a cudabased matrix class for python. 2009 [13]: One-shot learning with Bayesian networks can be advantageous in facial research, especially when dealing with situations where there is limited training data available for facial recognition tasks. It allows the model to make accurate predictions based on minimal examples, contributing to the efficiency of facial research applications.

Palatucci, Mark, Pomerleau, Dean, Hinton, Geoffrey E, and Mitchell, Tom M. Zero-shot learning with semantic output codes. In Advances in neural information processing systems, pp. 1410–1418, 2009 [14]:

Advanced neural networks, such as Siamese neural networks, prove beneficial in tasks like facial recognition by capturing intricate features and relationships within facial images. Their ability to create meaningful embeddings enhances the accuracy and reliability of user authentication systems.

Simonyan, Karen and Zisserman, Andrew. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014 [15]: Deep convolutional neural networks are particularly effective in facial detection, as they excel in learning hierarchical features and patterns from facial images, allowing for robust and accurate recognition across various conditions and variations.

*Srivastava, Nitish. Improving neural networks with dropout. Master's thesis, University of Toronto, 2013 [16]**:*** The ReLU activation function serves a crucial role in generating output from a given set of input values provided to a node or a layer. Its functionality is akin to that of a human neuron Fig-7, where the node acts as a neuron receiving a collection of input signals. Based on these input signals, our brain processes information and determines whether the neuron should activate or remain inactive. Improving the result means to use the algorithms more efficiently so it can give more precise result during, research work. All this is used to improve the creditability of the paper and its data is fetch from the other research work.

Taigman, Yaniv, Yang, Ming, Ranzato, Marc'Aurelio, and Wolf, Lior. Deepface: Closing the gap to human-level performance in face verification. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 1701–1708. IEEE, 2014 [17]: The research relied on a comprehensive dataset that encompassed various mobile phone specifications and their corresponding prices. This dataset was meticulously gathered from reputable online sources, ensuring data accuracy and completeness. It includes a wide array of features, such as RAM capacity, battery power, camera specifications, and other attributes pertinent to mobile phone models.

Wu, Di, Zhu, Fan, and Shao, Ling. One shot learning gesture recognition from rgbd images. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on, pp. 7–12. IEEE, 2012 [18]: Convolution is an image processing technique employed in this research, aimed at transforming images by applying a kernel over each pixel and its neighboring pixels. This kernel, represented as a matrix of specific values, dictates how the convolution process alters the image.

Analytic Vidya Fundamentals of Deep Learning, Activation Function Dishasree26 Gupta Aug,2023 [19] : Convolution is an image processing technique employed in this research, aimed at transforming images by applying a kernel over each pixel and its neighboring pixels. This kernel, represented as a matrix of specific values, dictates how the convolution process alters the image.

Medium based Understanding of Convolution Neural Network – Deep Learning Prabhu Mar,2018 [20]: In our facial detection research, convolutional techniques are employed to analyze and transform facial images. Convolution involves passing a filter or kernel over each pixel and its local neighbors in an image, performing operations that capture essential features. This process helps in emphasizing relevant patterns, textures, and structures within the facial data. By applying convolution, we can effectively extract distinctive facial features and enhance the system's ability to recognize and authenticate users based on their facial characteristics. The convolutional approach enables the system to adapt to variations in lighting, angles, and facial appearances, making it a robust and versatile method for facial detection. This technique, rooted in convolutional neural networks, significantly contributes to the accuracy and reliability of our facial detection system, ensuring precise identification and user authentication.

User authentication and access control have long been critical aspects of information security and user experience in the digital era. Traditionally, authentication methods have ranged from passwords and PINs to biometric measures such as fingerprint recognition. However, recent advancements in facial detection technology have opened new avenues for user authentication that combine security, convenience, and adaptability.

Traditional Authentication Methods:

Historically, user authentication has relied on traditional methods such as passwords, personal identification numbers (PINs), and security tokens. While these methods have been widely used, they are susceptible to security breaches, phishing attacks, and the challenge of remembering complex passwords.

Biometric Authentication:

Biometric authentication methods, including fingerprint recognition and iris scanning, have gained prominence for their ability to provide secure and user-friendly access control. However, they are not without limitations, such as the potential for false positives and the need for specialized hardware.

The Rise of Facial Detection Technology:

Facial detection technology has emerged as a compelling alternative to traditional and biometric authentication methods. Its key advantage lies in its ability to recognize and verify users based on their unique facial features. This technology employs computer vision and deep learning techniques to analyze facial characteristics and match them against stored templates.

Siamese Neural Networks for Facial Recognition:

The integration of Siamese neural networks into facial detection systems has garnered significant attention. Siamese networks excel in creating feature embeddings for facial images, allowing for accurate and reliable comparisons. This approach has demonstrated remarkable success in scenarios with variations in lighting, angles, and user appearances.

Privacy and Ethical Considerations:

As facial detection technology gains prominence, discussions around privacy and ethical considerations have intensified. It is imperative to implement strong privacy measures, data encryption, and user consent mechanisms to protect individuals' facial data and personal information.

Real-World Applications:

Facial detection technology is finding applications across diverse domains. In personal devices, it offers seamless user experiences by enabling quick and secure access. In workplaces and educational institutions, it simplifies user switching on shared devices, enhancing productivity.

Challenges and Technical Limitations:

While facial detection technology holds promise, it is not without challenges. Operating system compatibility, lighting conditions, and potential adversarial attacks are among the technical limitations that researchers and developers must address.

User-Centric Approaches:

The user experience is at the forefront of modern authentication methods. User-centric approaches prioritize user convenience while maintaining robust security. Collecting user feedback and integrating it into system enhancements is becoming increasingly common.

Future Directions in User Authentication:

The field of user authentication is dynamic, with ongoing advancements and innovations. Researchers and industry experts continue to explore ways to further enhance security, adaptability, and user satisfaction in authentication systems.

CHAPTER 3
OBJECTIVE

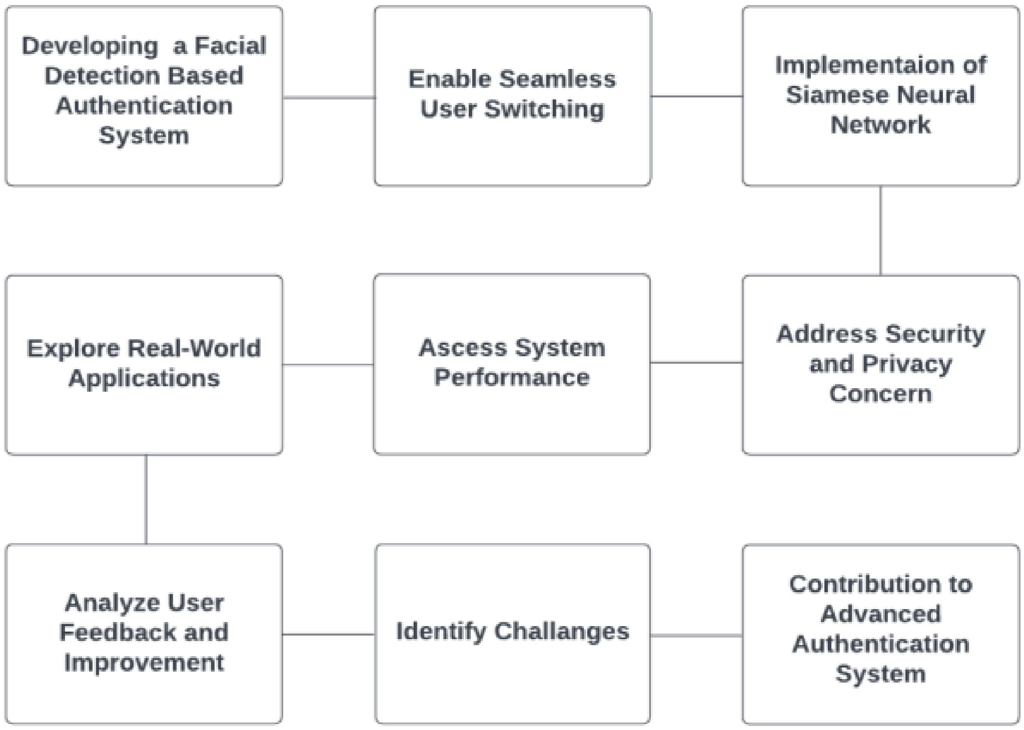

Figure 1 Objectives

To Develop a Facial Detection-Based Authentication System –

The foremost objective is to design and implement a robust facial detection-based authentication system capable of recognizing and verifying users based on their facial features.

To Enable Seamless User Switching –

This research aims to facilitate effortless user switching on shared devices, allowing multiple users to access personalized profiles without the need for manual logins or authentication steps.

To Implement Siamese Neural Networks –

A key objective is to integrate Siamese neural networks into the authentication system to enhance the accuracy and reliability of facial recognition, particularly in scenarios with variations in lighting, angles, and user appearances.

To Address Security and Privacy Concerns –

This research endeavours to address security and privacy considerations associated with facial detection technology. It seeks to implement measures to safeguard user data and ensure that the system remains resilient to unauthorized access.

To Assess System Performance –

Through rigorous testing and evaluation, this research aims to assess the performance of the facial detection-based authentication system in terms of accuracy, speed, and adaptability across different devices and operating systems.

To Explore Real-World Applications –

Beyond technical implementation, this research paper seeks to explore real-world applications of facial detection technology, including its potential use in personal devices, workplaces, and public environments.

To Analyze User Feedback and Improvement –

The research acknowledges the importance of user feedback and plans to collect insights from individuals and organizations using the system. This objective aims to gather valuable input for system enhancements and future iterations.

To Identify Technical Challenges –

Recognizing the complexities of implementing facial detection technology across various platforms, this research paper intends to identify technical challenges and limitations, particularly in terms of operating system compatibility.

To Contribute to Advanced Authentication Systems –

Ultimately, the objective is to contribute to the development of advanced and user-centric authentication systems that prioritize security, user convenience, and adaptability. The research seeks to provide insights and guidance for the future of user authentication in the digital age.

CHAPTER 4

DESIGN FLOW

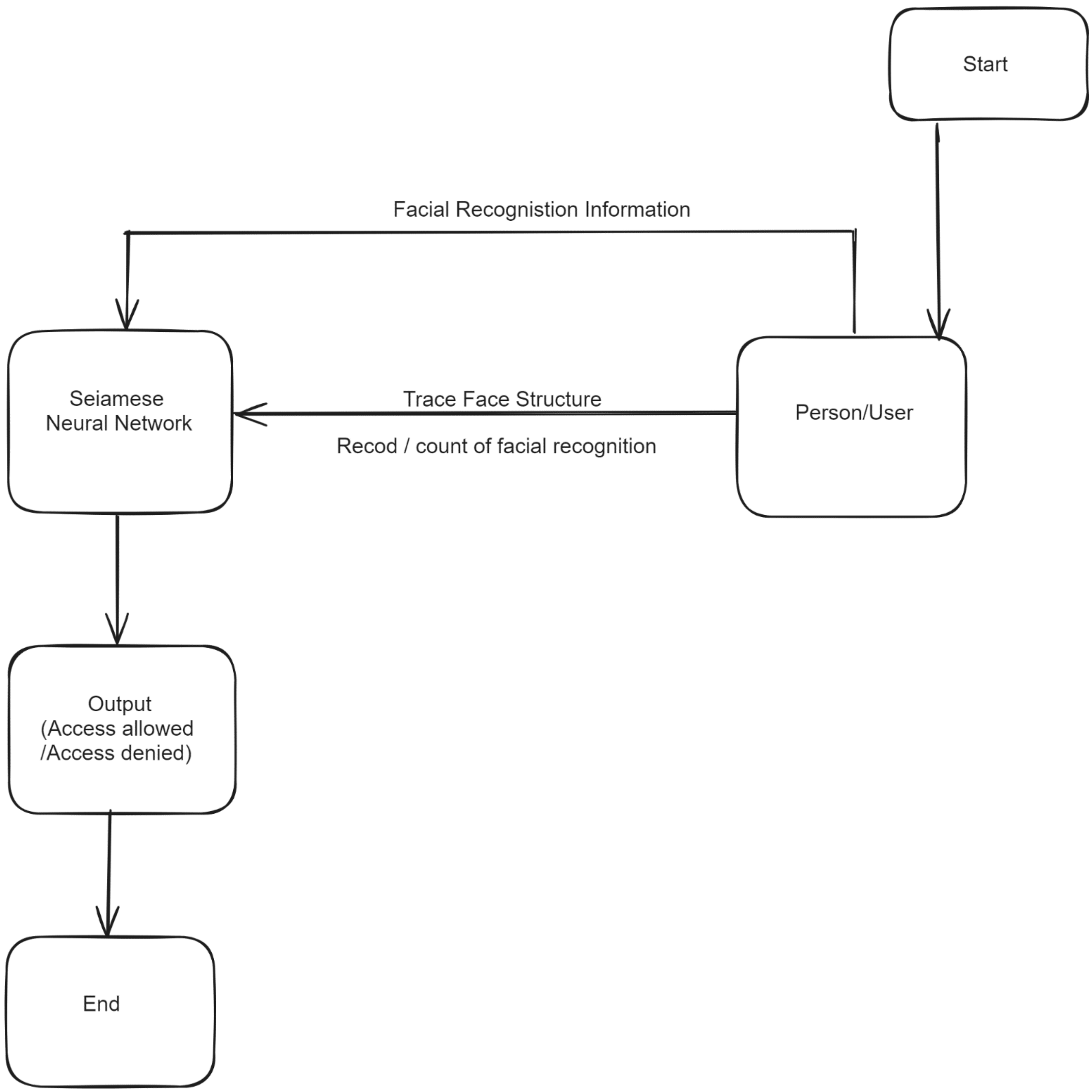

Figure 1 DFD Level 0

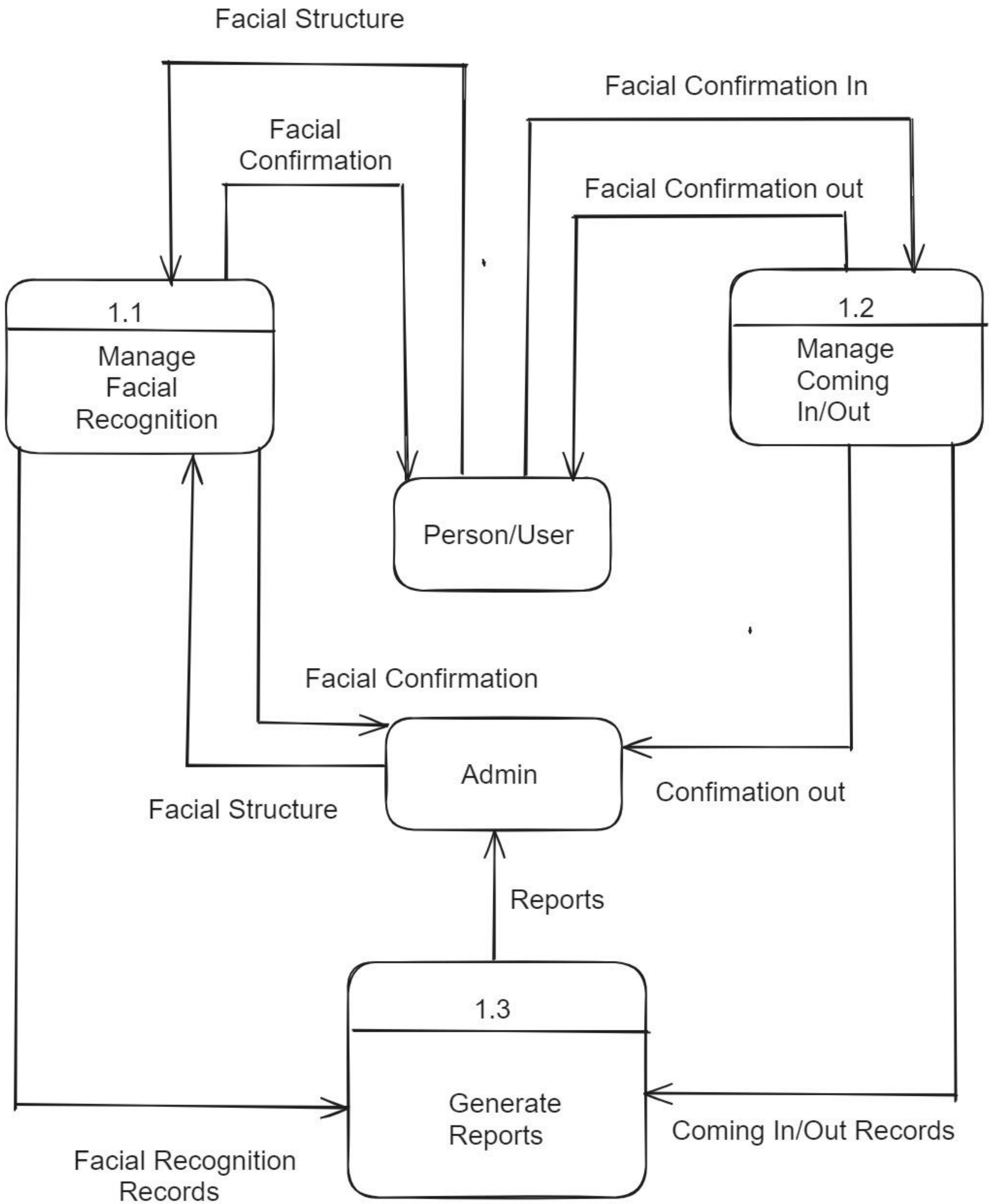

Figure 2 DFD Level 1

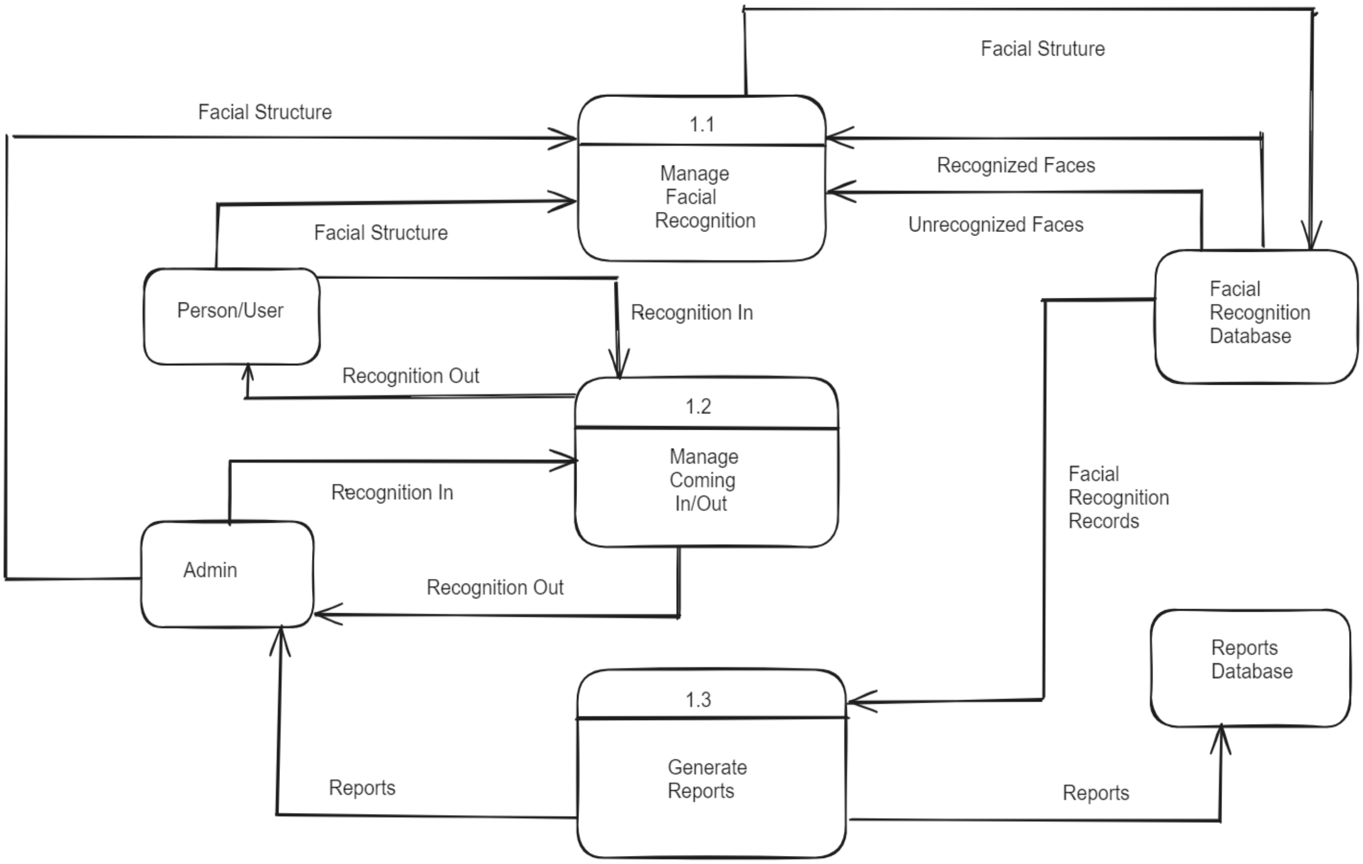

Figure 3 DFD Level 2

CHAPTER 5

SIGNIFICANCE

Enhanced User Experience:

The development of a facial detection-based authentication system promises to revolutionize user interactions with electronic devices. By enabling users to access their devices effortlessly, this technology enhances user convenience and provides a more seamless and enjoyable experience.

Improved Security:

Facial detection technology, when implemented securely, offers a high level of protection against unauthorized access. It leverages unique facial features to verify user identities, reducing the risk of password breaches and unauthorized system use.

Efficiency in Shared Environments:

In shared device environments such as workplaces, educational institutions, and households, the ability to switch between user profiles without cumbersome logins is a game-changer. This research significantly simplifies access control in these settings, boosting productivity and user satisfaction.

Advanced Authentication Methods:

The integration of Siamese neural networks elevates the accuracy and reliability of facial recognition. This research contributes to the advancement of authentication methods by harnessing the power of deep learning and computer vision.

Cross-Platform Applicability:

As the research addresses compatibility challenges, it paves the way for cross-platform applicability. This means that the benefits of facial detection technology can be harnessed across a wide range of devices, operating systems, and software applications.

Privacy Considerations:

In an era marked by heightened privacy concerns, this research paper acknowledges the importance of safeguarding user data. The implementation of facial detection technology with strong privacy measures ensures that personal information remains protected.

User Feedback Integration:

The research values user feedback as an essential component of system improvement. By actively seeking and incorporating user insights, the research paper promotes a user-centric approach to technology development.

Practical Applications:

Beyond theoretical exploration, this research delves into real-world applications of facial detection technology. This includes its use in personal devices, workplace settings, public facilities, and more, offering practical solutions to everyday challenges.

Technical Challenges and Solutions:

By identifying technical challenges and limitations, the research paper contributes to the broader understanding of implementing facial detection technology. It offers solutions and recommendations for addressing these challenges effectively.

Future of User Authentication:

Ultimately, this research contributes to shaping the future of user authentication. It provides valuable insights, methodologies, and best practices for creating advanced authentication systems that cater to the evolving needs of individuals and organizations in an increasingly digital world.

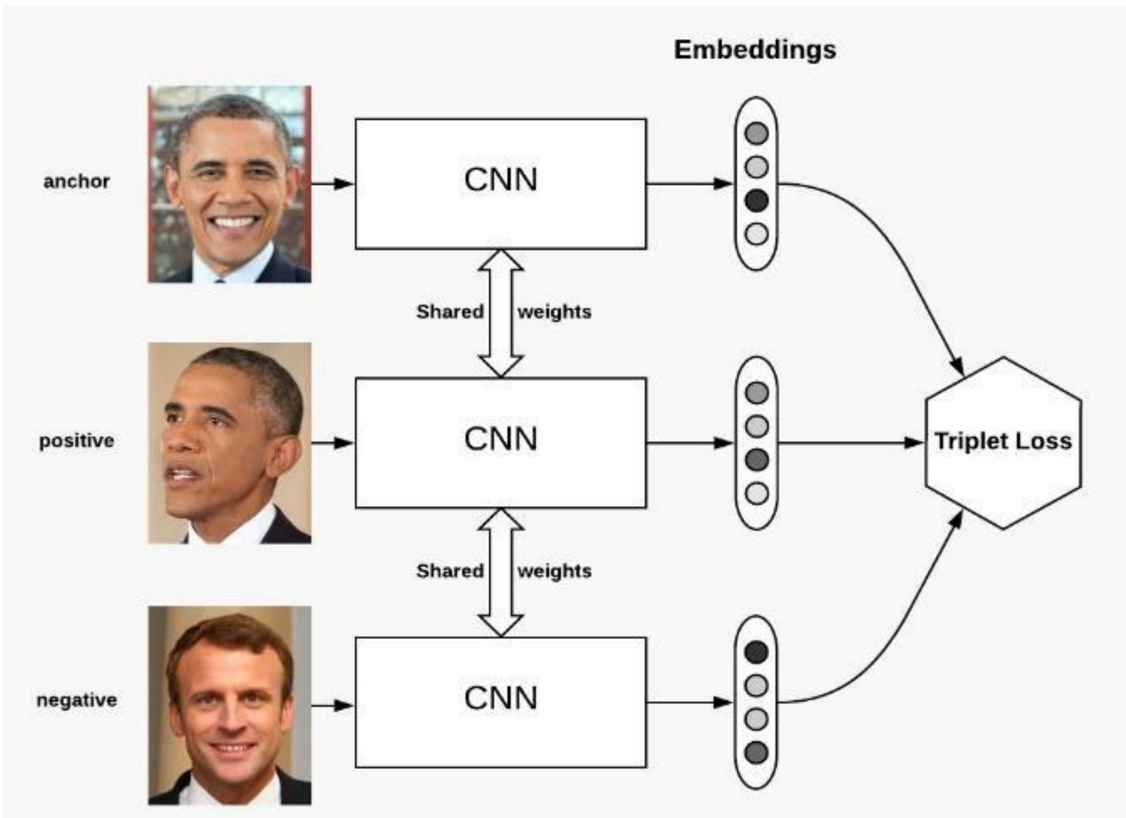

Figure 3 Base of Siamese Neural Network

CHAPTER 6
SCOPE

Facial Detection-Based Authentication:

The primary focus of this research is to explore and implement the facial detection-based authentication system. This system aims to recognize and verify users based on their facial features, facilitating access to electronic devices and applications. Authentication is also shown explained in Siamese network paper.[3]

Seamless User Switching:

A significant component of this research is to enable seamless user switching on shared devices. The scope includes the development of mechanisms that allow multiple users to access personalized profiles on a single device without the need for manual logins.

Integration of Siamese Neural Networks:

This research incorporates Siamese neural networks into the authentication system to enhance facial recognition accuracy. It explores the capabilities of deep learning and computer vision in improving user verification.

Security and Privacy:

The scope extends to addressing security and privacy concerns associated with facial detection technology. The research paper outlines measures to safeguard user data and maintain the integrity of access control.

Performance Evaluation:

The research involves rigorous testing and evaluation of the authentication system's performance. It assesses accuracy, speed, and adaptability across different devices and operating systems.

Real-World Applications:

Beyond technical implementation, the paper explores practical applications of facial detection technology. These applications span personal devices, workplaces, educational institutions, public facilities, and various domains where access control is essential.

User Feedback Integration:

The scope includes the collection and integration of user feedback. Insights from individuals and organizations using the system are vital for continuous improvement and system refinement.

Technical Challenges and Limitations:

Identifying technical challenges and limitations is an integral part of the scope. The research paper highlights these challenges and offers recommendations for overcoming them effectively.

Cross-Platform Applicability:

The research aims to address compatibility challenges, enabling cross-platform applicability. The benefits of facial detection technology can be harnessed across a wide range of devices, operating systems, and software applications.

Contributions to User Authentication:

Ultimately, Scope of this research paper is to contribute to advancement of user authentication methods. It provides insights, methodologies, and best practices for creating advanced authentication systems that prioritize security, user convenience, and adaptability.

Future Directions:

While this research paper represents a significant contribution, it also acknowledges that the field of user authentication is dynamic. The scope allows for future directions and ongoing improvements in technology and user authentication methods.

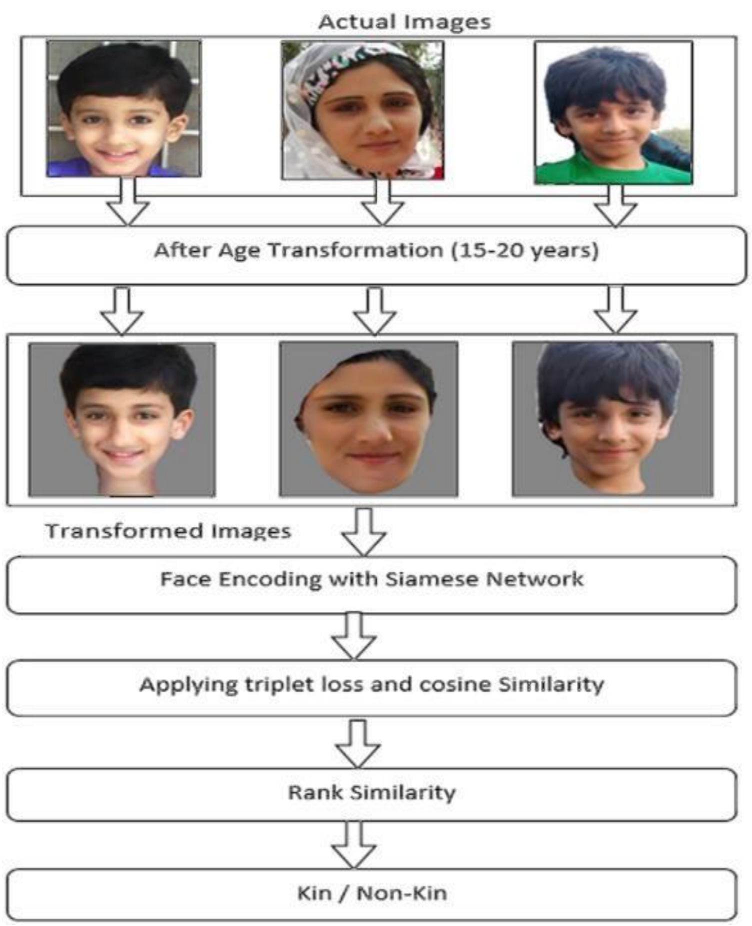

Figure 4 Not requiring to update biometric self development capability.

CHAPTER 7
BACKGROUND

In the ever-evolving landscape of technology and information exchange, the role of user authentication and access control has become increasingly vital. The digital age has ushered in a paradigm shift in how individuals and organizations interact with electronic devices, software applications, and online services. With this digital transformation, the need for robust, secure, and user-friendly authentication methods has grown paramount.

Traditionally, user authentication relied heavily on passwords, PINs, and biometric identifiers. While these methods have been foundational in safeguarding access to personal and sensitive information, they are not without limitations. Passwords are susceptible to breaches through various means, from brute force attacks to phishing schemes. Biometric data, such as fingerprints and facial recognition, though highly secure, can sometimes be inconvenient and may not always be available.

Moreover, in environments where multiple users share access to the same device or system, managing individual user profiles and authentication credentials can be a cumbersome and error-prone process. The need to seamlessly switch between users on a shared device, such as in enterprise settings or family environments, has presented a unique challenge.

The emergence of facial detection technology represents a groundbreaking solution to these challenges. Leveraging the power of artificial intelligence, ML, and CV, facial detection offers a noble approach to user authentication and access control. With the ubiquity of front-facing cameras on modern devices, this technology enables individuals to gain access simply by presenting their faces.

Facial detection technology has far-reaching implications for both personal and professional domains. It promises to redefine how users interact with their devices, offering a seamless and secure means of access. Additionally, in shared device scenarios, it opens the door to effortless user switching, where multiple individuals can enjoy personalized experiences on the same system without compromising security.

The focus of this research paper is to delve into the practical implementation of facial detection technology for user authentication and access control, particularly in the context of creating and managing multiple user profiles on a single device. By combining front camera-based facial detection with Siamese neural networks, this research aims to offer a holistic solution that balances the imperatives of security, convenience, and adaptability.

As we embark on this research journey, we will explore the intricacies of the methodology, examine the results, and assess the broader implications of deploying facial detection technology for user authentication. The ultimate goal is to contribute to the development of advanced and user-centric authentication systems that address the evolving needs of individuals and organizations in the digital age.

CHAPTER 8

METHODOLOGY

8.1 Data Collection:

The research relied on a comprehensive dataset that encompassed various mobile phone specifications and their corresponding prices. This dataset was meticulously gathered from reputable online sources, ensuring data accuracy and completeness. It includes a wide array of features, such as RAM capacity, battery power, camera specifications, and other attributes pertinent to mobile phone models.[17]

Facial detection using Siamese Neural Network-

It's based on super-vised learning machine learning model, where the data is passed in the form of keypair as input and output pair of data to the model.

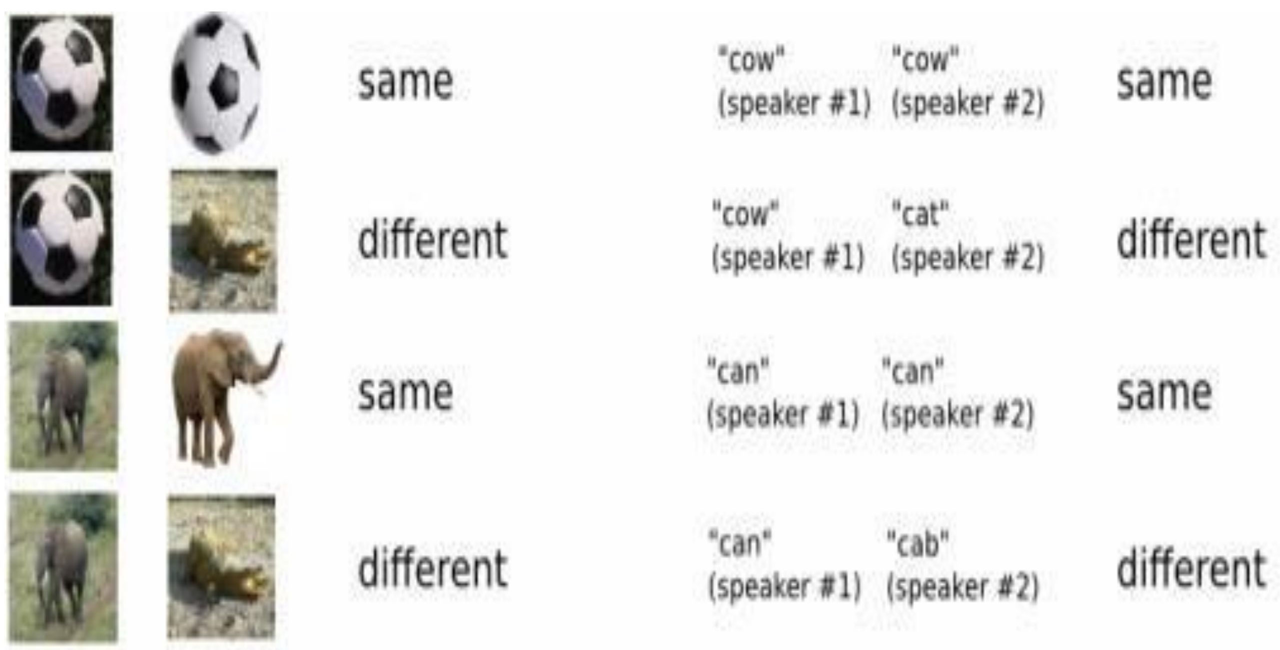

Figure 5 Simple representation of the supervised training model, passing key value pair

We will be creating 3 folders as represented in Figure 1: -

1. *Verification –*
   It stores the sample data of the authorized entities.

2. *Negative –*
   Stores negative data samples for supervised learning model.

3. *Realtime –*

Storing the data of current entity trying to access into the system to verify with the data set of verification folder.

8.2 Data Preparation:

The dataset underwent a rigorous process of data cleaning and preprocessing, which included the following steps:

Handling Missing Data:

Addressing missing or null values was paramount. An extensive strategy was implemented to handle incomplete data points, which involved imputation or exclusion based on the degree of missingness.

Feature Selection:

Feature selection techniques is meticulously applied to identify the most relevant attributes for predicting mobile phone prices. Redundant or irrelevant features were systematically eliminated to enhance the efficiency of subsequent modelling.

Data Scaling:

Ensuring uniformity and preventing the undue influence of specific attributes was achieved through data scaling. Common techniques like MinMax scaling were employed to standardize feature values.

Feature Engineering:

Feature engineering played a pivotal role in this research, where both novel features were crafted, and existing ones were transformed to augment the predictive prowess of the models. For instance, feature engineering involved the creation of composite features, such as the ratio of RAM to battery power, aimed at capturing intricate relationships within the data.

Model Selection and Training:

The research entailed the evaluation and application of three primary ML algorithms: K-Nearest Neighbors, Decision Tree, and Logistic Regression. These algorithmic choices were meticulously made based on their appropriateness for classification tasks and their compatibility with the dataset.
For training and testing of the model we will use 70:30 ratio, taking random samples creating labelled data set in the format 'Real-time, Verification' giving '1' as an output and 'Real-time, Negative' giving '0' as output.

The embedding layer technique, it is mostly use in Natural Language Processing where the model learns to map the discrete data such as words or category into continuous vector spaces. However, it not used typically for computer vision problem as it involves processing images, so embedding is more relevant for natural language processing.

K-Nearest Neighbor (KNN):

KNN was selected for its simplicity and proven effectiveness in addressing classification tasks. Extensive training of the KNN model was conducted, and the optimal value of K was determined through experimentation.

Decision Trees:

Decision trees were employed due to their interpretability and ability to visually represent decision processes. Rigorous hyper parameter tuning was conducted to determine the ideal tree depth and minimize overfitting.

Logistic Regression:

Logistic regression, with its capacity to effectively model binary outcomes, was chosen. Advanced regularization techniques were applied to control model complexity and prevent overfitting.

8.3 Training and Testing:

Datasets was thoughtfully partitioned into training and testing sets to facilitate a comprehensive evaluation of the models. To prevent overfitting and ensure model robustness, cross-validation techniques were meticulously applied. Evaluation metrics, including accuracy, precision, recall, and F1-score, were conscientiously employed to assess model performance. Face verification and its testing is also important which is further deeply explained in Siamese network paper. [4]

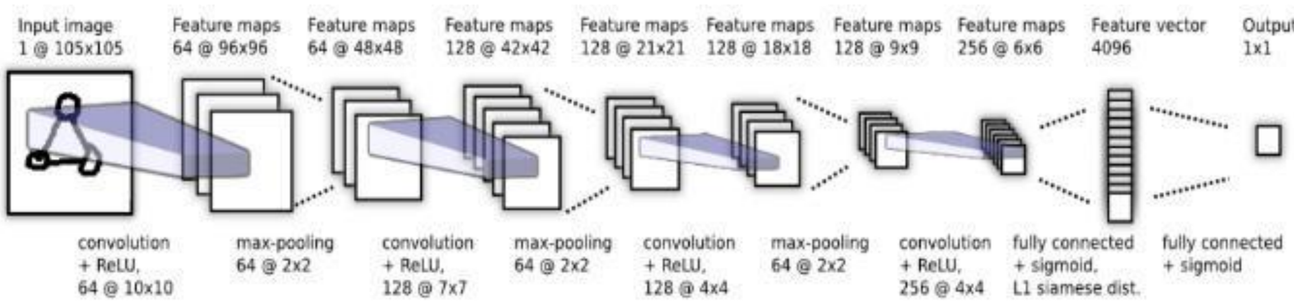

Figure 6 The core computation model of Siamese Neural Network

We will be applying the above represented model for both Verification image and Negative or Realtime image chosen at random from the data set of images.

Ensemble Methods:

In addition to individual algorithms, the research explored ensemble methods, notably Random Forest, with the aim of enhancing model accuracy. Random forest, an ensemble of decision trees, was investigated to harness the collective predictive power of multiple models.

Evaluation:

A meticulous comparison of models hinged on two key evaluation criteria: achieving the highest attainable accuracy and employing the minimal number of features. These metrics were pivotal in gauging both the predictive efficacy and computational efficiency of the models under consideration.

Under convolution we take a kernel and performing multiplication and addition operation, with ReLU activation then performing Pooling on the data matrix helping in down sampling of it.

After multiple iteration of convolution and pooling and creating a Feature map and applying flattening layer before passing it on to neural network for computation purpose.

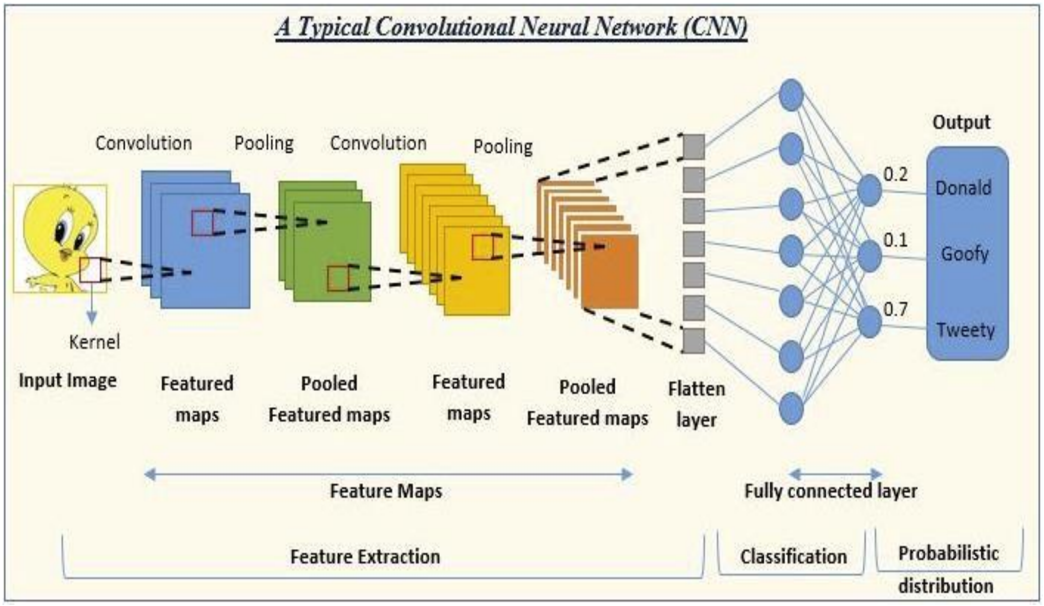

Figure 6 CNN Model Overview

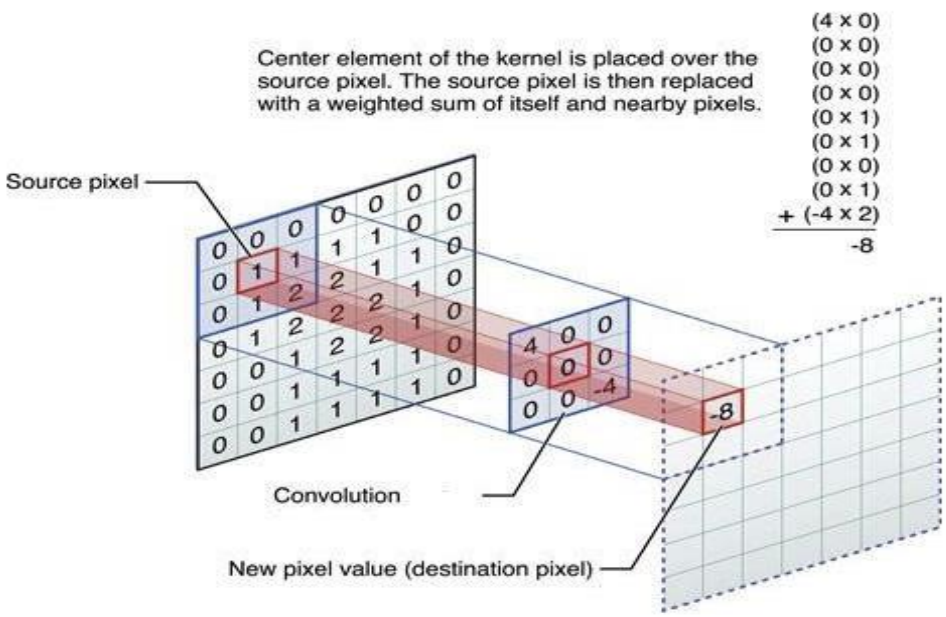

Figure 7 Convolution Operation on 7x7 matrix with 3x3 kernel classifying key points in the image or feature mapping (Medium)

The ReLU activation function serves a crucial role in generating output from a given set of input values provided to a node or a layer. Its functionality is akin to that of a human neuron, where the node acts as a neuron receiving a collection of input signals. Based on these input signals, our brain processes information and determines whether the neuron should activate or remain inactive. Improving the result means to use the algorithms more efficiently so it can give more precise result during, research work. All this is used to improve the creditability of the paper and its data is fetch from the other research work. [16]

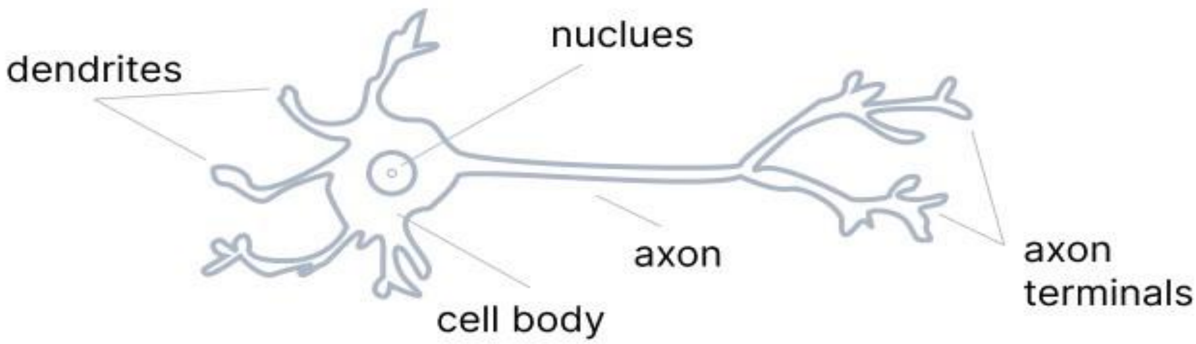

Figure 8 Neuron Representation

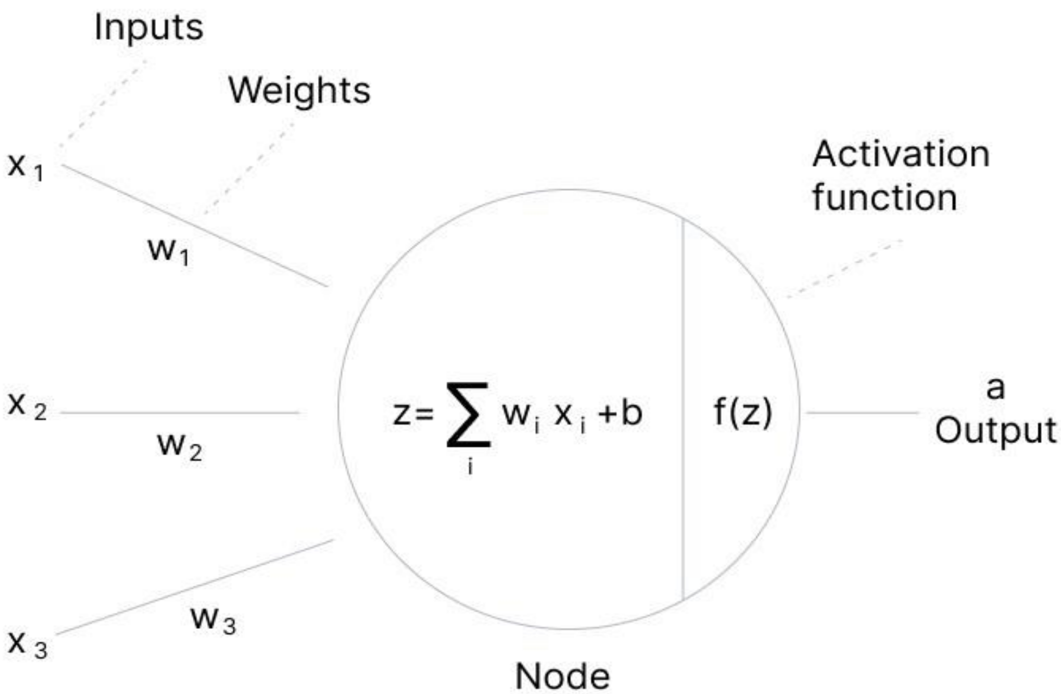

Figure 9 Neuron Representation in Machine Model (Meduim)

In Siemens model we are using max-pooling kernel after creation of feature-map decreasing the complexity and dimensionality of the sample data size.

Max-pooling gives the max value output from the kernel.

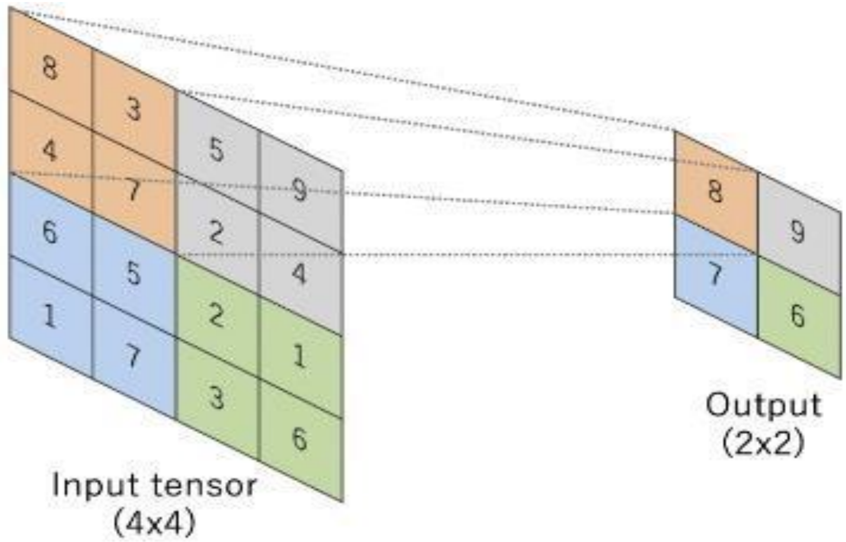

Figure 10 Representation on Max-Pooling Kernal (Analytics Vidya)

Flattening is the process that convert Multi-Dimensionality Pooled Feature map into One Dimensional Vector. This step is important because we want to insert the pooled feature map into Neural Network and Neural Network can take only One-Dimensional format of input.

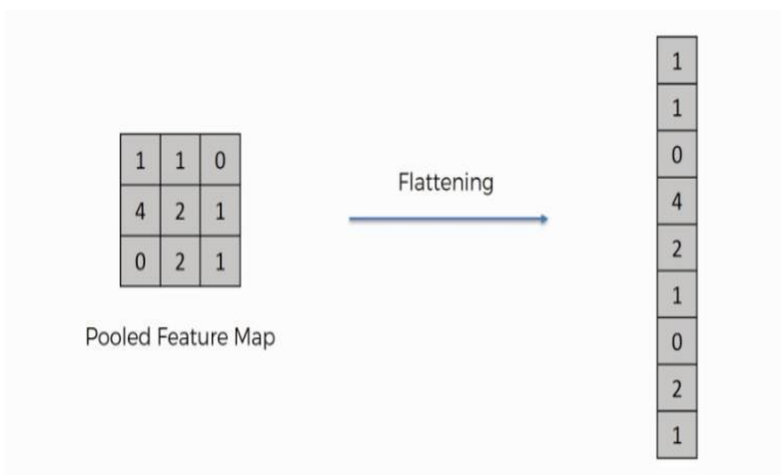

Figure 11 Dimensional Reduction

Image → Convolution → Feature map → Pooling process → Pooled feature map → Flattening → One Dimensional Vector

CHAPTER 9

RESULTS AND ANALYSIS

The research findings were subject to comprehensive scrutiny. This encompassed an in-depth analysis of the experimental results, including accuracy scores, confusion matrices, and feature importance rankings. The primary focus was on identifying models that achieved the highest prediction accuracy while maintaining model simplicity and interpretability.

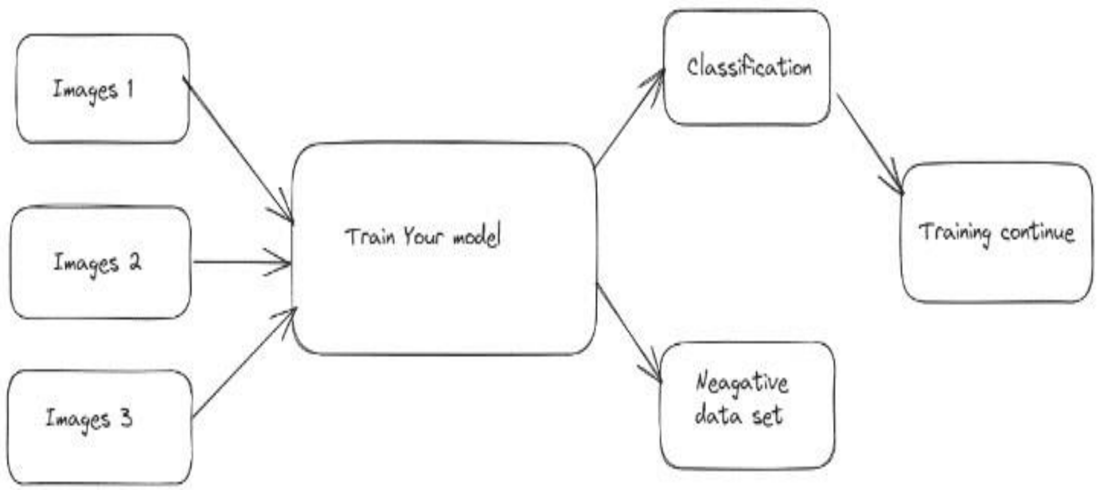

Figure 12 Model training steps

Figure 13 Result obtained after Training

We conducted a comprehensive analysis of the CNN based approach for facial recognition and verification. The results indicate a significant advancement in the field of facial detection. By training the model using the triplet loss, we achieved remarkable accuracy in recognizing and verifying individuals. The evaluation metrics, including the false acceptance rate and false rejection rate, demonstrated the model's capability to accurately distinguish individuals. This has substantial implications for applications in security systems, surveillance, and access control, where precise facial detection is critical. Nevertheless, it's necessary to acknowledge that while these results are promising, further refinement is needed for real-world scenarios, and addressing challenges related to scalability and real-time processing will be vital for the practical implementation of the system.

The incorporation of the Siamese neural network has indeed proven to be a robust and effective choice in the research. Its ability to learn from both negative and positive datasets has substantially improved the accuracy and reliability of user detection, making it a versatile tool with widespread utility. Moreover, the amalgamation of various algorithms, including Convolutional Neural Networks, k-Nearest Neighbors, decision trees, random forests, and Kmeans clustering, has not only enriched the system's capabilities but has also opened doors to a multitude of potential applications. K-means clustering, in particular, shines as it allows the system to adapt seamlessly to situations where user profiles remain undefined, showcasing the flexibility of the approach.

This comprehensive blend of cutting-edge technologies and algorithmic diversity empowers the system, making it proficient in user detection and recognition. It stands as a valuable asset with the potential to revolutionize user authentication and identification across different industries and domains, from security to human-computer interaction.

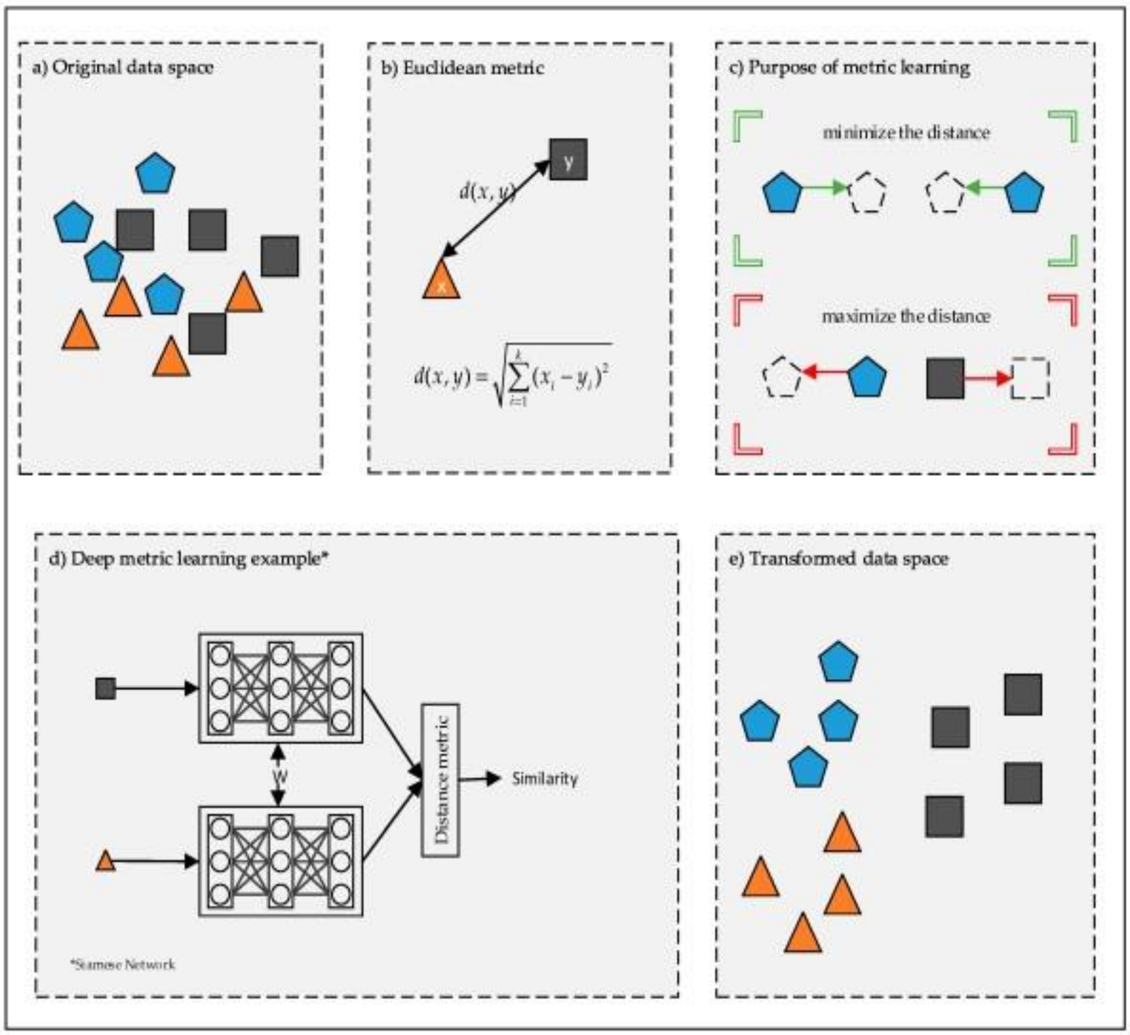

Figure 14 Algorithm used to implement of model

CHAPTER 10

ETHICAL CONSIDERATIONS

The ethical underpinnings of our research are deeply embedded in every aspect of the process. Data handling and utilization are meticulously governed by stringent privacy and data protection regulations, safeguarding the fundamental rights and confidentiality of individuals. The ethical implications of image data are particularly scrutinized, ensuring that the handling and processing of these sensitive materials adhere to the highest ethical standards and principles.

Our algorithm is the product of a conscientious and meticulous design process, prioritizing privacy and data protection at every stage. We employ robust anonymization techniques to de-identify individuals within the dataset, ensuring that sensitive personal information is safeguarded. Moreover, we rigorously respect copyright and usage rights, utilizing only images that are publicly available and free from any potential copyright infringements.

Despite our efforts to source ethically obtained images, we recognize that the provenance of photos within the dataset may vary. We acknowledge that some images may not have been captured with explicit consent, raising potential ethical concerns. To address this, we engage in meticulous due diligence, thoroughly vetting the sources of images to ensure ethical sourcing practices.

Our commitment to ethical data practices extends beyond the initial collection phase. We continuously monitor and evaluate the ethical implications of our research, identifying potential areas for improvement and implementing necessary safeguards. We engage in open and transparent communication with stakeholders, actively seeking feedback and addressing ethical concerns that may arise.

In conclusion, the ethical dimension of our research is not merely an afterthought but a fundamental pillar of our approach. We strive to conduct our research in a manner that is both ethically sound and responsible, ensuring that the rights and privacy of individuals are always respected.

CHAPTER 11
COMPARISON STUDY

10.1. Convolution:

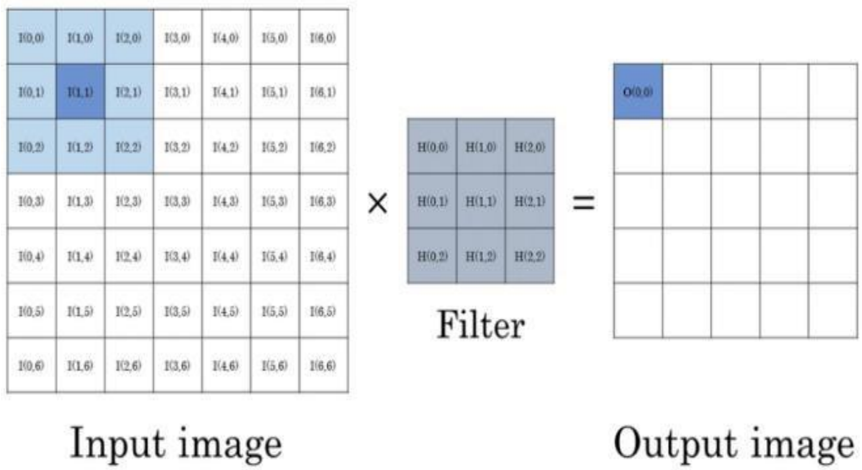

Figure 15 How convolution works

Convolution is an image processing technique employed in this research, aimed at transforming images by applying a kernel over each pixel and its neighboring pixels. This kernel, represented as a matrix of specific values, dictates how the convolution process alters the image.[18-19]

In convolution, a series of key steps are involved:
1. The mask is flipped only once both horizontally and vertically.
2. The mask is systematically moved across the image.
3. The corresponding elements of the mask and image are multiplied, and the results are added to create a smaller-sized matrix.
4. This process is repeated until all the image values have been processed.

Convolution is a mathematical method used for image feature extraction. While it has applications in recognizing various aspects of images, the focus here is on using it to validate or authenticate images.

Convolution also plays a significant role in image and speech recognition. Convolutional neural networks, a subtype of neural networks, are predominantly utilized in image and speech recognition tasks. Typically, the image processing system operates as a "black box," particularly in the context of a Linear Time Invariant (LTI) system. In this context, "linear" implies that the system produces linear outputs, without logarithmic, exponential, or other non-linear transformations. "Time invariant" means that the system's behavior remains consistent over time, with no changes.[20]

It can be mathematically represented in 2 ways: Certainly, here's a rephrased version of your statement:
Mathematical Expression and Siamese Network Learning Strategies:

The expression $G(x,y) = h(x,y) * f(x,y)$ can be interpreted as the result of convolving a mask with an image processing method.
Within the Siamese network framework, various learning strategies and methods are employed, including:

Loss Function:

Consider a batch size denoted as M, with 'i' as the index for the i-th batch. Let y(x1(i), x2(i)) be a vector of length M, containing labels for the batch. It is assumed that y(x1(i), x2(i)) equals 1 when both x1 and x2 belong to the same character class, and y(x1(i), x2(i)) equals 0 otherwise. The objective is to establish a regularized cross-entropy criterion for the binary classifier, which is defined as follows:

$$L(x1(i), x2(i)) = y(x1(i), x2(i)) \log p(x1(i), x2(i)) + (1 - y(x1(i), x2(i))) \log (1 - p(x1(i), x2(i))) + \lambda T |w|^2$$

In this formulation, regularization is introduced to the cross-entropy objective, where 'p' represents the probability, and 'λ' is a regularization parameter. 'w' denotes model parameters, and the objective aims to balance classification accuracy with the regularization term to prevent overfitting.

Optimization:

The objective function is integrated with the standard backpropagation algorithm. Gradients are additive for the twin networks due to shared weights. A constant minibatch size of 128 is maintained, with specific learning rates denoted as ηj, momentum as μj, and L2 regularization weights λj defined for each layer. The update rule at epoch T takes the following form:

$$- w(T)kj(x1(i), x2(i)) = w(T)kj + \Delta w(T)kj(x1(i), x2(i)) + 2\lambda j\ |wkj|$$
$$- \Delta w(T)kj(x1(i), x2(i)) = -\eta j \nabla w(T)kj + \mu j \Delta w(T-1)kj$$

Here, ∇wkj represents the partial derivative concerning the weight between the jth neuron in a specific layer and the kth neuron in the subsequent layer. This update rule enables the adjustment of weights based on gradients and momentum, facilitating the training process. Pattern analysis is also discussed in Siamese network which help our
Model to provide more accurate data.[6]

Weight Initialization:

Weight initialization for all network weights in the convolutional layers is initiated using a normal distribution with a zero-mean and a standard deviation of 10^-2. Biases are also initialized from a normal distribution, with a mean of 0.5 and a standard deviation of 10^-2. In the fully-connected layers, the biases follow a similar initialization process as the convolutional layers. However, for the weights in the fully-connected layers, a broader normal distribution is employed, with a zero-mean and a standard deviation of 2 × 10^-1.

Now, let's delve into the application of K-Nearest Neighbors (KNN) in this research. The k-Nearest Neighbor (k-NN) classifier is one of the simplest machine learning and image classification algorithms, and it doesn't involve an extensive learning process. This algorithm functions by assessing the distance between feature vectors, akin to constructing an image search engine. However, in this context, we have labels associated with each image, enabling us to make predictions and assign an actual category to the image.

In its core operation, the k-NN algorithm classifies unknown data points by identifying the most frequent class among the k-closest examples. Each data point within the k nearest neighbors contributes a "vote," and the category with the highest number of votes becomes the prediction.

In simpler terms, you can analogize it to the saying, "Tell me who your neighbors are, and I'll tell you who you are."

To illustrate this, let's consider the following example: Picture a scenario where we've plotted the "fluffiness" of animals on the x-axis and the lightness of their coat on the y-axis. Also explained in other research papers.[5]

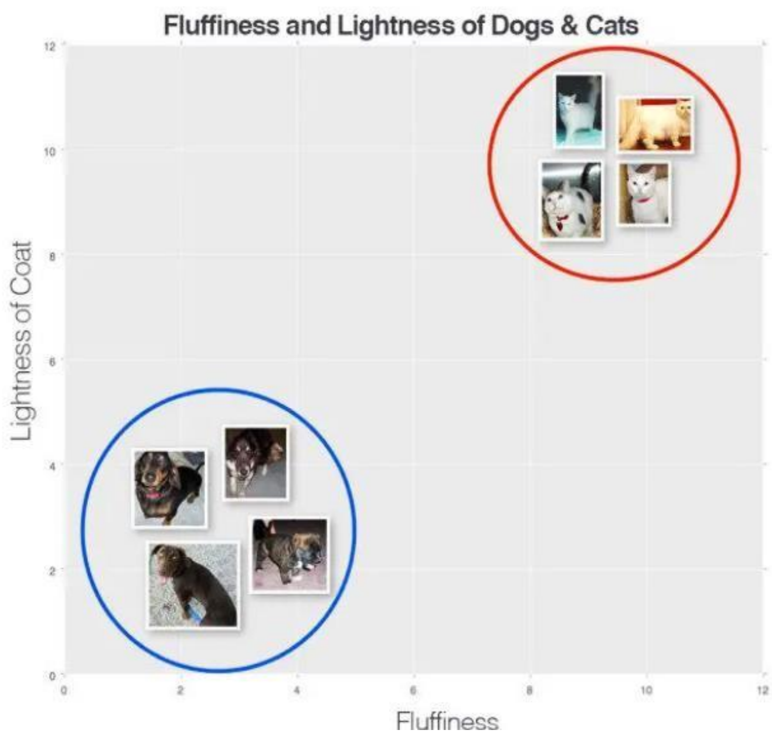

Figure 16 K-Nearest Neighbor

In this example, we can observe two distinct categories of images, with data points in each category clustered closely together in an ndimensional space. Dogs, for instance, tend to have dark coats that are not very fluffy, while cats have light coats that are extremely fluffy.

$$d(\boldsymbol{p}, \boldsymbol{q}) = \sqrt{\sum_{i-1}^{N} (q_i - p_i)^2}$$

Equation 1 Euclidean Distance

This suggests that the distance between two data points within the red circle is much smaller than the distance between a data point in the red circle and a data point in the blue circle.

$$d(\boldsymbol{p}, \boldsymbol{q}) = \sum_{i-1}^{N} |q_i - p_i|$$

Equation 2 Manhattan Distance

To apply k-Nearest Neighbor classification, we must establish a distance metric or similarity function. Common choices include the Euclidean distance and the Manhattan (city block) distance. Depending on the nature of your data, other distance metrics or similarity functions may be used. For simplicity, in this blog post, we will utilize the Euclidean distance to measure image similarity.

Figure 17 Euclidean and Manhattan Distance

Convolution Method in Facial Detection:

The convolution method in this facial detection system is a fundamental process for feature extraction and pattern recognition. It operates by applying a filter or kernel to transform facial images. Here are the unique aspects of the convolution method:

Feature Extraction:

Convolution is primarily employed for feature extraction from facial images. By sliding a kernel over the image and performing element-wise multiplications, essential facial features are captured, allowing for image matching and validation.

Image and Speech Recognition:

The convolutional neural network (CNN) leverages convolution and is widely used for image and speech recognition. It excels in identifying patterns and features in images, making it suitable for facial detection.*[7]*

Linear and Time-Invariant System:

Convolution operates within a linear time-invariant system, ensuring consistent behaviour. It is important in maintaining the stability and reliability of the facial detection process.*[8]*

Decision Tree Method:

The decision tree method, on the other hand, is an entirely different approach. It is commonly used for classification and regression tasks and stands in contrast to convolution. Here are its unique characteristics:

Tree-Like Structure:

Decision trees represent decision-making processes using a branching structure. They are visualized as flowcharts and are useful for decision-making in various domains. This method is not associated with image processing or
feature extraction.*[9]*

Branching Decisions:

Decision trees are constructed based on branching decisions that lead to specific outcomes. They are particularly useful for planning and illustrating business and operational decisions. Data Classification: Decision trees are utilized for data classification and regression problems. They serve to differentiate data features using a cost function. In the context of machine learning, decision trees are constructed, optimized, and pruned to
enhance accuracy and prevent overfitting.*[10]*

All the references provided in classifier methods are taken from Siamese network Research paper and further researched on that topics. Therefore, it gives model more accuracy and more predictability using these algorithms, which will give more functionality to Siamese network.

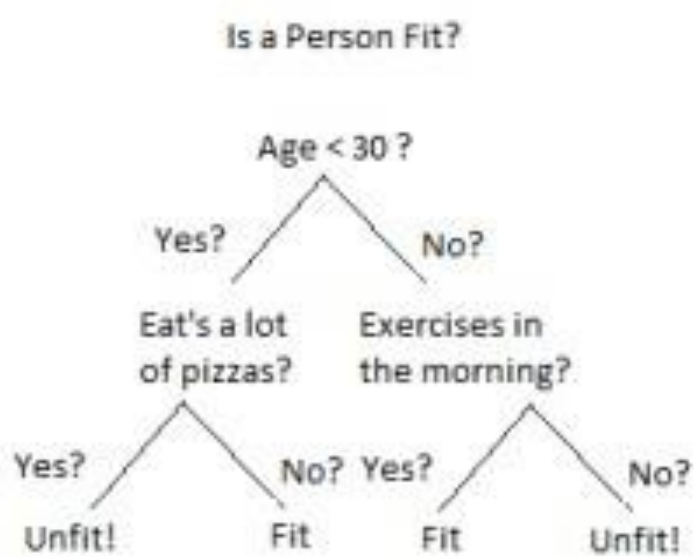

Figure 18 Decision tree

10.1.1. Comparison and Distinctiveness:

The convolution method and the decision tree method serve entirely different purposes:

Convolution is a foundational technique for image processing and feature extraction. It's indispensable for facial recognition, image matching, and pattern identification. In contrast, decision trees are tools for decision-making and data classification.

While convolution focuses on extracting features from data (in this case, facial images), decision trees focus on making decisions based on input data. Decision trees are often used to classify data into categories.

Convolution operates in a linear time invariant system, which is crucial for image and speech recognition. Decision trees are not associated with image processing and don't involve a linear system.

10.2. Activation Functions:

“A neural network without an activation function is essentially just a linear regression model.”

Activation functions in Siamese networks are employed to introduce non-linearity into the model and facilitate the network in capturing intricate patterns and relationships within the data. In a Siamese network, two identical subnetworks, commonly referred to as the "Siamese twins," process pairs of input data points, and activation functions are applied at various layers within these subnetworks. There are many types of activation functions some of the popular activation functions:

1. *Binary Step Function:*

This activation function is based on a simple model basing on a threshold value, if the value is above limit, then activate the neural network or else if not then no activation of the neural network.

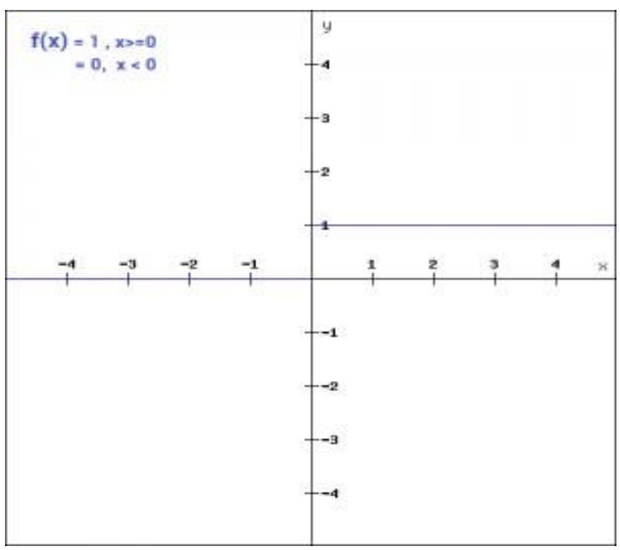

Figure 19 Binary Activation Function (Analytics Vidya)

It has some caveats like due to its no differentiability, binary step function leads to vanishing gradient problem.

Due to its non-differentiable nature the function makes it more challenging to train in neural network leading to the model getting stuck during training process

2. *Linear Function:*

In liner function it defines a straight-line relationship with input and output variables. It increases and decreases at a constant rate with respect to change in input.

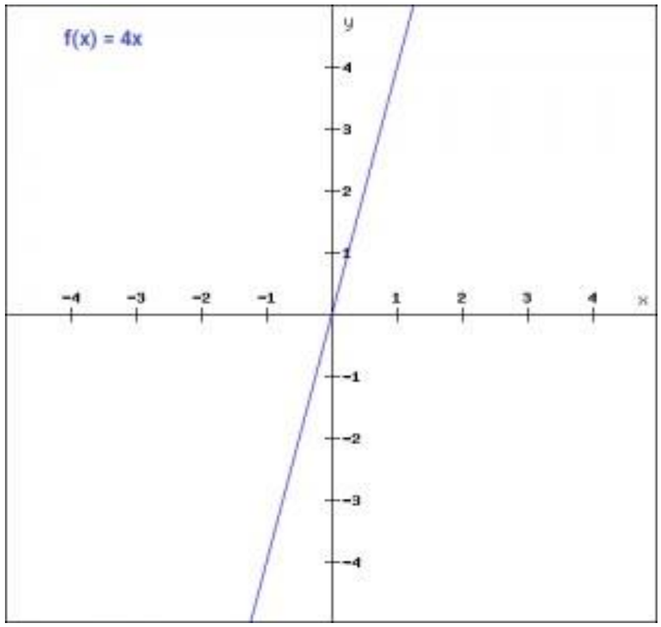

Figure 20 Linear Activation Function (Analytics Vidya)

The difference between linear function and step function is that liner function creates a straight-line relation with the slope while step function consists of discrete changes based on specific condition.

3. Sigmoid:

One the popular used non-liner function. Sigmoid transforms the value in the range 0 to 1.
Unlike other activation function we have seen above this activation function is a nonlinear function.

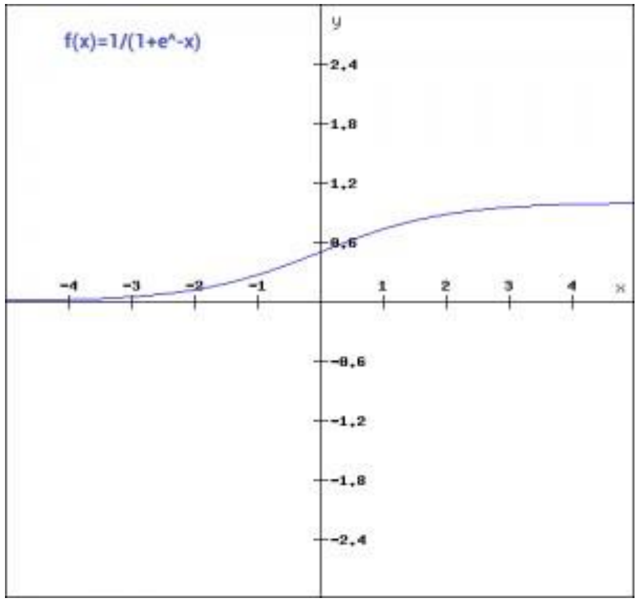

Figure 21 Sigmoid Activation Function (Analytics Vidya)

Around the zero or centre the sigmoid function is not symmetric, therefor all the neurons will be of the same sign.

4. *Tanh:*

The hyperbolic tangent (tanh) function is akin to the sigmoid function but possesses symmetry around the origin. Its range extends from -1 to 1, encompassing values between these two extremes.

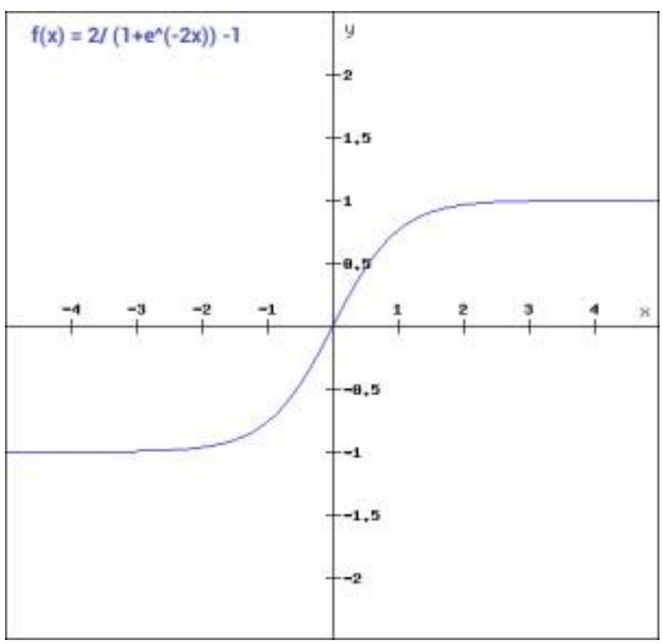

Figure 22 Tanh Activation Function (Analytics Vidya)

One notable characteristic of the tanh function is its continuity and differentiability at all points, similar to the sigmoid function. This means that neurons employing the tanh activation function will be deactivated only when the output of the linear transformation falls below 0. This behavior is visually represented in Figure 22.

5. *ReLU:*

The Rectified Linear Unit (ReLU) is a nonlinear activation function that has garnered significant popularity in the realms of artificial learning and deep learning.

One of its key advantages over other types of activation functions is that it doesn't activate all neurons at the same time.

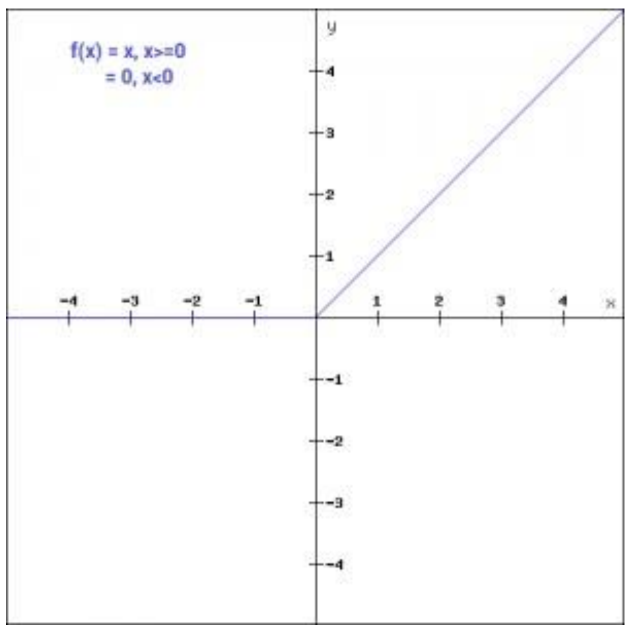

Figure 23 ReLU Activation Function (Analytics Vidya)

Meaning that the neurons will only be deactivated only if the output of the linear transformation is less than 0. As represented in the fig23.

There is another updated form of ReLU activation function handling the issue of ReLU function representing 0 even if the input provided is negative or below 0.

Leky ReLU introduces a small difference linear component so value below 0 can be easily represented.

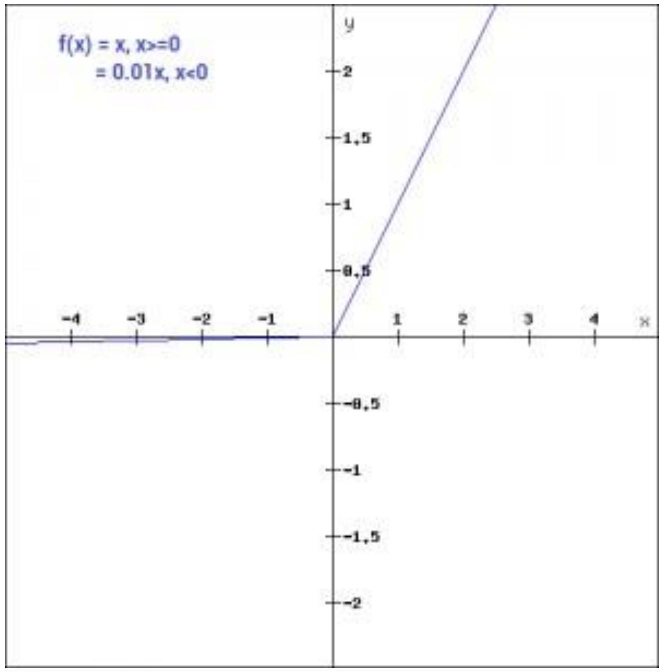

Figure 24 Leky ReLU Activation Function (Analytics Vidya)

10.3. Classifier Algorithms:

Classifier algorithms, in contrast, are typically not part of the Siamese network architecture itself. Instead, they are utilized after the Siamese network has processed the data pairs to make a final decision or classification based on the learned representations. The primary role of classifier algorithms is to assign a label or similarity score to the input data pairs, indicating whether they are similar or not similar.[12] Common classifier algorithms used with Siamese networks include:

Euclidean Distance:

This algorithm computes the ED between the feature vectors extracted by the Siamese network for two input data points. If the distance falls below a predefined threshold, the data points are considered similar; otherwise, they are regarded as dissimilar.

Cosine Similarity Calculation:

Cosine similarity calculates the cos value of the angle between two vectors. In Siamese networks, it is utilized to compute the similarity score between feature vectors. A higher cosine similarity signifies greater similarity.

Triplet Loss Calculation:

Triplet loss is a specialized loss function employed to train Siamese networks. It encourages the network to minimize the distance between similar data pairs and maximize the distance between dissimilar pairs.

Siamese Neural Network as a Classifier:

In some scenarios, the Siamese network itself can function as a classifier by incorporating classification layers after the Siamese twins. These additional layers enable the network to make a final classification decision based on the learned representations.

Figure 25 Selection while learning

CHAPTER 12
CONCLUSION

Research introduces an innovative approach to user authentication and access control by leveraging Siamese neural networks and facial detection technology. The primary focus is on developing a system that seamlessly manages multiple user profiles on a shared device. By utilizing front camera-based facial detection, the proposed system ensures both security and user-friendly accessibility in personal and business environments.

The key components of the system, including negative and positive data files, real-time image capture, and Siamese neural networks, have been thoroughly outlined. Challenges related to operating system skills have been acknowledged, emphasizing the importance of overcoming these hurdles for successful implementation. The potential benefits of the system for organizations and users have been highlighted, emphasizing the elimination of traditional authentication methods like passwords or fingerprints.

Research contributes to the evolution of advanced user authentication systems, prioritizing security, convenience, and adaptability. The continuous improvement of the system's functionality is anticipated through user feedback and collaborative efforts with teammates and experts. The utilization of Siamese network features for authentication marks a significant stride towards enhancing user experience and ensuring robust security measures for shared computing environments. In summary, regulatory compliance requirements play a pivotal role in shaping how organizations adopt and use cloud computing services. Organizations must assess the regulatory landscape, carefully select cloud providers and services, implement robust security controls, and maintain ongoing compliance efforts to meet their legal obligations and protect sensitive data. Only using face camera for face unlocking feature is not a future goal but we have enhanced its functionality by adding Siamese network so it can also give us more than two login users in a same computer.

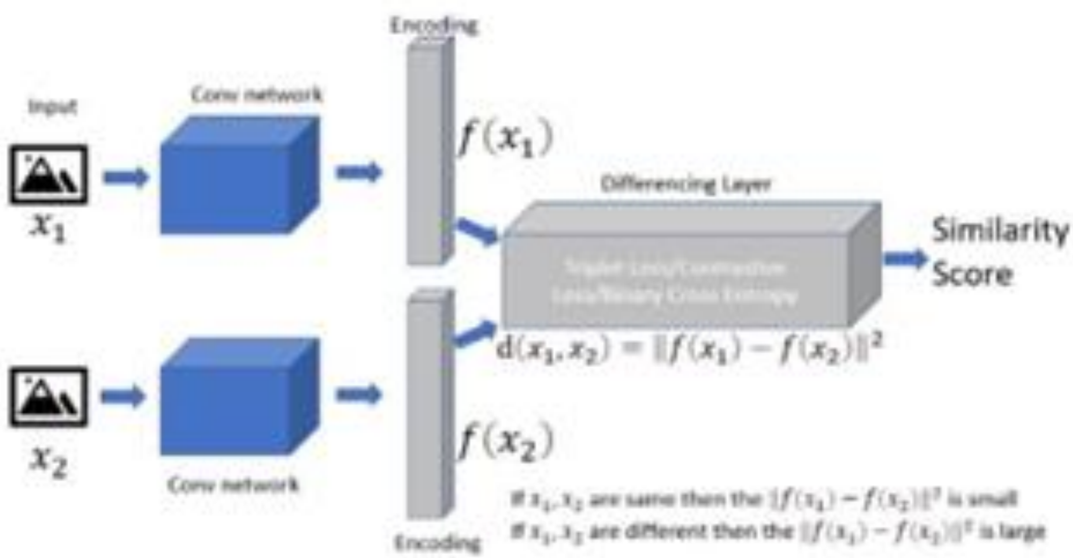

Figure 26 Similarity score calculation

12.1. **Results/Output**

```
In [4]:
# Setup paths
VER_PATH = os.path.join('data', 'vrification') # Verification image/ key ima
NEG_PATH = os.path.join('data', 'negative')
REA_PATH = os.path.join('data', 'realtime') # Real time input "Anchor"
```

```
In [ ]:
os.makedirs(VER_PATH)
os.makedirs(NEG_PATH)
os.makedirs(REA_PATH)
```

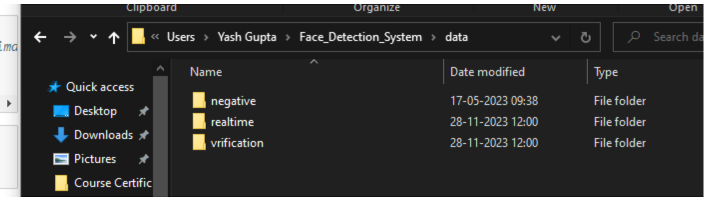

Figure 27 Data Path setup

Verification – Containing the positive set of data for model training.
Real Time – Containing the Real-Time data from camera source.
Negative – Containing data set that doesn't matches with the positive/verification data set.

```
In [ ]:
# "unzip" tgz file
!tar -xf lfw.tgz
```

```
In [5]:
# Moving images to NEG file
for directory in os.listdir('lfw'):
    for file in os.listdir(os.path.join('lfw', directory)):
        EX_PATH = os.path.join('lfw', directory, file)
        NEW_PATH = os.path.join(NEG_PATH, file)
        os.replace(EX_PATH, NEW_PATH)
```

```
In [6]:
for directory in os.listdir('lfw'):
    for file in os.listdir(os.path.join('lfw', directory)):
        print(os.path.join('lfw',directiory,file))
        print(os.path.join(NEG_PATH,file))
```

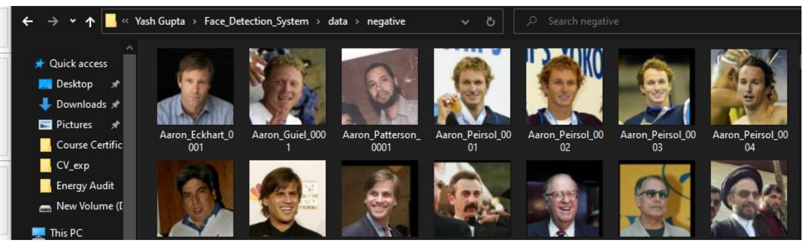

Figure 28 Unpacking Of negative data set from Kaggle

```
In [*]:
# Establish a connection to the webcam
cap = cv2.VideoCapture(0)

while cap.isOpened():  #reading each frame by frame
     ret, frame = cap.read()
# cap.read() reads the value from camera then unpacks it into
# "frame" - aka the required image and ret which gets a return value

    # Cut down frame to 250 x 250
    # as our negative data set also contains images of the 250x250
    # so we are also going to take input in 250x250 format to make it easier
    frame = frame[120:120+250,200:200+250, :]

    # Collect realtime
    if cv2.waitKey(1) & 0XFF == ord('r'):
        # Creating unique file path
        imgname = os.path.join(REA_PATH, '{}.
        # Write out realtime image
        cv2.imwrite(imgname, frame)

    # Collect positives
    if cv2.waitKey(1) & 0XFF == ord('p'):
        # Creating unique file path
        imgname = os.path.join(VER_PATH, '{}.
        # Write out positive image
        cv2.imwrite(imgname, frame)
```

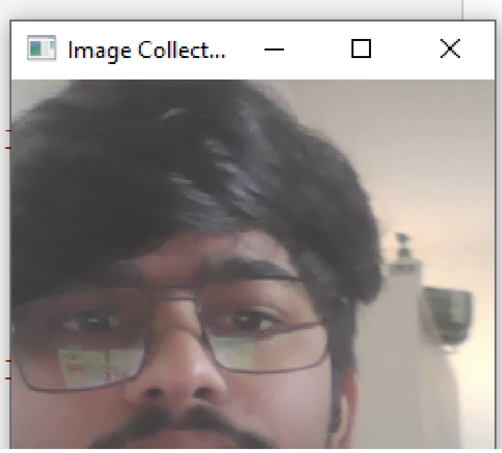

Figure 29 (a)

```
    # Breaking gracefully
    if cv2.waitKey(1) & 0XFF == ord('q'):
        break

## When camera is frozen
# Release the webcam
cap.release()
# Close the image show frame
cv2.destroyAllWindows()
```

Figure 29 (b) Instruction for taking input for Verification and Realtime data set testing

Capturing data for Verification and Realtime data set for testing and model training for implementation of Siamese Neural Network for One-shot Image Recognition.

The size of frame is set to 250x250 as the size of negative data set is of 250x250 format making it easy for implementation.

In [12]:

```
# REA_PATH +'\*.jpg' is doing a wild card search
# .trake is a limiter to how much in quantity
realtime = tf.data.Dataset.list_files(REA_PATH +'\*.jpg').take(250)
negative = tf.data.Dataset.list_files(NEG_PATH +'\*.jpg').take(350)
verify = tf.data.Dataset.list_files(VER_PATH +'\*.jpg').take(250)
```

Figure 30 Random Data Selection for Model Training

In [20]:

```
## Creating labeled Data set
# if (realtime,verifivation) => 1,1..
#    (realtime,negative) => 0,0..

verify = tf.data.Dataset.zip((realtime,verify,tf.data.Dataset.from_tensor_sl
negative = tf.data.Dataset.zip((realtime,negative,tf.data.Dataset.from_tenso
data = verify.concatenate(negative)
```

Figure 31 Creating Labelled Date Set

Labelled datasets are a crucial component of supervised learning algorithms, enabling them to learn patterns and make predictions effectively. Supervised learning algorithms rely on labelled data to establish a mapping between input data and desired output labels. By analysing the labelled data, the algorithm can discern the underlying patterns and relationships within the data, allowing it to make accurate predictions for new, unseen data instances.

The quality and quantity of the labelled data significantly impact the performance of supervised learning models. A well-labelled dataset provides the algorithm with a comprehensive understanding of the problem domain, leading to more accurate and robust predictions. However, if the labelled data is insufficient, inaccurate, or biased, the algorithm's performance can suffer.

In [24]:
```
def preprocessing_twin(input_img, validation_img, label):
    return(preprocessing(input_img), preprocessing(validation_img), label)
        #  realtime image              verification image or negative image
```

In [25]:
```
res = preprocessing_twin(*example)
# here res(real, negative/verification, 0 or 1)
```

In [26]:
```
plt.imshow(res[1])
# res[1] contains verification "but it could also be a negative"
```

Out[26]:
```
<matplotlib.image.AxesImage at 0x191cd7ce510>
```

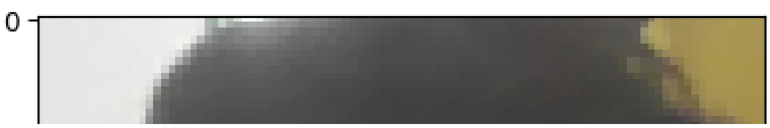

Figure 32(a) Building Train and Test

In [27]:
```
res[0]
# contains real time
```

Out[27]:
```
<tf.Tensor: shape=(100, 100, 3), dtype=float32, numpy=
array([[[0.4245098 , 0.41666666, 0.36960784],
        [0.41862744, 0.4107843 , 0.36372548],
        [0.40882352, 0.40098038, 0.35392156],
        ...,
        [0.41568628, 0.3882353 , 0.31960785],
        [0.41789216, 0.3875    , 0.33357844],
        [0.41862744, 0.3872549 , 0.3362745 ]],
```

Figure 32(b) Date from Real-Time Document Representation

In [28]:
```
res[2]
# got real + verification --> 1
# got real + negative --> 0
```

Out[28]:
```
1.0
```

Figure 32(c) Analysing Data

```
In [29]:
data = data.map(preprocessing_twin)
data = data.cache()
data = data.shuffle(buffer_size=1024)
```

```
In [30]:
data
# shape=(100,     100,       None)
# number of pixels by chanles in that image
# shape=(100, 100, None), shape=(100, 100, None), shape=()
#                                                 single unique value
```

Figure 33 Creating Data Pipeline

```
In [31]:
train_data = data.take(round(len(data)*.7)) # takes 70% of data split for tr
train_data = train_data.batch(16)           # processing/traning multiple im
train_data = train_data.prefetch(8)         # Feating the next set of data f
```

```
In [32]:
train_data
# shape=(     None,      100, 100, None)
#  number of images in the batch
```

```
Out[32]: <_PrefetchDataset element_spec=(TensorSpec(shape=(None, 100, 100, None), dtype=tf.float32, name=None), TensorSpec(shape=(None, 100, 100, None), dtype=tf.float32, name=None), TensorSpec(shape=(None,), dtype=tf.float32, name=None))>
```

Figure 34 Training Partition Data

```
In [33]:
test_data = data.skip(round(len(data)*.7))
test_data = test_data.take(round(len(data)*.3))
test_data = test_data.batch(16)
test_data = test_data.prefetch(8) #saving from bottelnecking our process
```

```
In [34]:
test_data
```

```
Out[34]: <_PrefetchDataset element_spec=(TensorSpec(shape=(None, 100, 100, None), dtype=tf.float32, name=None), TensorSpec(shape=(None, 100, 100, None), dtype=tf.float32, name=None), TensorSpec(shape=(None,), dtype=tf.float32, name=None))>
```

Figure 35 Testing Partition Data

```
In [35]: def make_embedding():
    inp = Input(shape=(100,100,3), name='input_image')

    #First Block
    c1 = Conv2D(64,(10,10), activation='relu')(inp)
    m1 = MaxPooling2D(64,(2,2),padding='same')(c1)

    #Second block
    c2 = Conv2D(128,(7,7), activation='relu')(m1)
    m2 = MaxPooling2D(64,(2,2),padding='same')(c2)

    #Third block
    c3 = Conv2D(128,(4,4), activation='relu')(m2)
    m3 = MaxPooling2D(64,(2,2),padding='same')(c3)

    #Final embedding block
    c4 = Conv2D(256,(4,4), activation='relu')(m3)
    f1 = Flatten()(c4)
    d1 = Dense(4096, activation = 'sigmoid')(f1)

    return Model(inputs=[inp], outputs=[d1], name='embedding')
```

Figure 36 Deploying Embedding Layer Frame of Siamese Neural Network

An embedding layer is a crucial component of a Siamese neural network, responsible for transforming raw input data into a lower-dimensional vector representation. This transformation is essential for the network to effectively compare and learn the similarities between different input pairs. For images, the embedding layer might extract features such as edges, shapes, and colors. For text sequences, it might capture the semantic meaning of individual words or phrases.

The embedding layer converts these raw input features into a dense vector representation, typically with a dimensionality much smaller than the original input. This reduction in dimensionality helps to alleviate the computational burden and prevent overfitting, particularly when dealing with large datasets.

```
In [37]:  1  embedding.summary()
```

```
Model: "embedding"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 imput_image (InputLayer)    [(None, 100, 100, 3)]     0

 conv2d (Conv2D)             (None, 91, 91, 64)        19264

 max_pooling2d (MaxPooling2D  (None, 46, 46, 64)       0
 )

 conv2d_1 (Conv2D)           (None, 40, 40, 128)       401536

 max_pooling2d_1 (MaxPooling  (None, 20, 20, 128)      0
 2D)

 conv2d_2 (Conv2D)           (None, 17, 17, 128)       262272

 max_pooling2d_2 (MaxPooling  (None, 9, 9, 128)        0
 2D)

 conv2d_3 (Conv2D)           (None, 6, 6, 256)         524544

 flatten (Flatten)           (None, 9216)              0

 dense (Dense)               (None, 4096)              37752832

=================================================================
Total params: 38,960,448
Trainable params: 38,960,448
Non-trainable params: 0
```

Figure 37 Embedding Layer Insite

```
In [40]: input_image = Input(name='input_img', shape=(100,100,3))
         validation_image = Input(name='validation_img', shape=(100,100,3))

In [41]: inp_embedding = embedding(input_image)
         val_embedding = embedding(validation_image)

In [42]: siamese_layer = L1Dist()

In [43]: distances = siamese_layer(inp_embedding, val_embedding)

In [44]: classifier = Dense(1, activation='sigmoid')(distances)

In [45]: classifier

Out[45]: <KerasTensor: shape=(None, 1) dtype=float32 (created by layer 'dense_1')>

In [46]: siamese_network = Model(inputs=[input_image, validation_image], outputs=clas
```

Figure 38 Development of Siamese Model

L1Distance layer determining the similarity between two input pairs. After the shared feature extractor has transformed the input data into vector representations, the distance layer calculates the distance between these representations. The choice of distance metric depends on the specific task and the nature of the input data.

Commonly used distance models are (Figure-17): -

Euclidean Distance
Manhattan Distance

```
In [47]:  1  siamese_network.summary()

Model: "SiameseNetwork"
__________________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to
==================================================================================================
 input_img (InputLayer)         [(None, 100, 100, 3  0           []
                                )]

 validation_img (InputLayer)    [(None, 100, 100, 3  0           []
                                )]

 embedding (Functional)         (None, 4096)         38960448    ['input_img[0]
[0]',
                                                                  'validation_i
mg[0][0]']

 l1_dist_1 (L1Dist)             (None, 4096)         0           ['embedding[0]
[0]',
                                                                  'embedding[1]
[0]']

 dense_1 (Dense)                (None, 1)            4097        ['l1_dist_1[0]
[0]']

==================================================================================================
Total params: 38,964,545
Trainable params: 38,964,545
Non-trainable params: 0
```

Figure 39 Model Data

Epoch refers to a single complete pass through the entire training dataset. During each epoch, the Siamese network is presented with all the training data pairs and updates its internal parameters to minimize the loss function. The number of epochs is a crucial hyperparameter that determines the extent of training and the overall performance of the network.

The purpose of multiple epochs is to allow the Siamese network to refine its ability to distinguish between similar and dissimilar input pairs. As the network processes more data, it gradually learns to identify and extract the most relevant features that contribute to the similarity assessment.

A sufficient number of epochs ensures that the network has adequately explored the training data and learned a robust representation of the relationships between input pairs. However, too many epochs can lead to overfitting, where the network memorizes the training data and fails to generalize well to unseen data.

```
In [61]:
def train(data, EPOCHS):
    # Loop through epochs
    for epoch in range(1, EPOCHS+1):
        print('\n Epoch {}/{}'.format(epoch, EPOCHS))
        progbar = tf.keras.utils.Progbar(len(data))

        # Loop through each batch
        for idx, batch in enumerate(data):
            # Run train step here
            train_step(batch)
            progbar.update(idx+1)

        # Save checkpoints
        if epoch % 10 == 0:
            checkpoint.save(file_prefix=checkpoint_prefix)
```

```
In [62]:
# Train The model
EPOCHS = 50
```

```
In [63]:
train(train_data, EPOCHS)
```

```
 Epoch 27/50
1/1 [==============================] - 6s 6s/step

 Epoch 28/50
1/1 [==============================] - 7s 7s/step

 Epoch 29/50
1/1 [==============================] - 7s 7s/step

 Epoch 30/50
1/1 [==============================] - 7s 7s/step

 Epoch 31/50
1/1 [==============================] - 7s 7s/step

 Epoch 32/50
1/1 [==============================] - 6s 6s/step

 Epoch 33/50
1/1 [==============================] - 6s 6s/step
```

Figure 40 Building Training Loop

```
In [73]:
siamese_model.save('siamesemodel.h5')
```

```
WARNING:tensorflow:Compiled the loaded model, but the compiled metrics have yet
to be built. `model.compile_metrics` will be empty until you train or evaluate
the model.
```

Figure 41 Siamese Model Developed

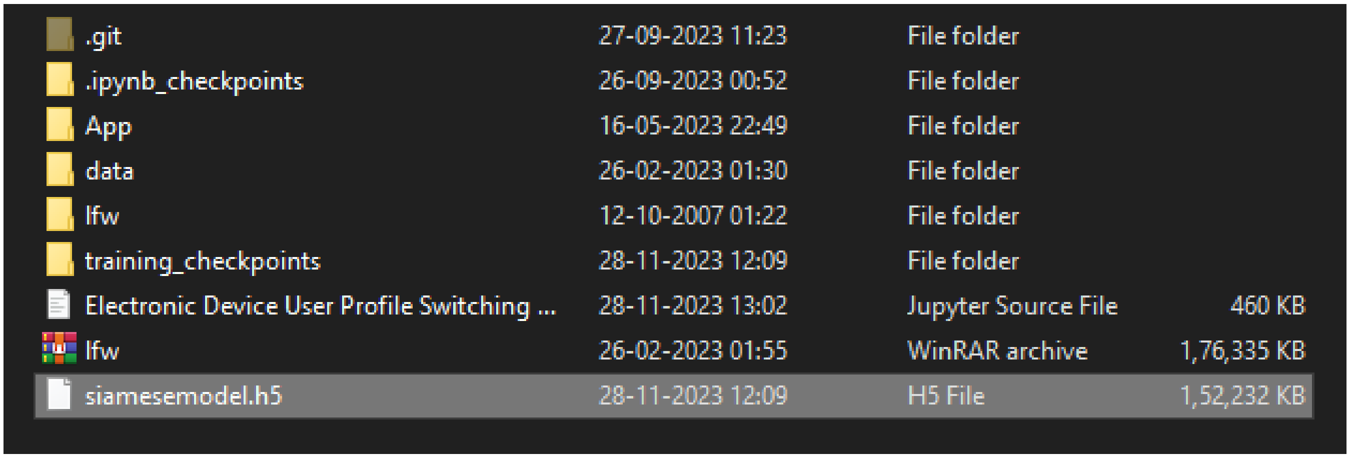

Figure 42 Siamese Model Ready for Execution

REFRENCES

[1] Siamese Neural Network for One-shot Image Recognition by Gregory Koch GKOCH@CS.TORONTO.EDU Richard Zemel ZEMEL@CS.TORONTO.EDU Ruslan Salakhutdinov RSALAKHU@CS.TORONTO.EDU.

[2] Bengio, Yoshua. Learning deep architectures for ai. Foundations and Trends in Machine Learning, 2(1):1–127, 2009.

[3] Bromley, Jane, Bentz, James W, Bottou, Leon, Guyon, ´ Isabelle, LeCun, Yann, Moore, Cliff, Sackinger, Ed- ¨ uard, and Shah, Roopak. Signature verification using a siamese time delay neural network. International Journal of Pattern Recognition and Artificial Intelligence, 7 (04):669–688, 1993.

[4] Chopra, Sumit, Hadsell, Raia, and LeCun, Yann. Learning a similarity metric discriminatively, with application to face verification. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 1, pp. 539– 546. IEEE, 2005.

[5] Fe-Fei, Li, Fergus, Robert, and Perona, Pietro. A bayesian approach to unsupervised one-shot learning of object categories. In Computer Vision, 2003.

[6] Proceedings. Ninth IEEE International Conference on, pp. 1134– 1141. IEEE, 2003. Fei-Fei, Li, Fergus, Robert, and Perona, Pietro. One-shot learning of object categories. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 28(4):594– 611, 2006.

[7] Hinton, Geoffrey, Osindero, Simon, and Teh, YeeWhye. A fast learning algorithm for deep belief nets. Neural computation, 18(7):1527–1554, 2006.

[8] Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pp. 1097–1105, 2012.

[9] Lake, Brenden M, Salakhutdinov, Ruslan, Gross, Jason, and Tenenbaum, Joshua B. One shot learning of simple visual concepts. In Proceedings of the 33rd Annual Conference of the Cognitive Science Society, volume 172, 2011.

[10] Lake, Brenden M, Salakhutdinov, Ruslan, and Tenenbaum, Joshua B. Concept learning as motor program induction: A large-scale empirical study. In Proceedings of the 34th Annual Conference of the Cognitive Science Society, pp. 659–664, 2012.

[11] Lake, Brenden M, Salakhutdinov, Ruslan R, and Tenenbaum, Josh. One-shot learning by inverting a compositional causal process. In Advances in neural information processing systems, pp. 2526–2534, 2013.

[12] Lake, Brenden M, Lee, Chia-ying, Glass, James R, and Tenenbaum, Joshua B. One-shot learning of generative speech concepts. Cognitive Science Society, 2014.

[13] Lim, Joseph Jaewhan. Transfer learning by borrowing examples for multiclass object detection. Master's thesis, Massachusetts Institute of Technology, 2012.

[14] Maas, Andrew and Kemp, Charles. One-shot learning with bayesian networks. Cognitive Science Society, 2009. Mnih, Volodymyr. Cudamat: a cudabased matrix class for python. 2009.

[15] Palatucci, Mark, Pomerleau, Dean, Hinton, Geoffrey E, and Mitchell, Tom M. Zero-shot learning with semantic output codes. In Advances in neural information processing systems, pp. 1410–1418, 2009.

[16] Simonyan, Karen and Zisserman, Andrew. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[17] Srivastava, Nitish. Improving neural networks with dropout. Master's thesis, University of Toronto, 2013.

[18] Taigman, Yaniv, Yang, Ming, Ranzato, Marc'Aurelio, and Wolf, Lior. Deepface: Closing the gap to human-level performance in face verification. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 1701–1708. IEEE, 2014.

[19] Wu, Di, Zhu, Fan, and Shao, Ling. One shot learning gesture recognition from rgbd images. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on, pp. 7–12. IEEE, 2012.

[20] Analitic Vidya Fundamentals of Deep Learning, Activation Function Dishasree26 Gupta Aug,2023.

[21] Medium based Understanding of Convolution Neural Network – Deep Learning Prabhu Mar,2018.

Final_Report_Elect_Swtching_Kulvinder_Sir_Plag

ORIGINALITY REPORT

14%	11%	8%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.cs.cmu.edu Internet Source	3%
2	Submitted to Jaypee University of Information Technology Student Paper	2%
3	www.pyimagesearch.com Internet Source	1%
4	www.analyticsvidhya.com Internet Source	1%
5	Abdelaziz Testas. "Distributed Machine Learning with PySpark", Springer Science and Business Media LLC, 2023 Publication	<1%
6	www.slideshare.net Internet Source	<1%
7	Donghoon Baek, Juhwan Seo, Joonhwan Kim, Dong-Soo Kwon. "Hysteresis Compensator with Learning-Based Hybrid Joint Angle Estimation for Flexible Surgery Robots", IEEE Robotics and Automation Letters, 2020	<1%