IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 5, May 2023

DOI: 10.17148/IJARCCE.2023.125118

Livenessnet with Hardware Interface for higher Security

Dr. Anil Kumar D[1], Amulya C S[2], Harshitha R[3] , Vinutha P[4], Sujay N[5]

Associate Professor, Electronics and Communications, BMS Institute of Technology and Management, Banglore, India[1]

Electronics and Communications, BMS Institute of Technology and Management, Banglore, India[2-5]

Abstract: This paper introduces a user-friendly face recognition system that includes liveness detection. The system's core programming is implemented in Python, which provides access to powerful machine learning libraries such as TensorFlow and Keras. These libraries are known for their high-level neural network capabilities, which greatly assist in training the dataset. In terms of hardware interfacing, a basic setup is employed to demonstrate the functionality of the liveness system. This setup incorporates a mechanism that automatically opens and closes a door when a recognizable face is detected. The face recognition system outlined in this paper aims to provide a straightforward and accessible solution for identifying individuals. By integrating liveness detection, it enhances security by verifying that the detected face is not a still image or a video playback. The choice of Python programming language for the core implementation offers several advantages. Python is widely used for machine learning tasks due to its simplicity and extensive libraries. TensorFlow and Keras, in particular, provide a comprehensive set of tools for neural network-based tasks, including facial recognition. Overall, this paper presents a practical and effective approach to face recognition, combining the power of Python and machine learning libraries to achieve accurate and reliable results while incorporating liveness detection for enhanced security applications.

Keywords— liveness, neural networks, Biometrics, training, Python, ESP controller.

I. INTRODUCTION

Face recognition is a dynamic field within computer vision and machine learning, playing a pivotal role in practical applications like banking, forensic science, and automatic attendance. With the advancement of digital technologies, there has been a surge in the demand for robust access control systems that prioritize security. Various authentication techniques, including encrypted usernames and passwords, smart cards, and biometrics, are employed to safeguard information. Biometrics, which encompass bodily characteristics such as iris patterns, palm shape, fingerprints, speech, and face, are an integral part of this new approach. Each biometric trait possesses unique qualities that enable distinct identification under specific circumstances. In this context, a face recognition method based on deep neural networks is proposed, aiming to achieve more accurate results by analyzing human face detection comprehensively.

Furthermore, recognizing a face as a distinctive object class identification event is crucial. This involves identifying the location, size, and spacing of specific facial features like the eyes, nose, and mouth. The positioning, size, and spacing of these features are meticulously examined. The primary focus of this object detection methodology is frontal face detection. By employing deep neural networks and comprehensive analysis of facial features, the proposed approach enhances the reliability of face recognition. This advancement addresses the growing need for secure access control systems and extends the application of face recognition to various domains such as banking, forensics, and attendance tracking. As the field continues to evolve, face recognition remains at the forefront of technological advancements in computer vision and machine learning. [1].

II. RELATED WORKS

Over the years, there has been an increasing demand for advancements in face recognition systems, and extensive research has been conducted to understand their evolution. One particular study [1] focuses on the use of traditional systems that relied on infrared rays for pattern recognition to define facial features. This approach, which originated in the early stages of face recognition, involved identifying facial characteristics based on detected patterns. Another research effort [2] delves into identifying the critical features necessary for achieving accurate recognition, while [3] presents its findings based on a dataset specifically designed for face recognition. Additionally, [4] explores the potential of a neural network-based approach to enhance the capabilities of the system.However, these systems face a significant

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 5, May 2023
DOI: 10.17148/IJARCCE.2023.125118

limitation due to their heavy reliance on lighting conditions and facial styling for effective recognition. Consequently, training the system to handle various inputs from the same individual becomes crucial, as evidenced in [5, 7]. The COVID-19 pandemic has prompted a widespread shift to online workspaces and education, necessitating the development of systems capable of detecting the liveness of individuals. This requirement becomes particularly crucial when monitoring is necessary for online assessments. To address this need, we propose the integration of liveness detection into the face recognition system.By incorporating liveness detection, our system aims to ensure that only live face samples are used for enrollment or authentication purposes. This technology examines physiological indicators of life in the face, enabling the differentiation between genuine faces and fraudulent or manipulated ones. In our proposed approach. This integration of liveness detection will enhance the reliability and security of face recognition, making it applicable in various domains, especially in the context of online assessments that require rigorous monitoring.

III. METHODOLOGY

In order to mitigate these concerns, the system has been augmented with live detection capabilities. By incorporating live detection technology, the system examines the face for physiological indicators of life to ensure that only live face samples are retained for enrollment or authentication purposes. In our proposed system, we will address the detection of brightness as a binary classification problem. Specifically, we will train a convolutional neural network to differentiate between genuine faces and fraudulent or faked ones using the input image as the basis for analysis. Figure 1 below shows the block diagram of the system[1].

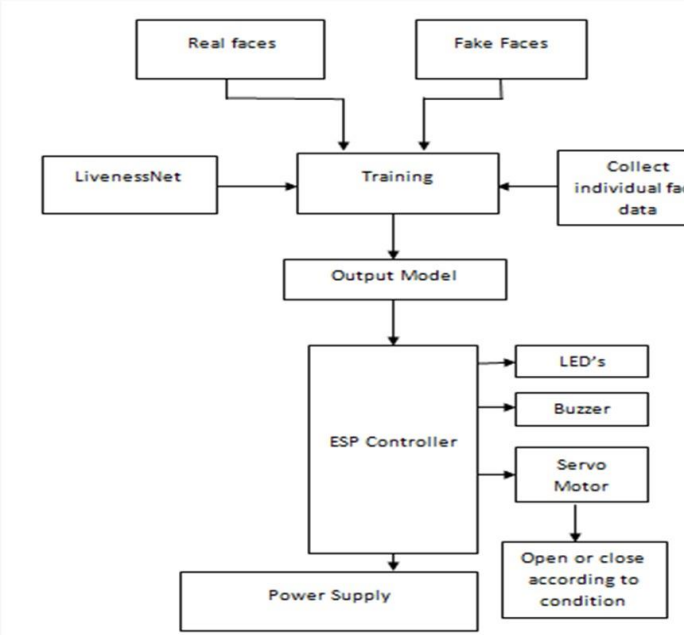


Figure 1. Block diagram of the system

A. Steps involved in training the system.

- **Creation/Collecting Of Dataset**: Gathering and assembling a set of data for analysis or study.
- **Face Embedding**: Transforming facial features into a numerical representation for analysis.
- T**raining The Face Recognizer**: Teaching the facial recognition system to identify specific individuals.
- **Recognize The Face**: Identifying and matching a face to a known individual.
- **Collection Of Real Data Set**: Gathering information from an authentic dataset for analysis.
- **Collect Of Fake Dataset**: Gathering a dataset comprising fabricated or synthetic information.
- **Training Livnessnet** (Fake Or Real Model File): Educating a model to distinguish between real and fake inputs.
- **Prediction Output**: The result or outcome produced by the system's analysis or inference.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 5, May 2023

DOI: 10.17148/IJARCCE.2023.125118

B. Architecture

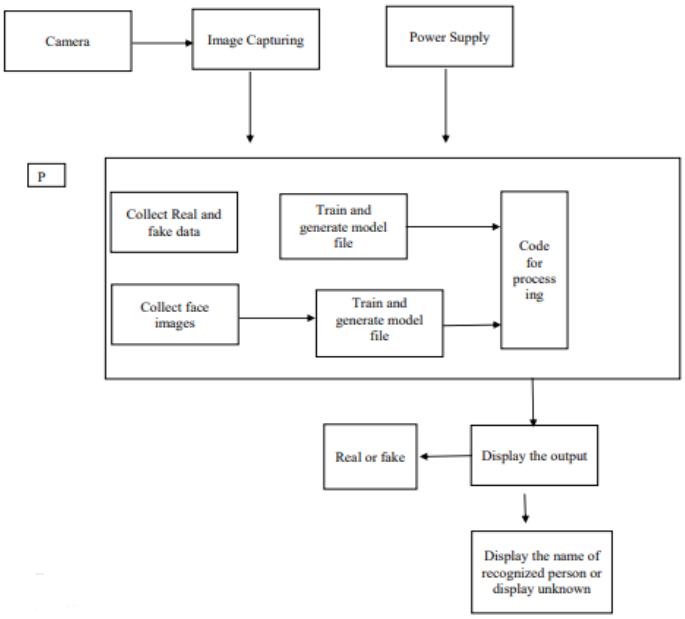


Figure 2. Architecture of system layers

C. Algorithm

Convolution Neural Networks (CNN)

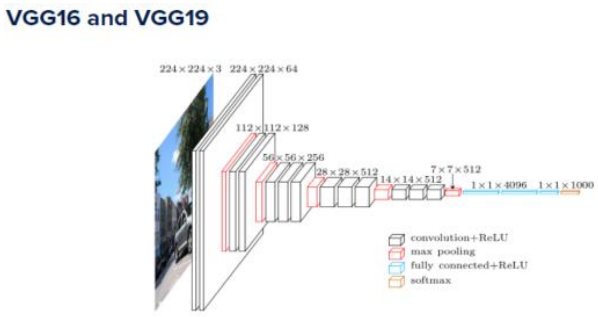


Figure 3: Visualization of the VGG architecture

In a convolutional neural network (CNN), each layer applies a distinct set of filters, typically comprising hundreds or even thousands of filters. These filters' outcomes are combined and passed on as input to the subsequent layer in the network. During the training process, the CNN automatically learns the optimal values for these filters. In the specific context of image classification, our CNN has the potential to learn the following tasks:

- In the first layer, it can identify edges by analyzing the raw pixel data of the input image.
- Building upon these detected edges, the second layer can identify shapes or "blobs."
- The higher layers of the network can leverage these identified shapes to detect more complex features, such as facial structures or specific parts of objects like cars.

Ultimately, the last layer of the CNN utilizes these higher-level features to generate predictions concerning the contents of the input image. In the realm of deep learning, an image convolution operation involves an element-wise

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 5, May 2023
DOI: 10.17148/IJARCCE.2023.125118

multiplication of two matrices, followed by the summation of the resulting elements. The process can be summarized as follows:

- Take two matrices, both having the same dimensions.
- Perform element-wise multiplication between the corresponding elements of these matrices (not to be confused with the dot product).
- Sum the resulting elements together to obtain the output of the convolution operation.

IV. WORKING

A. Software requirement

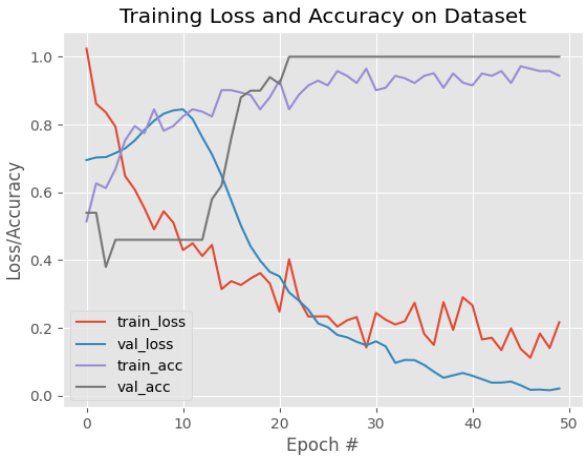
- Keras
- Tensorflow
- Tkinter
- Imutils
- Numpy
- OpenCV

The Python code encompasses all the necessary layers for activation and execution, ensuring smooth operation. During the recognition process, the code runs for an epoch of 50 iterations to improve accuracy and reliability. To optimize the system's efficiency, the epoch value can be modified as per the requirements within the source code. However, it is important to note that adjusting the epoch value may result in longer execution times and increased storage requirements due to the extended computational operations involved. Careful consideration should be given to balancing efficiency with performance when making such adjustments.

The layer types involved are as follows

- Convolutional (CONV): A mathematical operation that applies filters to input data for feature extraction.
- Activation (ACT or RELU): A function that introduces non-linearity to the network, commonly ReLU (Rectified Linear Unit).
- Pooling (POOL): Downsampling technique that reduces spatial dimensions and extracts key features.
- Fully-connected (FC): A layer in a neural network where each neuron is connected to every neuron in the previous layer.
- Batch normalization (BN): A technique that normalizes the input data within a batch to improve network training.
- Dropout (DO): A regularization method that randomly drops out a fraction of neurons during training to prevent overfitting.

Once the code is executed, the application window opens and the epochs and efficiency graphs can be seen.



IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 5, May 2023

DOI: 10.17148/IJARCCE.2023.125118

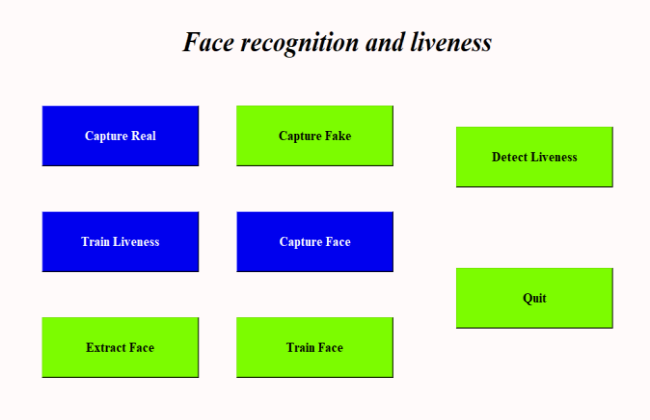


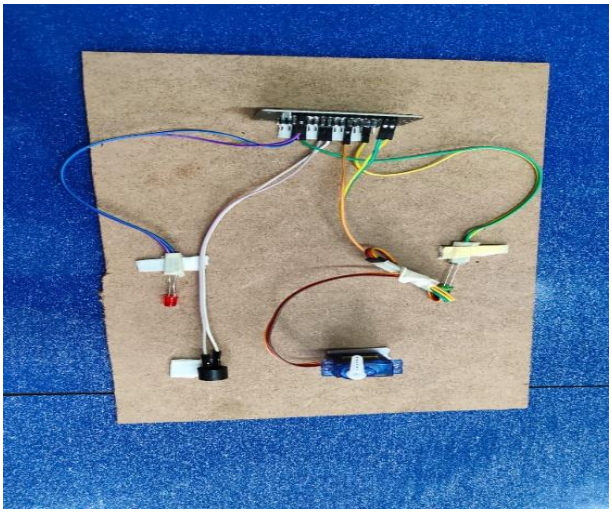
Figure 4. The application window and the training plot

Training a face recognition system involves several steps to ensure its accuracy and reliability. First, a dataset of faces needs to be prepared and stored in the "dataset" folder of the code. This dataset should include both real and fake faces to make the system more robust against spoofing attacks. Collecting a diverse dataset with a variety of poses, lighting conditions, and backgrounds is crucial for training a robust face recognition system.

To detect the liveness of a face, a hardware component is used to trigger the opening and closing of a door in a basic demo. The liveness detection system uses a combination of deep learning techniques and computer vision to distinguish between real and fake faces. The training process involves utilizing face embedding techniques to extract features from the faces in the dataset. These features are used to create a numerical representation of each face, which can be compared to other faces to determine if they belong to the same person or not.

The training process involves optimizing the parameters of the neural network by minimizing the loss function between the predicted and actual outputs. Several techniques, such as batch normalization and dropout, can be used to prevent overfitting and improve the generalization capability of the model.

Overall, by using a combination of face recognition and liveness detection, this system can help to improve security in a variety of settings, such as access control for buildings or devices. By accurately identifying individuals and preventing spoofing attacks, this system can provide a high level of security and convenience for various applications.



The hardware is connected to Arduino for interfacing purposes. However, it offers flexibility as different hardware implementations are possible. As long as the Arduino program is executed, it defines the board's operations, enabling various hardware configurations to be utilized.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 5, May 2023
DOI: 10.17148/IJARCCE.2023.125118

V. FUTURE WORK

As we progress into the future, there will be an increasing availability of training datasets collected from various settings and circumstances. These datasets will encompass images or frames showcasing faces of diverse skin tones, sizes, and shapes, contributing to a more inclusive and representative training process. It is important to note that our model has been trained using a fabricated dataset comprised exclusively of photos captured from mobile phones, tablets, and other devices, rather than relying on printed images or pictures. Consequently, when constructing a fake dataset, it is essential to incorporate different types of data to enhance the model's ability to recognize and distinguish real from fake faces. The potential applications of this work extend beyond face recognition alone. It could be further developed into an app that facilitates the monitoring of students during tests and exams conducted online. By employing the face recognition system, the app can ensure the authenticity and integrity of the examination process by verifying the identity of the students and detecting any potential fraudulent activities. This can contribute to maintaining academic integrity and fairness in remote educational environments.

VI. CONCLUSION

The suggested approach for person detection is capable of identifying faces with a high degree of accuracy and robustness. In addition to its face recognition capabilities, the system incorporates a liveness detector that utilizes a CNN-based model called "Liveness Net" to differentiate between phone-generated and real faces. The initial step in developing the suggested liveness detector facial recognition model involves collecting data specifically for training the Liveness Net model. This is achieved by holding a smartphone in front of the camera while capturing footage. The genuine dataset used for liveness detection is derived directly from the camera's recorded footage, making it similar to the dataset used for face recognition. Once the dataset has been prepared, the Liveness Net model is employed to initially determine the liveness of each frame. This deep learning model, trained on the dataset, effectively distinguishes between real and phone-generated faces. Subsequently, the discovered real face is recognized using the face recognition model, which has been developed using the same training dataset.By combining the liveness detection functionality of the Liveness Net model with the face recognition capabilities of the deep learning model, the system can accurately identify and differentiate between genuine faces and phone-generated faces, ensuring enhanced security and reliability in various applications. [1]

REFERENCES

[1] Samana Jafri, Satish Chawan, Afifa Khan, “Face Recognition using Deep Neural Network with “LivenessNet”,” *Proceedings of the Fifth International Conference on Inventive Computation Technologies, 2020.*
[2] R.S.Ghiass, O. Arandjelović, H. Bendada, and X. Maldague, “Infrared face recognition: A comprehensive review of methodologies and databases,” *A Treatise on Electricity and Magnetism*, 3rd edition, vol. 2, pp.68–73,2014.
[3] N. Abudarham, L. Shkiller, and G. Yovel, “Critical features for face recognition*,”Cognition*, vol. 10, pp. 73-83, 2019.
[4] I. Masi, A. T. Tran, T. Hassner, G. Sahin, G. Medioni, “Face-specific data augmentation for unconstrained face recognition*,” International Journal of Computer Vision (IJCV*), vol. 127, pp. 642-660, 2019.
[5] C. Huang, Y. Li, C. C. Loy, “Deep imbalanced learning for face recognition and attribute prediction,” *IEEE transactions on pattern analysis and machine intelligence*, pp. 1-14, 2019.
[6] J. Deng, D. Zhang, Y. Deng, J. Guo, “Lightweight face recognition challenge," *Proceedings of the IEEE International Conference on Computer Vision Workshops*, vol. 10, pp. 1-10, 2019.
[7] S. L. Happy, A. Dasgupta, A. George and A. Routray, “A Video Database of Human Faces under Near Infra-Red Illumination for Human Computer Interaction Applications,” *IEEE Proceedings of 4th International Conference on Intelligent Human Computer Interaction*, Kharagpur, India, 2012.
[8] K. Shajahan, R. D. Rai, Ravishankara, “Face Recognition Using Livenessnet,” *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 7, no. 4, pp. 143-148, 2021.
[9] M. Sharma, J. Anuradha, H. K. Manne, G. S. C. Kashyap, “Facial detection using deep learning,” IOP Conference series Material and Engineering journal, vol. 10, pp. 1-10, 2017.
[10] Y. Singh, R. Toldo, L. Magri, S. Fantoni, A. Fusiello, “Method for 3D modelling based on structure from motion processing of sparse 2D images,” *United States Patent*, vol. 10, pp. 1-24, 2019.
[11] A. Bijoura, S. Banerjee, K. S. Pandey, Dr A. Arivoli, “Facial recognition using deep learning,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 07, no. 08, pp. 2719-2726, 2020