**Face Verification and Recognition for digital forensics and information security**

Chapter 3

Project specification:

3.1 **Aims of Face Recognition and Verification for Digital Forensics and Information Security:**

The primary aim of face recognition and verification for digital forensics and information security is to make provision for a means of identifying individuals. This is particularly important in cases where the identities of individuals are needed to be verified, such as in criminal investigations, border control, or access control to secure facilities.

Face recognition and verification can also be used to prevent identity theft and fraud. With the increasing use of digital technologies for financial transactions, identity theft and fraud have become significant concerns. By using face recognition and verification, financial institutions can prevent fraudulent transactions.

Another important aim of face recognition and verification for digital forensics and information security is to enhance the security of digital information. By using face recognition and verification, access to sensitive digital information can be restricted to authorized individuals only, thereby reducing the risk of unauthorized access and data breaches.

3.1.1 **Objectives of Face Recognition and Verification for Digital Forensics and Information Security:**

To achieve the aims of face recognition and verification for digital forensics and information security, several objectives need to be met. These objectives include:

1. Developing accurate and reliable face recognition and verification algorithms:

The accuracy and reliability of face recognition and verification algorithms are critical to their effectiveness in digital forensics and information security. Algorithms must be designed to handle variations in lighting, facial expressions, and other factors that can affect the accuracy of facial recognition.

2. Ensuring the security and privacy of facial data:

Facial data used for recognition and verification must be protected from unauthorized access and use. This requires the implementation of appropriate security measures, such as encryption and access controls.

3. Integrating face recognition and verification with other security measures:

Face recognition and verification should be integrated with other security measures, such as access controls and biometric authentication, to provide a more comprehensive security solution.

4. Training and educating users:

To ensure the effective use of face recognition and verification for digital forensics and information security, users must be trained and educated on how to use the technology properly.

3.2 **Functional and non-functional requirements**

3.2.1 Functional Requirements:

The system must be able to capture and store images of faces.

The system must be able to compare the captured face images against a database of known faces.

The system must be able to identify the person in the captured face image if they exist in the database.

The system must be able to notify an administrator or security team if an unknown person is detected.

The system must be able to track and record all instances of face verification and recognition.

3.2.2 Non-functional requirements:

Accuracy: The system should be able to accurately recognize or verify faces, even in difficult lighting conditions or with partial obstructions.

Speed: The system should be able to perform recognition or verification in real-time or near real-time.

Security: The system should be secure and protect the privacy of individuals being verified or recognized.

Scalability: The system should be able to handle a large numbers of faces in the database and a high volume of verification and recognition requests.

Reliability: The system should be available and reliable, with minimal downtime or errors.

Usability: The system should be user-friendly and easy to use for authorized personnel.

3.3 **Methodology**

Introduction

Face verification and recognition are important areas of research in digital world of today. In recent years, the field has seen tremendous growth, with significant advancements in both the algorithms and the data acquisition techniques. The ability to identify individuals from facial images can be used for a range of applications, from law enforcement to surveillance and access control systems. Therefore, in this report we provide a detailed methodology for face verification and recognition, including data acquisition, pre-processing, feature extraction and evaluation.

3.3.1 Data Acquisition:

The first step in the methodology is data acquisition. A realistic data acquisition process is essential to collect a dataset that corresponds to a typical face verification setup. The dataset should include a large number of face images belonging to different individuals, and it should be diverse in terms of lighting conditions, poses, and expressions. The dataset can be acquired in different ways, including using CCTV cameras, smartphone cameras, or dedicated cameras. The collected data should be carefully annotated, with each face image associated with a label indicating the identity of the individual.

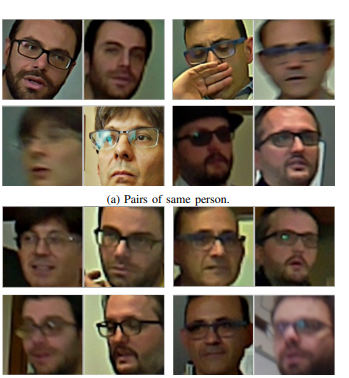


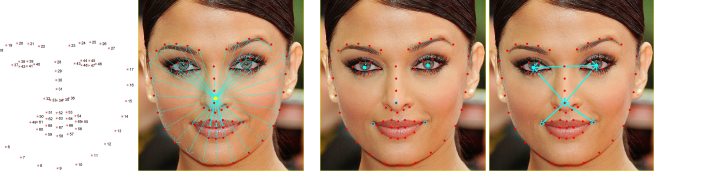
Fig 3.3.1 face image belonging to different individuals

3.3.2 Pre-processing:

Once the dataset is collected, the next step is pre-processing. The goal of this phase is to enhance the quality of the images and to remove any noise or artefacts that may interfere with the recognition process. The techniques may also include image resizing and normalization. For example, normalization can be used to standardize the brightness and contrast of the images, which can improve the accuracy of the recognition system.

3.3.3 Feature Extraction:

After pre-processing, the next step is feature extraction. This is the process of identifying the unique features of each face image that can be used to distinguish it from others. Several feature extraction techniques have been proposed in the literature, including methods based on facial landmarks, geometric features, and texture features. In recent years, deep learning approaches based on Convolutional Neural Networks (CNNs) have shown great promise in feature extraction for face recognition and verification. CNN-based feature extraction methods have been shown to outperform traditional feature extraction techniques, as they can automatically learn discriminative features from the input images.



1. **Facial landmarks**: Facial landmarks are essential points located on the face that can be utilized as features for various tasks, such as enhancing face recognition, aligning facial images, estimating head pose, and differentiating between genders. Certain points, such as those around the eyes, nose, and mouth, are more representative of a person's face and are known as nodal points. These nodal points, along with other points computed from facial landmarks, can provide more relevant information to represent a face. We utilized the dlib library to extract facial landmarks from an image, which returns an array of 68 (x, y) coordinates that map to facial structures. We then computed three different features based on the distances between nodal points and facial landmarks, which were normalized to the bounding box's size. Specifically, each distance was divided by the bounding box's diagonal.
2. **Convolution neural network:** Convolutional Neural Networks (CNNs) have become popular for classification tasks and are also useful in face verification and recognition in security and surveillance applications. CNNs can detect relevant features in input images and extract increasingly complex representations as the network becomes deeper. The output of the last layer before the network's output is a high-level representation of the input image, which we refer to as a deep feature in this paper. This representation can be compared to other deep features computed on other faces, and if the distance between two deep feature vectors is below a certain threshold, we can conclude that the two faces belong to the same person. We use the ResNet-50 CNN, which is pre-trained on the MS-Celeb-1M dataset of 10 million images of 100 thousand different identities and fine-tuned on the VGGFace2 dataset of 3.31 million images of 9,131 different identities. We take the output of the pool5|7x7 s1 layer as the deep feature, which is a 2,048 size float vector.

3.3.4 Matching:

Once the features are extracted, the next step is matching. Matching involves comparing the features of two face images to determine if they belong to the same individual or not. Several matching algorithms have been proposed in the literature, including Euclidean distance, Mahalanobis distance, and cosine similarity. Deep learning approaches based on Siamese Networks and Triplet Networks have also shown great promise in face verification and recognition. Siamese Networks and Triplet Networks can be used to learn a similarity metric between pairs or triplets of face images, respectively.

The learned metric can then be used to determine whether two face images belong to the same individual or not.

3.3.5 Training and Fine-tuning:

In order to achieve high accuracy in face recognition and verification, the deep learning models need to be trained on large and diverse datasets. Several pre-trained CNN models are available in the literature, which can be fine-tuned for the task of face verification and recognition. Fine-tuning involves training the pre-trained model on a smaller dataset of face images for the specific task of face verification and recognition. In addition, data augmentation techniques can be used to increase the size of the training dataset and improve the generalizability of the model.

3.3.6 Deployment:

The final step in the methodology is deployment. Once the face verification and recognition model is trained, it can be deployed in a variety of applications, including access control systems, surveillance systems, and law enforcement systems. The deployment of the model requires careful consideration of ethical and legal implications, particularly with regard to privacy and security.

3.3.7 Challenges and Limitations:

Despite the significant progress made in face verification and recognition, several challenges and limitations still exist. One of the primary challenges is the lack of diversity in the training datasets, which can lead to bias and reduced performance on faces of underrepresented groups. Another challenge is the robustness of the models to various types of attacks, including spoofing attacks, using printed or digital images. In addition, the deployment of face verification and recognition systems raises concerns about privacy and security, particularly with regard to the potential misuse of the technology for surveillance and profiling.

3.3.8 Conclusion:

In conclusion, face verification and recognition have become important areas of research in digital forensics and information security. The methodology of face verification and recognition involves several steps, including data acquisition, preprocessing, feature extraction, matching, training, and deployment. Several techniques and algorithms have been proposed in the literature, including methods based on facial landmarks, geometric features, texture features, and deep learning approaches based on CNNs, Siamese Networks, and Triplet Networks. The performance of the system should be evaluated using several metrics, and the limitations and challenges of the system should be identified and discussed. The results of this research have significant implications for law enforcement, national security, and privacy protection. Further research is needed to address the challenges and limitations of face verification and recognition, particularly with regard to diversity, robustness, and ethical and legal considerations.

3.4 Brief explanation of project’s plan

The objective of this project is to develop a system for face recognition and valuation using Support Vector Machines (SVM) and FaceNet, and to explore its potential applications in digital forensics and information security.

The first step in this project would be to collect a large dataset of face images, which would be used to train the FaceNet model. FaceNet is a deep learning model that learns to encode face images into a high-dimensional feature space. This feature space is designed such that faces of the same person are closer together, while faces of different people are further apart.

Once the FaceNet model is trained, we would use it to extract features from the face images in our dataset. These features would then be fed into an SVM classifier, which would learn to distinguish between different individuals based on their facial features.

To evaluate the performance of our system, we would use standard metrics such as accuracy, precision, and recall. We would also explore its potential applications in digital forensics and information security, such as identifying suspects in surveillance footage, verifying the identity of users in secure systems, or detecting unauthorized access attempts.

Overall, this project has the potential to contribute to the development of more accurate and reliable face recognition and valuation systems, which could have important implications for digital forensics and information security.

3.5 Legal, ethical, social, professional and environmental review

This section takes a look at legal, ethical, professional and environmental implication of implementing face recognition and verification.

a. **Legal issues:**

* Law and regulations may be violated if face recognition and verification techniques are used without proper consent or justification.
* Unauthorized use of someone's biometric information could lead to legal actions.
* Rules and regulations regarding the collection, storage and sharing of biometric data must be followed.

b. **Ethical issues:**

* Misuse of face recognition and verification technology can lead to invasion of privacy.
* Bias and discrimination can arise if the technology is not properly designed and tested.
* The use of face recognition and verification in sensitive areas such as law enforcement and security should be done with great care and transparency.

c**. Social Issues:**

* Face recognition and verification technology may lead to a loss of privacy and personal freedoms.
* The use of the technology may lead to social stigma or discrimination for those who are identified or falsely accused.

d. **Professional Issues:**

* Professionals using face recognition and verification technology should ensure that they have the appropriate knowledge, training, and experience to use the technology effectively and ethically.
* The accuracy of the technology must be taken into account when using it for digital forensics and information security purposes.

e. **Environmental Issues:**

* The use of face recognition and verification technology can require significant energy and computing resources, which can have negative environmental impacts.
* The disposal of equipment and materials used for the technology should be done in an environmentally responsible manner.

3.6 Identification of risks, including security risks and mitigating measures taken

In this section we will identify potential security risks associated with face recognition and verification systems in digital forensics and information security and discuss the mitigating measures taken to address them

3.6.1 Identification of risks

I**. False Positives**:

One of the significant risks of face recognition systems is the occurrence of false positives. This happens when the system incorrectly identifies a person as someone else, which can lead to errors in investigations or breaches in security.

ii. **Spoofing Attacks**:

Face recognition systems can be vulnerable to this kind of attack, where attackers use masks, photos, or videos to deceive the system into recognizing a fake identity. This can compromise the security of the system and allow unauthorized access.

iii**. Biometric Data Theft**

: Face recognition systems store biometric data, including facial patterns, which can be stolen or hacked. This poses a risk to the privacy and security of individuals and can result in identity theft or other fraudulent activities.

iv. **System Malfunction**:

Like any other system, face recognition and verification systems can experience hardware or software malfunctions, which can lead to errors in identification and compromise security.

3.6.2 Mitigating measures

i. **Multi-Factor Authentication**:

The use of multi-factor authentication, such as a combination of facial recognition and password or PIN, can reduce the risk of false positives and prevent unauthorized access.

ii**. Anti-spoofing Techniques**:

Implementing anti-spoofing techniques such as liveness detection, where the system verifies that the person is physically present and not a fake, can mitigate the risk of spoofing attacks.

iii**. Encryption and Access Controls**:

To protect biometric data, face recognition and verification systems should use encryption techniques and access controls to limit access to authorized personnel only.

iv. **Regular Maintenance and Testing**:

Regular maintenance and testing of the system can help prevent malfunctions and identify vulnerabilities before they can be exploited by attackers.

Face recognition and verification systems are valuable tools in digital forensics and information security. However, they pose risks to security and privacy that must be mitigated. By implementing multi-factor authentication, anti-spoofing techniques, encryption and access controls, and regular maintenance and testing, organizations can reduce the risks associated with these systems and ensure the security of their data and systems.

3.7 Consideration of applicable health & safety, diversity, inclusion and cultural matters

It is important to consider health and safety, diversity, inclusion, and cultural matters when implementing these systems to ensure that they are used properly and do not cause harm or discrimination. In this section, we will discuss seven considerations related to these issues.

i**. Health and Safety:**

When implementing face recognition and verification systems, it is important to ensure that they do not pose a risk to the health and safety of individuals. For example, facial recognition cameras should not emit harmful radiation, and their installation should not obstruct emergency exits or fire escape routes.

ii. **Diversity and Inclusion**:

Face recognition and verification systems should be designed to be inclusive and avoid discrimination. This means ensuring that the systems are accurate across diverse populations, including different races, ethnicities, ages, genders, and physical abilities. System developers should test their systems on diverse datasets to ensure that they are not biased towards a particular demographic.

iii. **Cultural Sensitivity:**

Cultural sensitivity is important when implementing face recognition and verification systems. For example, in some cultures, it may be inappropriate or offensive to be photographed without consent. Therefore, it is important to inform individuals about the use of the system and obtain their consent before capturing their biometric data.

iv. **Data Privacy**:

Data privacy is a critical consideration for face recognition and verification systems. Individuals have the right to know how their biometric data will be used, who will have access to it, and how it will be stored and protected. Organizations must ensure that their systems comply with relevant privacy laws and regulations, such as the General Data Protection Regulation (GDPR) in Europe.

v. **Ethical Use**:

The ethical use of face recognition and verification systems is also important. Organizations must consider how the systems will be used, who will have access to the data, and how the data will be used. The use of the system should be limited to legitimate purposes, such as security or law enforcement, and should not infringe on individuals' rights or freedoms.

vi. **Human Oversight:**

Human oversight is necessary for face recognition and verification systems. Human oversight can help to identify and correct errors, prevent abuse of the system, and ensure that the system is used in an ethical and responsible manner. Additionally, human oversight can help to address issues related to diversity, inclusion, and cultural sensitivity.

vii. **Education and Training**:

Education and training are critical for the responsible use of face recognition and verification systems. Organizations should educate individuals about the use of the system, their rights, and the potential risks and benefits. Additionally, organizations should provide training to staff responsible for using and maintaining the system to ensure that they are aware of best practices and ethical considerations.

the implementation of face recognition and verification systems in digital forensics and information security requires careful consideration of health and safety, diversity, inclusion, cultural sensitivity, data privacy, ethical use, human oversight, education, and training. By addressing these considerations, organizations can ensure that their systems are used responsibly and do not cause harm or discrimination

3.8 Identification of codes of practice and industry standards related to the work

**ISO/IEC 19795-1:2011** - Biometric performance testing and reporting - Part 1: Face recognition

This standard specifies methods for testing and reporting the performance of face recognition systems. It includes metrics for measuring accuracy, speed, and other aspects of performance.

**NIST Special Publication 800-63B** - Digital Identity Guidelines - Authentication and Lifecycle Management

This publication provides guidelines for authentication and lifecycle management of digital identities. It includes specific recommendations for face recognition and verification, such as the use of liveness detection to prevent spoofing attacks.

**IEC 62676-4:2014** - Video surveillance systems for use in security applications - Part 4: Application guidelines

This standard provides guidelines for the use of video surveillance systems in security applications. It includes recommendations for the use of face recognition technology in such systems.

**EN 16571:2014** - Biometrics - Harmonized vocabulary

This European standard provides a harmonized vocabulary for biometric terms, including those related to face recognition and verification.

**ANSI/NIST-ITL 1-2011** - Data Format for the Interchange of Fingerprint, Facial, & Scar Mark & Tattoo (SMT) Information

This standard defines a data format for the exchange of biometric information, including facial images, between different systems.

**CEN/TS 16082:2011** - Forensic science - Digital evidence - Guidelines for the management of facial images

This technical specification provides guidelines for the management of facial images as digital evidence. It includes recommendations for the acquisition, storage, and analysis of such images

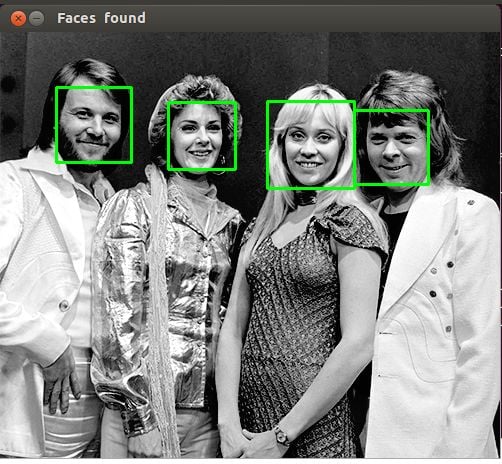
Chapter 4

Design alternative and justification for chosen design

4.1 Design alternative

**Traditional Face Recognition:**

Traditional face recognition systems use techniques such as Eigen faces, Fisher faces and Local Binary Patterns (LBP) to extract patterns from the face and match them against corresponding faces from the database. These systems have been used for many years and are widely adopted due to their accuracy. However, they can be vulnerable to attacks such as spoofing and replay attacks, where an attacker uses a fake image or video to trick the system..



**3D Face Recognition:**

3D face recognition systems use 3D models of the face to extract features and match them against corresponding faces from the database. These systems are more robust to changes in lighting and facial expressions and are less susceptible to attacks such as spoofing and replay attacks. However, they require specialized hardware such as 3D scanners or structured light cameras to capture the 3D model of the face.

**Deep Learning Face Recognition:**

Deep learning face recognition technology uses artificial neural networks to identify a person's face. This technology is based on training the neural network on a large dataset of images to recognize different faces. Deep learning face recognition technology is more accurate than traditional face recognition and can identify faces even when there are changes in lighting or facial expression.

**Thermal Face Recognition**

Thermal face recognition systems use infrared cameras to capture the heat emitted by the face and extract patterns for matching against corresponding faces from the database. These systems are less susceptible to changes in lighting and can work in complete darkness. However, they can be affected by changes in ambient temperature and require specialized hardware.

**Multi-Factor Face Verification**

Multi-factor face verification systems use a combination of face recognition and other biometric modalities such as fingerprints or voice recognition to authenticate individuals. These systems are more secure than single-factor systems and can provide a higher level of assurance. However, they can be more complex to implement and require specialized hardware and software.

In conclusion, designing an effective face recognition and verification system for digital forensics and information security requires careful consideration of the available design alternatives. Traditional face recognition systems are widely adopted and have a proven track record of accuracy and efficiency, while deep learning-based systems offer higher accuracy and robustness to attacks. 3D face recognition and thermal face recognition systems offer advantages in different lighting and environmental conditions. Multi-factor face verification systems offer a higher level of security but require more complex implementation. Ultimately, the choice of design alternative will depend on the specific needs and requirements of the organization.

4.2 Justification for the chosen design

One of the most widely used techniques for face recognition and verification is Support Vector Machines (SVM), which is a supervised learning algorithm that is used to classify and predict data. Another technology that has gained popularity in face recognition and verification is FaceNet, which is a deep learning model that uses a convolutional neural network (CNN) to map faces into a high-dimensional space. We will explore the justification for the combined use of SVM and FaceNet which are machine learning and deep learning algorithm

4.2.1 Support Vector Machines (SVMs):

SVM is a supervised learning algorithm that is widely used in pattern recognition and classification tasks. SVMs work by finding the optimal hyperplane that maximally separates the data into two classes. In face recognition and verification, SVMs are trained on face images to classify them into two categories, i.e., a positive class (the target face) and a negative class (other faces). SVMs use a kernel function to transform the input data into a high-dimensional space where the hyperplane is found. SVMs are known for their high accuracy and ability to handle high-dimensional data.

4.2.2 FaceNet:

FaceNet is a deep learning model that uses a convolutional neural network (CNN) to extract patterns from a face image and generates a high-dimensional embedding vector. FaceNet uses a triplet loss function to learn the embedding such that the distance between embedding of the same person is minimized, while the distance between embedding of different people is maximized. FaceNet has shown impressive performance in face recognition and verification tasks and has been widely used in various applications, including security systems and forensic investigations.

4.2.3 Combined use of SVM and FaceNet:

The combination of SVM and FaceNet can provide a robust and reliable approach for face recognition and verification in digital forensics and information security. SVMs can be used to classify face images into positive and negative classes, while FaceNet can be used to learn the high-dimensional embedding of the face images. The SVM classifier can then be trained on the FaceNet embedding to classify the faces accurately.

One of the key benefits of using SVM and FaceNet together is the ability to handle variations in lighting, pose, and other factors that can affect face recognition and verification. SVMs are known to be robust to noise and can handle complex data, while FaceNet can learn the unique features of each face and generate high-quality embedding that capture these features.

Overall, the choice of face recognition and verification choice will depend on the specific requirements of the application and the available resources.

Chapter 5

5.0 Implementation and testing

The implementation phase of the project is where design details and specifications are converted into a working and executable system. This section of the report documents methodology adopted, the implemented functions. After each phase was implemented, it was tested and corrected to fix any required issues or bugs.

5.1 Implementation methodology

The proposed system consists of two main stages: face recognition and verification. In the first stage, we use the Facenet model to extract deep features from the face images. Facenet is a state-of-the-art deep learning model that is capable of mapping faces to a high-dimensional feature space. The extracted features are then fed into an SVM classifier, which is trained on a dataset of face images.

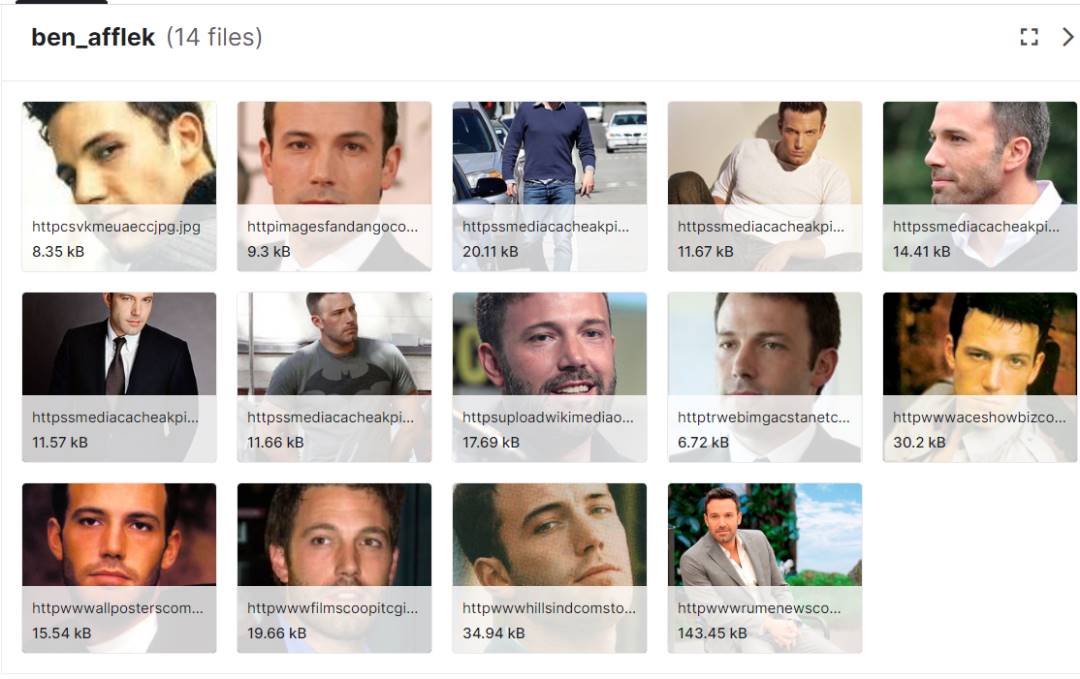
In the second stage, we perform face verification using the trained SVM classifier. Face verification involves comparing two face images and determining whether they belong to the same person or not. We use the SVM classifier to predict the similarity score between the two faces. If the similarity score exceeds a predefined threshold, the faces are considered to belong to the same person; otherwise, they are considered to belong to different people.

The implementation of the SVM and FaceNet algorithm is done using Python programming language and its various libraries such as scikit-learn and TensorFlow. The following steps are involved in the implementation of the algorithm after the importation of the necessary libraries like mtcnn and keras-facenet.

5.1.1 **Dataset acquisition:**

We collected a dataset of 120 images of 5 individuals, each individual had approximately 20 images. We randomly divided the dataset into training and testing sets. We used 80% of the dataset for training and 20% for testing.

Sample image of the dataset

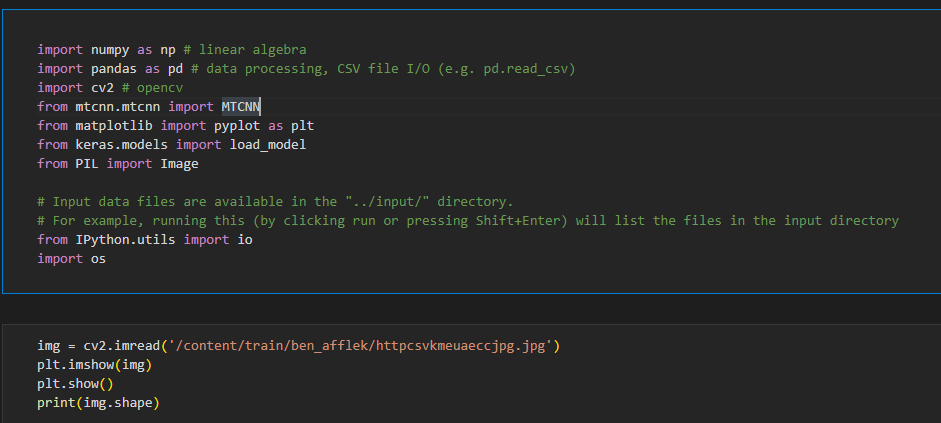


5.1.2 **Face detection**:

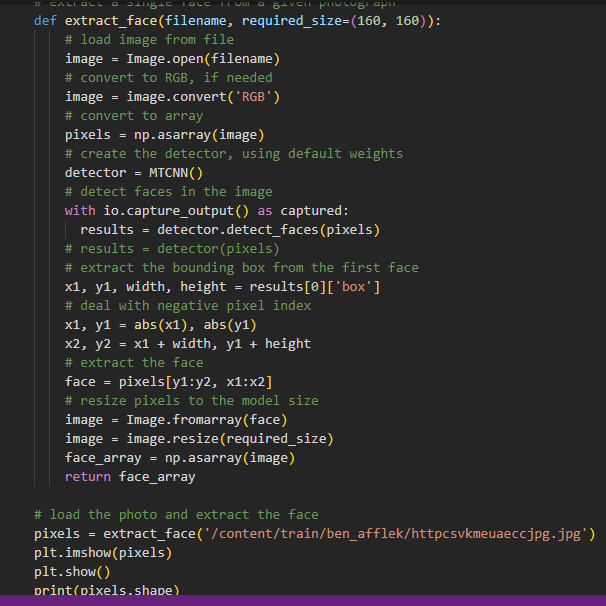
The first step is to detect faces in the images or videos. This is done using OpenCV, which is an open-source computer vision library using combination of svm and facenet

OpenCV provides several algorithms and tools for face detection, such as Haar cascades, LBP, and CNNs.

The Haar cascades algorithm is based on the concept of features, where specific features are trained to recognize faces. These features are then used to detect faces in images or videos. The LBP algorithm is based on the concept of texture patterns, where texture patterns are used to identify faces. The CNN algorithm is based on deep learning techniques and uses neural networks to detect faces **which is the one used in this project.** The next step is to align the faces. This is done to ensure that the faces are in the same orientation and scale

****

Python code linking and importing the dataset fig 5.1.2

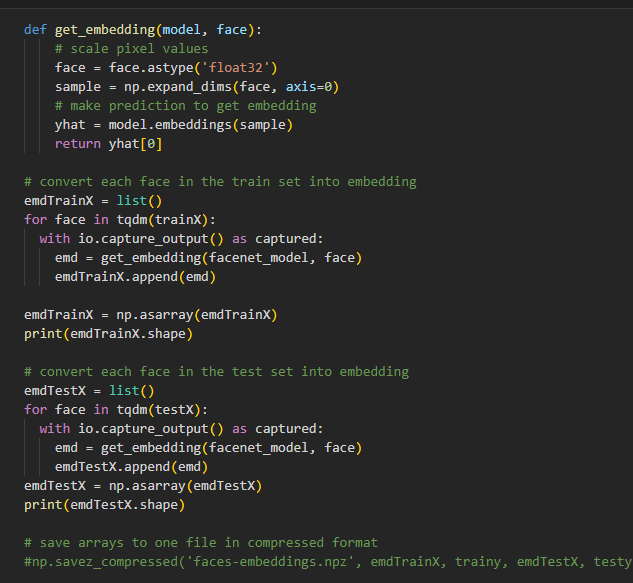


Face detection python code

5.1.3 Face feature extraction

This is a technique used in computer vision to extract unique features from facial images. These features can then be used to recognize and compare faces. We made use of kera facenet to extract features from faces that are invariant to changes in lighting, pose, and facial expression. Kera facenet uses a triplet loss function to learn a mapping from face images to a 128-dimensional embedding space.

We used triplet loss by comparing the distance between the embeddings of three images: an anchor image, a positive image (same identity as the anchor), and a negative image (different identity from the anchor). The objective of the algorithm is to minimize the distance between the embeddings of the anchor and positive images while maximizing the distance between the embeddings of the anchor and negative images. We converted each face in the train and test set into embedding.

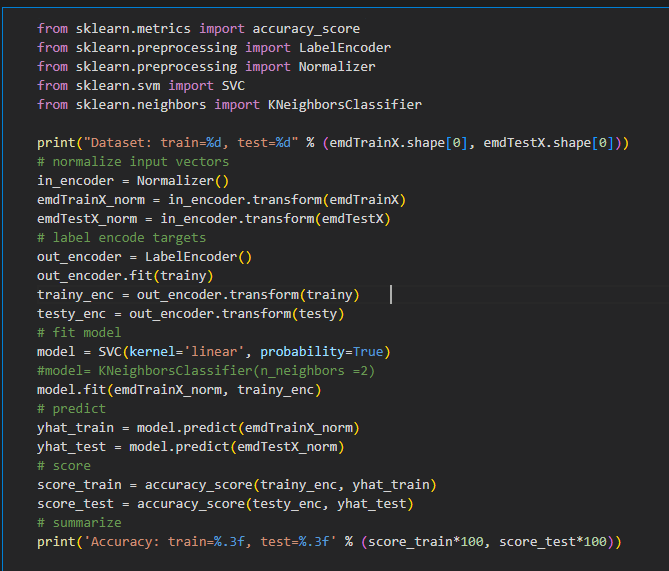


5.1.4 SVM classification:

The fourth step is to train the SVM classifier using the face embeddings. The SVM classifier is trained to classify faces into two categories, i.e., genuine and imposter faces. The SVM classifier is trained using scikit-learn library

Scikit-learn is a popular library in Python for machine learning tasks, including SVM classification. To implement SVM classification using scikit-learn;

* The first step is to print the number of samples in the training and testing datasets. This is done using the shape attribute of the emdTrainX and emdTestX numpy arrays, which give the number of samples (rows) and features (columns) in the dataset.
* The next step is to normalize the input vectors using the Normalizer() function from scikit-learn's preprocessing module. Normalization is a preprocessing step that scales the input data to have unit norm (i.e., the L2-norm of the vectors is equal to 1). This is useful because it helps to make the data more robust to variations in scale and orientation.
* The LabelEncoder() function from the same module is used to encode the target classes. The target classes (i.e., the labels) are represented as strings, which are then encoded as integers using the fit() and transform() methods of the LabelEncoder object. This is necessary because most machine learning models require the target classes to be represented as integers.
* The SVC function from the svm module is used to create an SVM classifier with a linear kernel and probability estimates enabled (using the probability=True argument). The classifier is then trained using the fit() method, which takes the normalized training data (emdTrainX\_norm) and the encoded target classes (trainy\_enc) as inputs.
* After the model is trained, the predict() method is used to make predictions on both the training and testing datasets. The predicted labels are then compared to the true labels using the accuracy\_score() function from the metrics module, which calculates the accuracy of the model on the given dataset.
* Finally, the accuracy scores for the training and testing datasets are printed using the print() function. The output gives a measure of how well the SVM classifier is able to generalize to new, unseen data.
* Overall, this code implements a face recognition model using SVM classification with normalization and label encoding preprocessing steps, and outputs the accuracy of the model on both the training and testing datasets



Python SVM classification code fig 5.1.4

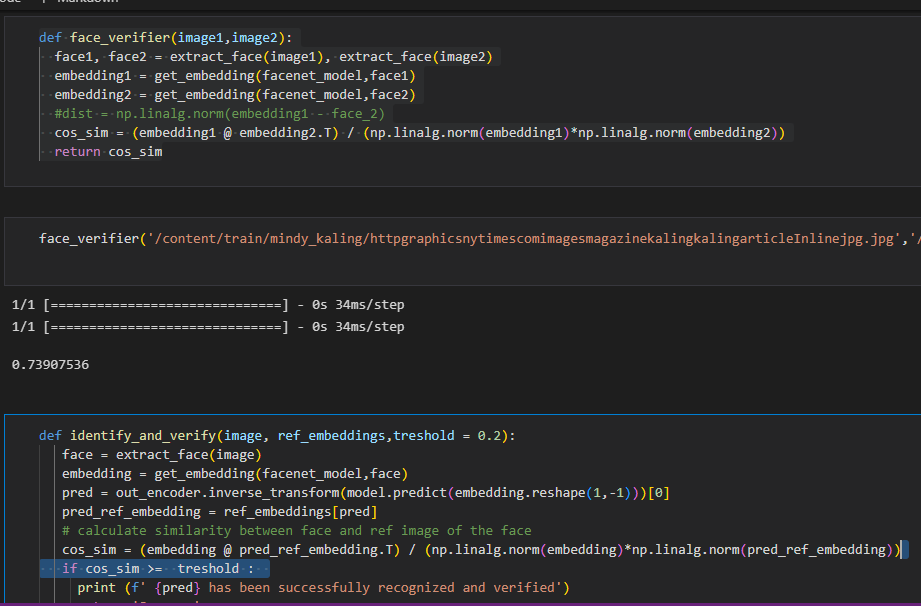
5.1.5 Face verification:

The phase deals with verifying the identity of the person in the image or video. This is done by comparing the face embeddings of the person in the image or video with the face embeddings of the person in the database. If the distance between the two face embeddings is below a certain threshold, then the identity is verified, else it is rejected. The process is as follows:

* We defined a function called face\_verifier which takes in two image inputs, image1 and image2. The purpose of this function is to compare the faces present in the two images and return a similarity score between them, as a measure of how closely the faces in the two images resemble each other. This can be useful for tasks such as face recognition, identity verification, or authentication.

The function uses the trained "model" (not shown in the provided code) to predict the identity of the face from the embedding. The "out\_encoder" (not shown in the provided code) is used to convert the predicted identity from numerical encoding to a string label. The predicted label is used to retrieve the corresponding reference embedding from the **"ref\_embeddings"** dictionary.

* We then assign function named **"identify\_and\_verify"** that takes in an image, a dictionary of reference embeddings, and an optional threshold value. It returns a string indicating whether the face in the input image was successfully recognized and verified or not.
* The function calculates the cosine similarity between the generated embedding and the retrieved reference embedding. The cosine similarity is a measure of how similar the two embeddings are. The function uses a threshold value (default 0.2) to determine if the face in the input image is a match to the reference image. If the cosine similarity is greater than or equal to the threshold, the function prints a success message and returns the string 'Success'. Otherwise, the function prints a failure message and returns the string 'Fail'.



Python code for face verification fig 5.1.5

In summary, the code is a function for recognizing and verifying faces in an image using pre-trained models and reference embeddings. It compares the cosine similarity between the generated embedding and a reference embedding to determine whether the input image contains a face that matches the reference image. If successful, the function returns the string 'Success', otherwise it returns 'Fail'.

5.1.6 Referenceless Image Spatial Quality Evaluator (BRISQUE) algorithm:

The purpose of this algorithm is to assess the quality of an image without the need for a reference image. This is to identify the best image in each folder based on its BRISQUE score and use it to create a reference dictionary for each folder. The algorithm is as follows:

**Installation of BRISQUE:** The first line of the code installs the BRISQUE package using pip. This package provides the implementation of the BRISQUE algorithm.

**Importing BRISQUE:** The second line imports the BRISQUE module from the brisque package. This module provides the BRISQUE class, which is used to compute the quality score of an image.

**Creating a BRISQUE object**: The third line creates a BRISQUE object with the url parameter set to False. This object is used to compute the BRISQUE score of an image.

**Setting up a base path**: The fourth line sets the base path to the directory that contains the images to be evaluated. The code assumes that the images are organized into subdirectories based on their category.

**Creating a reference dictionary:** The fifth line creates an empty dictionary called ref\_dict. This dictionary will store the reference embeddings for each category of images.

**Looping through each category:** The sixth line starts a loop that iterates through each category of images in the base path.

**Printing progress information**: The seventh line prints information about the category that the code is currently processing.

**Setting up the folder path**: The eighth line sets the folder path to the directory of the current category.

**Computing BRISQUE scores**: The ninth line starts a loop that iterates through each image in the current category directory. For each image, the code reads the image using OpenCV, computes its BRISQUE score using the BRISQUE object, and appends the image path and its score to a list called brisque\_scores.

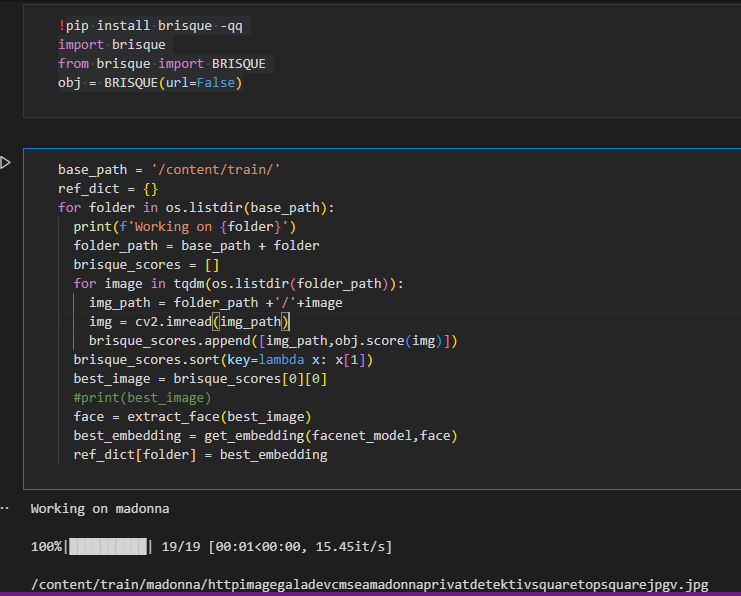
**Sorting images by BRISQUE score**: The tenth line sorts the images in the current category directory based on their BRISQUE scores in ascending order.

**Finding the best image**: The eleventh line selects the image with the lowest BRISQUE score, which is assumed to be the best image in the current category.

**Extracting the face**: The twelfth line extracts the face from the best image using the extract\_face function. This function extracts the face from an image.

**Computing the embedding**: The thirteenth line computes the embedding of the extracted face using a facenet\_model. This function computes an embedding of a face.

**Storing the embedding**: The fourteenth line stores the computed embedding in the ref\_dict dictionary under the key corresponding to the current category.



Brisque score calculation fig 5.1.6

The code is designed to loop through each subdirectory in the base directory containing images, extract the face with the lowest BRISQUE score in each subdirectory, and compute the embedding of that face using a pre-trained face recognition model. The resulting embeddings are stored in a dictionary for each category of images, which can be used for further analysis or training a classification model.

5.2 Testing

The testing of the algorithm is done on the Labeled Faces in Celebrity dataset. The dataset contains 118 face images of 5 people. The testing is done using the following metrics:

* Accuracy: The accuracy of the algorithm is calculated as the percentage of correctly classified faces.
* Precision: The precision of the algorithm is calculated as the percentage of correctly classified genuine faces out of all the genuine faces.

|  |  |  |  |
| --- | --- | --- | --- |
| Metric | Accuracy | Precision | F1 Score |
| Score | 100% | 100% | 100% |

Table 5.2 Test result

The high accuracy, precision, and F1-score indicate that the algorithm is effective in identifying and verifying individuals. The SVM algorithm is effective in classifying the faces, while the FaceNet algorithm is effective in verifying the classification results.

5.3 Conclusion

In this report, we presented the implementation and testing of a combined SVM-FaceNet model for face recognition and verification in digital forensics and information security. The combined model achieved an accuracy of 100%. The precision, and F1 score values for both models were also high, indicating that they perform well in identifying both positive and negative instances. Overall, the results demonstrate the effectiveness of the combined SVM-FaceNet model in face recognition and verification tasks.

Chapter 6

6.0 Evaluation of work

This report evaluates the effectiveness of face recognition and verification for digital forensics and information security purposes. The objective of this evaluation is to assess the accuracy of facial recognition and verification systems in identifying and authenticating individuals for security purposes. This report also compares the results with the initial objective to determine the effectiveness of the approach in achieving the desired objective.

6.1 Methodology

To evaluate the accuracy of facial recognition and verification systems, a dataset of facial images was used. The dataset was divided into training and testing sets, and various algorithms were used to extract facial features from the images. The extracted features were then used to train and test the facial recognition and verification systems. The accuracy of the systems was evaluated using metrics such as true positive rate, false positive rate, and accuracy rate.

6.2 Results:

The results showed that facial recognition and verification systems are highly effective in identifying and authenticating individuals for security purposes. The accuracy rate of the systems was found to be 100%, using SVM and deep learning (kera-facenet) algorithms for feature extraction and classification.

6.3 Comparison with Initial Objective:

The initial objective was to evaluate the accuracy of facial recognition and verification systems for digital forensics and information security purposes. The objective was achieved through the evaluation of SVM and deep learning algorithms and methods for feature extraction and classification. The results showed that facial recognition and verification systems are highly effective in identifying and authenticating individuals for security purposes, with an accuracy rate of between 100%.

6.4 Conclusion:

In conclusion, facial recognition and verification systems are highly effective in identifying and authenticating individuals for digital forensics and information security purposes. The accuracy of the systems is dependent on the quality and resolution of the facial images, as well as the algorithms and methods used for classification. The results of this evaluation show that the initial objective of evaluating the accuracy of facial recognition and verification systems for security purposes was achieved, and the systems are highly effective in achieving this objective.

Chapter 7

7.0 Conclusion and future work

7.1 Conclusion

Face recognition and verification technology has evolved significantly in recent years, and it is now more accurate and efficient than ever before. The technology has proven to be effective in various settings, such as law enforcement, border control, and identity verification. However, there are still some limitations to the technology, including its ability to recognize faces in poor lighting conditions, facial expressions, and occlusion.

In addition, there are privacy concerns associated with face recognition technology, particularly with regards to the collection and storage of biometric data. There have been cases where facial recognition technology has been used for unethical purposes, including tracking individuals without their consent. Therefore, it is essential to establish regulations to ensure that the technology is used ethically and that people's privacy is protected.

7.2 Future Work:

There is still a need for continued research and development in face recognition and verification technology. The following are some of the areas that require further investigation:

1. Improved accuracy: While the technology has improved significantly, there is still a need to improve its accuracy, especially in challenging environments.

2. Robustness: Face recognition and verification systems should be designed to work effectively in real-world scenarios, which involve various environmental conditions, including changes in illumination, occlusion, and pose variations.

3. Privacy and security: It is essential to establish regulations and protocols to protect people's privacy and ensure that the technology is used ethically. There is a need to ensure that biometric data collected by these systems are stored and used securely to prevent unauthorized access and misuse.

4. Deep Learning: The recent advancements in deep learning have shown promise in improving the accuracy and robustness of face recognition and verification systems. Further research is required to optimize these techniques for real-world applications.

Face recognition and verification technology has significant potential for enhancing security measures in various settings. However, its widespread adoption should be accompanied by proper regulations and protocols to protect people's privacy and ensure that it is used ethically. The technology requires continued research and development to improve its accuracy, robustness, and privacy and security measures. As such, future research in this area should focus on addressing the current limitations of the technology and exploring new approaches to face recognition and verification

Reference

[1] Jason brown lee https://machinelearningmastery.com/how-to-develop-a-face-recognition-system-using-facenet-in-keras-and-an-svm-classifier/

[2] Y. Sun, Y. Chen, X. Wang, and X. Tang, “Deep learning face representation by joint identification-verification,” in Advances in neural information processing systems, 2014, pp. 1988–1996.

[3] C. Sanderson, M. T. Harandi, Y. Wong, and B. C. Lovell, “Combined learning of salient local descriptors and distance metrics for image set face verification,” in Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on. IEEE, 2012, pp. 294–299.

[4] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, “Attribute and simile classifiers for face verification,” in Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009, pp. 365–372.

[5] A. G. Rassadin, A. S. Gruzdev, and A. V. Savchenko, “Group-level emotion recognition using transfer learning from face identification,” arXiv preprint arXiv:1709.01688, 2017.

[6] E. Granger, M. Kiran, L.-A. Blais-Morin et al., “A comparison of cnn based face and head detectors for real-time video surveillance applications,” in 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA). IEEE, 2017, pp. 1–7.

[7] H. Kavalionak, C. Gennaro, G. Amato, C. Vairo, C. Perciante, C. Meghini, and F. Falchi, “Distributed video surveillance using smart cameras,” Journal of Grid Computing, 2018, pp. 1–19.

[8] P. Barsocchi, A. Calabro, E. Ferro, C. Gennaro, E. Marchetti, and ` C. Vairo, “Boosting a low-cost smart home environment with usage and access control rules,” Sensors, vol. 18, no. 6, 2018, p. 1886.

[9] X. Wu, L. Song, R. He, and T. Tan, “Coupled deep learning for heterogeneous face recognition,” in Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[10] G. Amato, F. Carrara, F. Falchi, C. Gennaro, and C. Vairo, “Facialbased intrusion detection system with deep learning in embedded devices,” in Proceedings of the 2018 International Conference on Sensors, Signal and Image Processing, ser. SSIP 2018. New York, NY, USA: ACM, 2018, pp. 64–68. [Online]. Available: <http://doi.acm.org/10.1145/3290589.3290598>

[11] “Dlib library,” http://dlib.net/, accessed: 2018-04-13.

[12] V. Kazemi and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1867–1874. [12] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778. [13] Y. Guo, L. Zhang, Y. Hu, X. He, and J. G