Automatic Attendance Management System based on Deep One-Shot Learning

Angelo G. Menezes[1], João M. D. da C. Sá[1], Eduardo Llapa[2], Carlos A. Estombelo-Montesco[1]
[1]Federal University of Sergipe, Department of Computer Science, Av. Marechal Rondon, São Cristóvão, 49100-000, Brazil
[2]Federal University of Mato Grosso do Sul, Information Systems Course, Av. Rio Branco, 1270, Corumbá, 79304-902, Brazil
angelo.menezes@dcomp.ufs.br

***Abstract*—Due to the positive relationship between the presence of students in classes and their performance, student attendance assessment is considered essential within the classroom environment, even as a tiring and time-consuming task. We proposed a solution for student attendance control using face recognition with deep one-shot learning and evaluated our approach in different conditions and image capturing devices to confirm that such a pipeline may work in a real-world setting. For better results regarding the high number of false negatives that often occur in uncontrolled environments, we also proposed a face detection stage using HOG and a CNN with Max-Margin Object Detection based features. We achieved accuracy and F1 scores of 97% and 98.4% with an iPhone 7 camera, 91.9% and 94.8% with a Moto G camera, and 51.2% and 61.1% with a WebCam respectively. These experiments reinforce the effectiveness and availability of this approach to the student attendance assessment problem since the recognition pipeline can be either made available for embedded processing with limited computational resources (smartphones), or offered as "Software as a Service" tool.**

***Keywords*—Face Recognition, Deep Learning Applications, One shot Learning, Image Processing, Attendance System**

I. Introduction

There is an intrinsic positive relationship between class attendance and the performance of students in the academic environment [1]. For the learning to occur more naturally, it is necessary to encourage presence and participation in classes in a progressive way, so that the student can relate to topics discussed in previous courses.

When considering the importance of performing student attendance assessment, the current traditional ways still take up a great deal of class time and can be easily fooled. Regarding the real presence of students during a class, it is known that this specific situation deserves attention since the assessment can even be used as an alibi in some legal cases.

In recent years several facial recognition algorithms have been developed to perform recognition regardless of environment, angle, and facial expression. Considering its application for student attendance assessment, it becomes a promising approach, since face recognition has several benefits compared to other biometric methods that are intrusive and require human interaction with different devices [2] [3].

In this paper, we present the experiment for a system that automates class attendance assessment through facial recognition using machine learning algorithms. Our method is based on using a single image of a student for training a system that is meant to detect, segment, and verify student identities in an uncontrolled environment (class pictures). For this, we applied different concepts of computer vision (face detection, alignment, and verification) and one-shot learning through the use of a pre-trained deep neural network trained in over 200 million images (FaceNet) [4]. We evaluated the system in three different camera settings in order to check the robustness against image resolution.

The rest of the paper is divided as follows: Section II provides a quick technical background to the subjects here discussed; Section III discusses recent related works regarding face recognition and attendance automation; Section IV presents the methodology and the system overview; Section V and VI respectively show the results of the experiments, final considerations, and future work.

II. Technical Background

A. Deep Learning

Deep Learning (DL) is a branch of machine learning (ML) that is capable of learning the data representation through the use of a structure of hierarchical layers, similar to the way the brain handles new data. DL can be described as a concept that can be applied to some sub-fields of ML since it represents a way of how to approach problems [5].

The state-of-the-art in image classification has been dominated by DL algorithms since the launch of the ImageNet challenge [6]. Due to the increase in computational power provided by cheaper GPUs, researchers and practitioners have applied DL models to a range of different tasks that are related to image classification. This high interest of researchers in DL has paid off since this class of algorithms has become the state-of-the-art in significant branches of computer vision such as object detection, semantic segmentation, and face recognition [7].

Regarding face recognition, the application of deep learning models has helped the automation process within biometrics, where the accuracy of machines has overcome the one of humans. The importance of such claims can be noticed even in real-time situations where computational resources are limited (e.g., unlocking of a mobile device with face ID) [8].

978-1-7281-7539-3/20/$31.00 ©2020 IEEE

B. Face Detection

Face detection in an image is the first step in a recognition pipeline because it eliminates unnecessary information from the image. In this way, if the algorithm finds one or more faces, they are extracted from the original image so that they can be analyzed separately [9].

The training of such algorithms happens with the use of several different images with faces and others without them. Even though this problem presents itself as a simple binary classification, several face detection algorithms need to be trained exhaustively so that they can give good results [10].

Depending on the algorithm used, face detection and feature extraction are performed simultaneously. Two measures are responsible for evaluating the quality of these algorithms and whose ideal results are null [11]:

- False positive: Represents the number of objects that were detected wrongly as faces.
- False negative: Represents the number of faces that were not detected.

Face detection algorithms are usually divided into four different groups: knowledge, template, feature, and appearance-based models [12].

Knowledge-based methods use pre-defined rules based on human knowledge of the position, distance, and distribution of face elements. Template matching methods use pre-stored face templates to determine where a human face is depicted in an image. In this method, the existence of a face is determined based on the correlation values among face contour, eyes, nose, and mouth independently.

Feature invariant approaches aim to find face structure features robust to pose and lighting variations. Based on the extracted salience facial features usually using edge detectors, a statistical model is built to describe these features and to verify the existence of a face. A generally applied feature-based method for face detection is the Histogram of Oriented Gradients (HOG), which works by using the distribution of gradient directions as image features.

Appearance-based methods learn to generalize face models from a set of representative training face images which may be used for face detection. They usually require more training data than other traditional methods and may have increased computational complexity. A common concept for the state-of-the-art algorithms in this group of techniques is the use of Convolutional Neural Networks (CNN). CNNs derive problem-specific feature extractors from the training examples automatically, without making any assumptions about the features to extract or the areas of the face patterns to analyze since they are spatially invariant [13].

When training a CNN for a specific task such as the ones related to object detection, it is advisable to specify a loss function that better suits the convergence of the training process. One efficient type of loss function that works both for general object detection and face detection is the one based on max-margin optimization. This method optimizes over all sub-windows available within the dataset, which decreases the computational complexity when trying to find different sizes of objects. A general advantage of this cost function is that high-quality detectors can be proposed from relatively small amounts of training data [14].

The approach applied in this work utilizes a hybrid between appearance and feature-based algorithms (CNN detector and HOG) for improving face detection performance in uncontrolled environments.

C. Face Recognition

Face recognition consists of the representation and extraction of facial features of an image as input into a mathematical model, which is meant to specify whether the presented face matches one or any previously saved face in a database [15]. This task can be used for two possible outcomes:

- Verification: The model considers a binary classification to indicate the authenticity of a particular user.
- Identification: The model considers the challenge of multi-class classification (one-to-many) where it is found the relation of the tested face with all the others present in the database.

The implementation of recognition systems can range from low-throughput to process-intensive methods where, for example, GPUs are required. Some more straightforward methods can be based on nearest-neighbor or principal component analysis methods. On the other hand, the most sophisticated ones are usually based on analysis of probability densities, manifold learning, and deep neural networks, among other methods with a higher computational cost [16].

For extracting discriminative features of an image that only contains a face (after the pre-processing step), models based on CNN have been the ones most used by state-of-the-art approaches. This architecture is suitable for feature extraction because it takes advantage of local connections to extract the spatial information effectively and shared weights to significantly reduce the number of parameters for training the network [17].

The method used in this work presents a deep convolutional neural network structure that extracts the characteristics of the face and projects them onto a vector subspace where faces of the same person have a small Euclidean distance (L2 distance) and faces of different people present higher values for the same metric.

III. Related Work

One of the first works dealing with automation in attendance recording was presented in South Africa. In its operation, radiofrequency identification technology worked with an internet hotspot, where each student contained an identification card, and the information of that card was obtained automatically, regardless of the card's location (e.g., pocket or backpack). The model was implemented in some schools, but it had the disadvantage of being easily circumvented since a student or unknown person could pretend to be another by just carrying the card [2].

To solve problems regarding the veracity of the acquired records, facial recognition appeared like a reasonable solution since several approaches proved to be efficient and reliable when using it as a biometric tool [18]–[20].

The work of Sarkar, Mishra, and Subhramanyam (2019) presented an approach divided into two sections: face detection and face verification. For the former, they used a CNN with a specific residual architecture, proposed in the paper of He et al. (2016) [21], which was the state-of-the-art in 2017 for finding small faces in uncontrolled environments [22]. For the latter, they applied a Spatial Transformer Network for alignment and then used another deep CNN with inception architecture inspired by the Deep Face architecture [23]. They presented results close to 100% for the attendance system; however, their method was validated on an environment with an apparent high-resolution camera, and therefore it lacked answers regarding different camera settings [24].

In the work of Chauhan, Pandey, and M (2018), the pipeline used for face recognition and attendance assessment used HOG for face detection, feature extraction via CNN with an architecture inspired by FaceNet [4], and support vector machines (SVM) for classification. The presented experimentation results seemed successful. However, they had their experimentation set in a controlled environment, where they tested only using one camera setting [25].

Arsenovic et al. (2017) applied a CNN cascade method for face detection, another CNN with FaceNet architecture for feature extraction, and SVM for classification. Their work differs from others because they used some data augmentation tricks to enlarge their dataset using prior knowledge of nose, eyes, chin, and mouth. Also, they cross-validated the attendance results via face recognition against data from RFID cards, and their system reached over 95 % of accuracy. Nevertheless, their system was only evaluated with one high-resolution camera configuration also in a controlled environment [26].

IV. Methodology

A. Our approach

Our approach consists in performing the student attendance assessment through the use of a single picture from the class taken by a smartphone device or camera installed in the room and passing it forward to the recognition system. The system, which might be masked within the internal gateway of the university, is then responsible for detecting, aligning, and verifying if students are present or absent using the pipeline described in Figure 1. Also, we take into consideration three possible setups for the image capturing device in order to evaluate, which may give the most reliable results while keeping a low cost of implementation.

B. Face Detection

Since our approach needs to be able to work in an uncontrolled environment where the face of students might be occluded or not aligned with the camera, a hybrid approach was set specially to solve possible issues related to false negative occurrences.

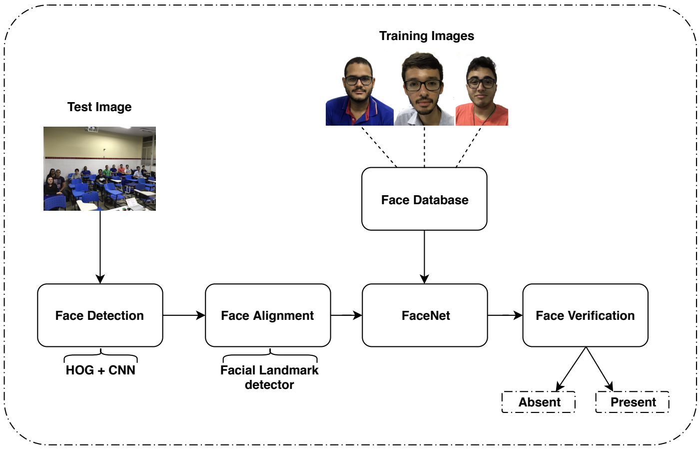

Fig. 1. Face Recognition Pipeline

We used a HOG detector along with a pre-trained CNN detector with max-margin architecture, both implemented by the dlib library [27], to overcome possible false negatives since the number of students that are supposed to be present in each class is known a priori. The HOG detector is first applied to the testing image in order to detect a number of faces equal to the number of students in the class. If the detector fails to find all possible faces, the CNN detector is then applied to the image. We evaluated our system with other more straightforward approaches, for example, using the Viola-Jones algorithm. However, our application is highly sensitive against false positives since we do not have many images for training on the recognition/verification step, which made the use of a simpler approach impracticable. Our approach takes into consideration the fact that HOG detectors are around 10x faster than CNN detectors regarding computational cost in devices such as the ones used in our experiment that may not have dedicated hardware for parallel image processing.

C. Face Alignment

For passing the segmented face forward to the FaceNet architecture, it was advised by their authors that such architecture works better with aligned frontal faces. Therefore, we applied a face alignment step using once again the dlib library with their pre-trained 68 facial landmark detector, which gave as output an enlarged 96x96 aligned face as can be seen in the example of Figure 2.

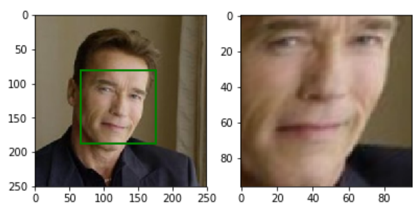

Fig. 2. Output example of the face alignment step.

D. Face Recognition

1) Face Feature Extraction: The FaceNet architecture [4] proposed a new approach for face recognition and face clustering with its deep convolutional architecture based on GoogLeNet [28], a 22 layers deep network. It takes as input an image of a segmented aligned face, and it outputs a 128 dimension embedding that better compacts the features presented in the face. The crucial point of using such a model is that it was trained to minimize the Euclidean distance of embeddings of the same person at the same time it maximizes the same distance between embeddings of different people through the use of a triplet loss following the structure presented in Equation 1.

$$\|x_i^a - x_i^p\|_2^2 + \alpha - < \|f(x_i^a) - f(x_i^n)\|_2^2, \forall (x_i^a, x_i^p, x_i^n) \in \tau \quad (1)$$

The loss to be minimized is then described in Equation 2:

$$Loss = \sum_{i}^{N} \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right] \quad (2)$$

Where: α is a margin/threshold that is enforced between positive and negative pairs; x_i^a, known as anchor embedding, is the reference face; x_i^p is the positive embedding with the same identity of the anchor; x_i^n is the negative sample; and τ is the list of all possible triplets within the dataset. A visual description of the learning process is described in Figure 3.

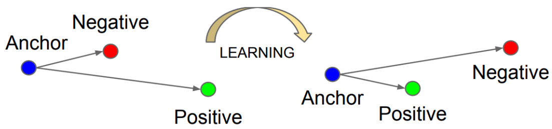

Fig. 3. Triplet Loss anchor example [4].

We used a pre-trained model with this architecture provided by the Openface community [29], which was trained in over 500k images from combining two large labeled face recognition datasets (CASIA-WebFace [30] and FaceScrub [31]). Since for our experiments we were able to provide only one image of each student for training, the solution using this type of architecture was particularly useful to our problem. This characterizes the approach as one-shot learning since the system needs to learn the best threshold that separates all classes (students) using only one example per class in the verification step. An example of the difference in Euclidean distances for two pairs of different students is shown in Figure 4.

2) Face Verification: In order to verify if two faces are from the same person, we use a threshold based on the Euclidean distance of the face embeddings. If the distance is below the threshold, the embeddings are from the same person. Otherwise, they belong to different people. This threshold was set initially based on experimentation on the Labeled Faces in the Wild dataset [32] along with the evaluation of other works that use

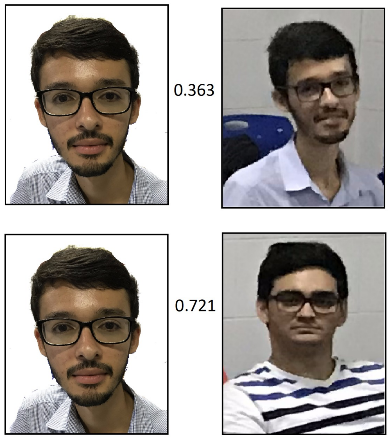

Fig. 4. FaceNet example of embedding distances.

the FaceNet architecture. This approach of comparing distances with only one example per class works similarly as a K-Nearest Neighbor classifier, where the K is set to 1.

One trick that can be applied to the context of face verification using this architecture is that, if we have a situation where more than one student has a Euclidean distance lower than the threshold for a specific identity, we assume that the student's real identity is when the distance is minimal (closer to zero).

E. Experiments

1) Training setup: For obtaining one training image for each student, the image capturing scheme was set with the placement of a camera at a distance of 1.5m from the student, which follows the well-known standard for 3x4 pictures. The height of the camera was adjusted ideally for taking a centered picture of the face.

As the first experiments showed the robustness of our pre-trained model, and there was only one image per student for training, we decided not to proceed with any fine-tuning technique for our final architecture. However, in a real-world setting, such a step may be needed, and an online training step may be applied along with each successful attendance recognition case in order to update the model.

For updating the threshold and finding its best value for discriminating the faces in testing time, we first calculated the Euclidean distance of every student's face against all the other student's faces within the class in order to determine which would be the minimum threshold that would imply that two students are different people.

2) Testing setup: For the testing setup, two experiments were designed for validating our approach: evaluation of different distances for face detection and assessment of the recognition pipeline using image capturing devices with different resolutions.

In order to grant a fair comparison of results among devices, seven student classrooms were analyzed for checking the face detection rates with each image capturing unit. Table I shows the number of students in each classroom.

TABLE I
NUMBER OF STUDENTS IN EACH CLASSROOM

Class	1	2	3	4	5	6	7
N# of students	16	18	12	17	10	33	19

The testing image of the class was taken according to the setup in Figure 5. For the first experiment, students sat in different positions inside the classroom based on the following distances far from the camera: 2, 4, 6, 7, 8, and 10 meters. The testing image was taken with the camera in a height position where there was no occlusions, or only partial ones in order to evaluate the robustness of the system for detecting and recognizing the faces in an uncontrolled environment.

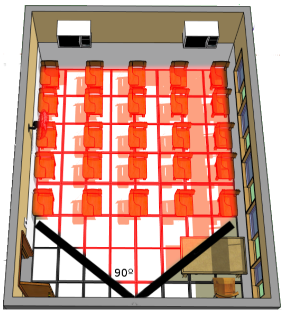

Fig. 5. Angle perspective for testing images.

The performance of the face recognition pipeline was evaluated with the following image capturing devices: iPhone 7 (12MP), Moto G (8MP), and a general WebCam (1.2MP). It is known that image resolution affects the recognition [33] directly, notwithstanding, one of the goals of this experiment was to evaluate how affected the results may be based on this difference in image quality. The findings related to these experiments are described in the next section.

V. RESULTS

A. Face Detection Results

Using the setup described in the last section, the face detection experiment was executed, and the algorithms were able to detect 100% of all faces present in the images taken by the smartphones, regardless of the distance inside the classroom, while it failed to recognize every face farther from 2 meters taken by the WebCam. One advantage in the design of such a system is that the number of faces supposed to be present in every test image is known a priori since the number of students in each classroom is easily determined. This situation facilitated the face detection phase since we were able to check "manually" for false negatives and explore different approaches, which improved the overall results of the face recognition pipeline consequently.

When testing with only the HOG detector instead of the hybrid approach, 5% of all the students' faces were not detected for the smartphones when using the testing images. This led to the application of the indicated hybrid setup, since it solved the occurrence of false negatives and resulted in no failing cases.

B. Face Recognition Results

The results regarding accuracy and the F1 score metrics can be seen in Table II and III. The described results were obtained using a threshold of 0.6 for face verification, which was the best discriminative value. The results are represented as absolute values since there was no variation in the recognition step when presenting the same testing images to the pipeline.

TABLE II
ACCURACY FOR EVERY TESTED CLASSROOM

Class	Iphone7 (12 MP)	Moto G (8 MP)	Webcam (1.2 MP)
1	100%	100%	62.5%
2	94.4%	88.9%	44.4%
3	100%	100%	25%
4	100%	82.4%	76.5%
5	90%	80%	50%
6	100%	97%	57.6%
7	94.7%	94.7%	42.7%
Average	97.1%	91.9 %	51.2%

TABLE III
F1 SCORE FOR EVERY TESTED CLASSROOM

Class N°	Iphone7 (12 MP)	Moto G (8 MP)	Webcam (1.2 MP)
1	100%	100%	66.6%
2	96.3%	92.3%	44.5%
3	100%	100%	40%
4	100%	88%	84.6%
5	94.7%	88%	66.6%
6	100%	98.3%	69.6%
7	97.3%	97.3%	56%
Average	98.3%	94.8 %	61.1%

The iPhone 7 got an average accuracy of 97.10 % while the MotoG achieved 91.90 %, and the Webcam 51.2 %, which reinforces the importance of resolution when designing such a system for a real-world face recognition application. The results were somehow expected to be around such range since similar related works got around 95% on average; however, their systems were not analyzed taking into consideration the F1 score which is an essential metric since a high amount of false negatives or false positives may make the implementation of such technology unfeasible.

Our system was able to reach an average 98.3%, 94.8%, and 61.1% F1 scores with the different related devices. The results obtained by the 1.2 MP Webcam show that even though cheaper, this image capturing device was not able to effectively provide enough details when students were placed far from the device, which makes it unable for implementation in such environment. However, the results obtained using smartphone images restate the effectiveness of such face recognition solution for the student assessment problem either being offered in a "Software as a Service" platform, where an API could be set for carrying the computational load, or utilizing the embedded processing power of recent smartphones with offline pre-trained models.

VI. Conclusion

As could be seen by the provided results, the student attendance assessment task may be solved by the use of face recognition even in the presence of limited resources. Although the performance of the pipeline was highly attached to the quality of the image capturing device, as the technology advances, the prices of such devices tend to be even more affordable, which contributes to the implementation of automation systems such as the one idealized in this work for student attendance assessment.

The difference in results obtained from the iPhone 7 and Moto G devices was consistent with the difference in their prices, which need to be taken into consideration when designing such a system. One point to highlight is the importance of the relation between image resolution and the face detection step since, if the device is not able to provide enough details in the image to detect the faces, the pipeline will not be able to achieve good results when checking for more robust metrics such as the F1 score.

Regarding system improvements, different setups for the verification setting may be implemented. For example, we may use data augmentation techniques to increase the number of images used for training within each student database. Also, we could explore other machine learning methods that can take advantage of a larger base for training in order to increase the reliability and robustness of the system.

As future work, we could extend the attendance assessment system by adding metrics related to student performance in order to enable early detection of dropout in undergraduate studies. In addition, such a system may allow educators to analyze other problems correlated with student attendance and take measures to improve the present educational environment.

References

[1] S. Devadoss and J. Foltz, "Evaluation of factors influencing student class attendance and performance," *American Journal of Agricultural Economics*, vol. 78, no. 3, pp. 499–507, 1996.

[2] B. N. Gatsheni, R. B. Kuriakose, and F. Aghdasi, "Automating a student class attendance register using radio frequency identification in south africa," in *2007 IEEE International Conference on Mechatronics*. IEEE, 2007, pp. 1–5.

[3] R. Kuriakose and H. Vermaak, "Developing a java based rfid application to automate student attendance monitoring," in *2015 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference*. IEEE, 2015, pp. 48–53.

[4] F. S. e D. Kalenichenko e J. Philbin, "Facenet: A unified embedding for face recognition e clustering," in *2015 IEEE Conference on Computer Vision e Pattern Recognition (CVPR)*, June 2015, pp. 815–823.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.

[7] O. M. Parkhi, A. Vedaldi, A. Zisserman *et al.*, "Deep face recognition." in *BMVC*, vol. 1, no. 3, 2015, p. 6.

[8] L. Zhao and R. Tsai, "Locking and unlocking a mobile device using facial recognition," 2015, uS Patent 8,994,499.

[9] Y. Muttu and H. Virani, "Effective face detection, feature extraction & neural network based approaches for facial expression recognition," in *2015 International Conference on Information Processing (ICIP)*. IEEE, 2015, pp. 102–107.

[10] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[11] V. Vezhnevets, "Face and facial feature tracking for natural human-computer interface," 2002.

[12] S. Zafeiriou, C. Zhang, and Z. Zhang, "A survey on face detection in the wild: past, present and future," *Computer Vision and Image Understanding*, vol. 138, pp. 1–24, 2015.

[13] C. Zhang and Z. Zhang, "A survey of recent advances in face detection," 2010.

[14] D. E. King, "Max-margin object detection," *arXiv preprint arXiv:1502.00046*, 2015.

[15] N. Crosswhite, J. Byrne, C. Stauffer, O. Parkhi, Q. Cao, and A. Zisserman, "Template adaptation for face verification and identification," *Image and Vision Computing*, vol. 79, pp. 35–48, 2018.

[16] M. Wang and W. Deng, "Deep face recognition: A survey," *arXiv preprint arXiv:1804.06655*, 2018.

[17] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6232–6251, 2016.

[18] M. G. Krishnan and S. B. Balaji, "Implementation of automated attendance system using face recognition," *International Journal of Scientific & Engineering Research*, vol. 6, no. 3, 2015.

[19] P. Wagh, R. Thakare, J. Chaudhari, and S. Patil, "Attendance system based on face recognition using eigen face and pca algorithms," in *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*. IEEE, 2015, pp. 303–308.

[20] B. Surekha, K. J. Nazare, S. V. Raju, and N. Dey, "Attendance recording system using partial face recognition algorithm," in *Intelligent techniques in signal processing for multimedia security*. Springer, 2017, pp. 293–319.

[21] 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.

[22] P. Hu and D. Ramanan, "Finding tiny faces," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 951–959.

[23] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[24] P. R. Sarkar, D. Mishra, and G. R. S. Subhramanyam, "Automatic attendance system using deep learning framework," in *Machine intelligence and signal analysis*. Springer, 2019, pp. 335–346.

[25] R. K. Chauhan, V. Pandey, and M. Lokanath, "Smart attendance system using cnn," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 15, pp. 675–680, 2018.

[26] M. Arsenovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Facetime—deep learning based face recognition attendance system," in *2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY)*. IEEE, 2017, pp. 53–58.

[27] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[29] B. Amos, B. Ludwiczuk, and M. Satyanarayanan, "Openface: A general-purpose face recognition library with mobile applications," CMU-CS-16-118, CMU School of Computer Science, Tech. Rep., 2016.

[30] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *arXiv preprint arXiv:1411.7923*, 2014.

[31] H.-W. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 343–347.

[32] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[33] P. Li, L. Prieto, D. Mery, and P. Flynn, "Low resolution face recognition in the wild," *arXiv preprint arXiv:1805.11529*, 2018.