

EmgAuth: An EMG-based Smartphone Unlocking System Using Siamese Network

Boyu Fan*, Xuefeng Liu*, Xiang Su[§], Pan Hui^{§¶}, Jianwei Niu^{*†‡||}

*State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China

†Beijing Advanced Innovation Center for Big Data and Brain Computing (BDBC), Beihang University

‡Hangzhou Innovation Research Institute, Beihang University

§Department of Computer Science, University of Helsinki, Finland

¶Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong

||School of Information Engineering, Zhengzhou University

zy1715107@buaa.edu.cn, csxfliu@gmail.com, xiang.su@helsinki.fi, panhui@cse.ust.hk, niujianwei@buaa.edu.cn

Abstract—Screen lock is a critical security feature for smartphones to prevent unauthorized access. Although various screen unlocking technologies including fingerprint and facial recognition have been widely adopted, they still have some limitations. For example, fingerprints can be stolen by special material stickers and facial recognition systems can be cheated by 3D-printed head models. In this paper, we propose EmgAuth, a novel electromyography(EMG)-based smartphone unlocking system based on the Siamese network. EmgAuth leverages the Myo armband to collect the EMG data of smartphone users and enables users to unlock their smartphones when picking up and watching their smartphones. In particular, when training the Siamese network, we design a special data augmentation technique to make the system resilient to the rotation of the armband. We conduct experiments including 40 participants and the evaluation results show that EmgAuth can effectively authenticate users with an average true acceptance rate of 91.81% while keeping the average false acceptance rate of 7.43%. In addition, we also demonstrate that EmgAuth can work well for smartphones with different sizes and at different locations, and is applicable for users with different postures. EmgAuth bears great promise to serve as a good supplement for existing screen unlocking systems to improve the safety of smartphones.

Index Terms—Electromyography, authentication, Siamese Network, unlocking, smartphone

I. INTRODUCTION

Screen lock is a critical security feature for smartphones to prevent unauthorized access. Various unlocking technologies have been developed to protect the safety of smartphones [1]. In particular, biometric-based technologies, including fingerprint, face recognition, and iris [2], have gradually replaced the traditional password-based methods [3] in recent years. These methods enable safe and convenient authentication compared with the password method. However, there are still some limitations on these methods. For example, fingerprints can be obtained with packing tapes to unlock the smartphone [4]. Face can also be deceived, and studies have shown that 3D printed head models plus taped glasses can easily fool Apple's Face ID authentication system [5].

Different from the above biometric features, the EMG signal, collected by placing electrodes on one's skin to detect the electrical activity of the muscle, shows unique features

for individuals and therefore has a great potential for authentication [6]. In particular, we observe that when different people pick up their smartphones, the speed, wrist movement, fingers movement and the positions they grab smartphones are generally different for different persons. In contrast, for a certain person, the movement when picking up a smartphone is generally similar, which can be attributed to the memory of one's muscles accumulated for a long period of time.

Based on above mentioned observations, we use EMG to achieve smartphone unlocking. The uniqueness of the EMG signal and the consistency of picking up movements are crucial for EMG-based authentication. Although there are some existing works which utilize EMG signal to unlock smartphones [7] [8], they require users to make a series of pre-defined gestures. In addition, the systems to collect EMG signal, like the Myo armbands, need to be placed in the same position on the arm for both training and testing stages. These limitations greatly limit the applicability of the EMG-based unlocking system as well as other EMG-based applications [9] [10].

In this paper, we propose EmgAuth, a new EMG-based smartphone unlocking system based on the Siamese network. EmgAuth utilizes data collected from the Myo armband and allows users to unlock their smartphones when picking up and watching their smartphones without making any pre-defined gestures. More importantly, when training the Siamese network, we design a special data augmentation technique to make the system resilient to the rotation of the armband. These two improvements greatly enhance the usability of the unlocking system. We implement EmgAuth on Android smartphones and recruit 40 participants to collect training and testing data for EmgAuth.

We conduct a series of experiments to choose the proper parameters of EmgAuth, including the hyperparameters of the Siamese network and the threshold of the classifier. The results of cross validation demonstrate that EmgAuth can achieve good authentication accuracy in a real-time manner. It can authenticate users with an average true acceptance rate of 91.81% while keeping the average false acceptance rate of 7.43%, and the overall accuracy reaches 92.06%. We also

discuss some influencing factors in real scenarios to verify the feasibility of EmgAuth. In addition, the authentication latency of EmgAuth that run in an Android smartphone is about 0.16s, which fulfills the requirement of real-time unlocking.

The contributions of this paper are summarized as follows:

- 1) EmgAuth, a system that unlocks smartphones by usual motions based on EMG signal and Siamese network. This is one of the first research effort that combines EMG signal with deep learning to unlock smartphones.
- 2) A novel method based on the structure of Myo armband to make EmgAuth resilient to rotation of armband. With this method, users do not need to calibrate or remember the position of armband.
- 3) Extensive experiments to verify the feasibility and reliability in different conditions.

The remainder of this paper is organized as follows. In Section II, we discuss some related work about common biometric authentication methods, EMG-based applications and the siamese network. Section III details our EmgAuth system architecture and each module. We then describe our dataset and provide the experimental results of our system in Section IV. In Section V, we discuss some influencing factors in real scenarios. The conclusion, limitation and future research directions are given in the last section.

II. RELATED WORK

This section reviews the related research in biometric authentication methods, EMG-based applications and the Siamese neural network.

A. Biometric Authentication

Biometric authentication is widely used in daily life, such as transaction, login, etc. Among various kinds of authentication methods, fingerprint is one of the most common technology. Anil Jain et al. [11] first describes the design and implementation of an online fingerprint authentication system. An alignment-based elastic matching algorithm is developed to find the correspondences between minutiae in the input image and the stored template. Facial recognition is another popular technology for identity authentication. Sun [12] combines deep learning techniques with face identification, they use deep convolutional neural networks to learn features to reduce intra-personal variations while enlarging inter-personal differences. The face verification accuracy can achieve 99.15%.

In addition to physiological characteristics related methods, behavioral characteristics also interest researchers. Keystroke dynamics is used as a kind of biometrics for authentication [13]. Monroe et al. [14] innovatively propose a new authentication method based on analyzing habitual rhythm patterns when users type. They present data extraction methods, as well as classification strategies to achieve user authentication and the accuracy can reach 92.14%. Gait, hand-waving and signature are also used to achieve authentication [15] [16] [17].

B. EMG-based applications

EMG records the movement of muscles. It is based on the simple fact that whenever a muscle contracts, a burst of electric activity is generated which propagates through adjacent tissue and bone and can be recorded from neighboring skin areas. Therefore, EMG signal is widely used in the area of medicine [18], control [19], human-computer interaction [20] and games [21]. Kiguchi et al. [22] use EMG signal to control an upper-limb power-assist exoskeleton robot, which is easy, humanlike and adaptable to any users. EMG-based hand gesture identification can help to develop a better human-computer interaction interface. In [23], Ahsan described the process of detecting different hand gestures using artificial neural network (ANN). They use a series of statistical methods to extract features and then feed these feature vectors to ANN to get a classification result. EMG signal is also combined with other sensors to achieve complicated control. Yoo et al. [24] propose an input device for a virtual reality game, which is based on EMG and accelerometers. The results show the device can offer good experience for players. In addition, Myo armband is one of the most popular devices for EMG-related research because of the portability and efficient data transmitting mechanism [25] [26] [27]. Comparing with the existing work [28] [29], EmgAuth does not need extra training set and is resilient to the positions of EMG sensors, and does not need users to do extra actions.

C. Siamese Network

Deep neural networks have excellent performance in the fields of image classification, speech recognition and natural language processing. They can automatically extract features from large-scale data rather than conducting feature engineering manually. Many structures of network are proposed to deal with different kinds of tasks [30] [31] [32]. Siamese network was first introduced by Bromley et al. to solve the problem of signature verification [33]. They design two identical sub-networks to extract features and combine them with a layer that computes the distance between the two outputs. Inspired by them, many researchers leverage the Siamese network structure in various kinds of fields. Bertinetto et al. [34] design a novel fully-convolutional Siamese network trained end-to-end on the ILSVRC15 dataset for object detection in video. In [35], the authors train a Siamese network to achieve human identification based on gait recognition. Siamese network is also the main technique in one-shot learning, Koch et al. [36] use a Siamese neural network for one-shot image recognition, which does not need a very large dataset.

III. SYSTEM DESCRIPTION

In this section, we introduce the hardware and architecture of our EmgAuth system and three main components, including data segmentation, the Siamese network and the unlocking simulation system. We also detail the novel method to make EmgAuth resilient to the rotation of the Myo armband.

A. System Architecture

EmgAuth consists of components deployed on a Myo armband and an Android smartphone. Myo armband is a device collecting EMG signal. It has eight channels, corresponding to eight sensors in different positions. Each channel has a sample rate of 200Hz and the data can be transferred by Bluetooth. Users could get the EMG signal easily by wearing it on the arm. We design two Android applications, one of them is for collecting and labelling data, the other one is an unlocking simulation application. The deep learning model trained in a GPU server is transplanted to an Android smartphone by TensorFlow Mobile [37].

Figure 1 presents the architecture of the system. The left side presents the offline model training. The data are collected by Myo armband and transferred to smartphone by Bluetooth in real-time. We label the different motions by clicking the corresponding buttons. After we get the labeled data, we conduct data segmentation to extract the valid EMG signal and make pairs to prepare training data. Data augmentation is also conducted to expand the dataset. Next, in the model training step, all pairs are fed into the neural network to train a Siamese convolution neural network.

The right side presents the online authentication. We transplant the trained Siamese neural network to an Android smartphone so we can evaluate the performance in real scenarios. Similar to most authentication systems, the first step that the user needs to do is enrollment. The enrollment phase needs users to do four sets of motions with only one time for each type. The system saves the EMG signal of these motions to the database and names it by the user's name and corresponding motion as identifiers. Next, when the user picks up the smartphone, the new EMG signal produced from the process will be compared with the previously stored signal and put into the model that we train in the offline phase. The Siamese neural network computes the distance of the input EMG pairs. If the output is less than the pre-defined threshold, the user will be successfully authorized and unlock the smartphone, otherwise the user will be rejected.

B. Data Segmentation

In most cases, data collected by sensors should be denoised. Various filters are applied to make the signal smoother and more stable. In the popular applications of EMG like gesture recognition, the signal should be roughly the same when different people do the same actions. While in the area of user authentication, the tiny difference is crucial. EMG is the external expression of the bioelectrical signal, which represents the structure of muscle and amount of muscle contraction. Figure 2 presents the raw EMG signal from Myo, showing five similar partial waves which present five times of picking up and putting down the smartphone. Those parts contained in five red dotted line boxes are valid data and we need to extract them as the EMG matrixes which will be described later. Here, we keep the raw signal without any filtering to maintain the uniqueness and use the convolution network to extract features.

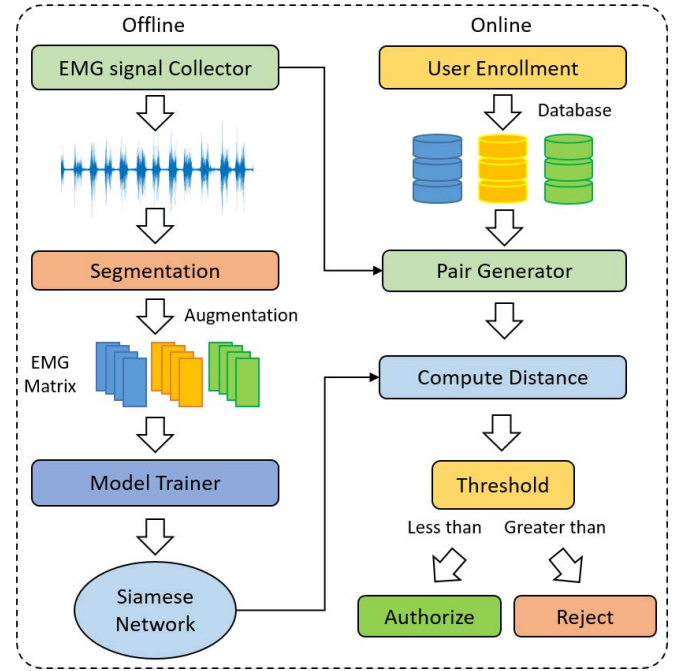


Fig. 1. System Architecture of EmgAuth.

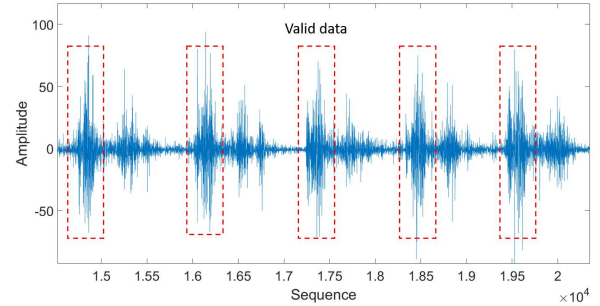


Fig. 2. Examples of the raw data from Myo armband.

We observe that the time from picking up the smartphone to watching the screen is generally no longer than two seconds. Hence, we set the valid action time to two seconds. As the sample rate is 200Hz and the number of channels is eight, we take an 8 by 400 signal matrix as one piece of basic training data. It is comparable to a picture with its resolution is 8 by 400 rather than 32 by 32 or 1024 by 1024. We divide these matrixes into different groups according to the corresponding people and make pairs as the following rules: the label of a pair that comes from different people is set to 0 while the same person is set to 1. The making pair algorithm is shown as Algorithm 1. The data after pairing are fed into the neural network for training.

C. Armband Rotation-Independence Method

One significant challenge that we need to address is the rotation problem of Myo armband. There are eight EMG sensors in the armband and each sensor corresponds to a specific skin area. We can not rotate the armband freely because the EMG signal is unique among different skin areas.

Algorithm 1 Making Training Pairs

```

1: Input Data
2:  $len = \text{length}(\text{Data})$ 
3: for each  $person \in \text{data}$  do
4:   for  $i \in \text{length}(\text{person})$  do
5:      $pair1 = \text{Data}[\text{person}][i] + \text{Data}[\text{person}][i + 1]$ 
6:      $y(pair1) = 1$ 
7:      $index = \text{Random}(0, len)$ 
8:      $diff\_person = (person + index) \% len$ 
9:      $pair2 = \text{Data}[\text{person}][i] + \text{Data}[diff\_person][i]$ 
10:     $y(pair2) = 0$ 
11:     $\text{FinalData.append}(pair1, pair2, y)$ 
12:   end for
13: end for
14: return  $\text{FinalData}$ 

```

However, fixing the position means we have to mark the position every time we wear it, which is inconvenient. To address this challenge, we propose a novel method based on the structure of Myo armband to make it rotation-independent. The Myo armband consists of eight rectangular sensors and they have unified sizes. Due to the fixed relative positions among these eight sensors, i.e. if the first sensor rotates to the position of the second, all the rest seven sensors will move in order and the last sensor will replace the position of the first sensor, we leverage the data augmentation technique from image classification tasks to expand our dataset. In our dataset, the eight channels correspond to eight sensors. Every time we roll the channels, a new dataset is created. The first picture of Figure 3 shows the result of rolling one channel. We get an eight times dataset until we roll a circle.

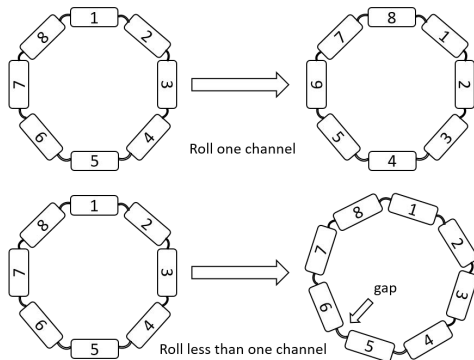


Fig. 3. Myo rotation sketch map.

From the second picture of Figure 3, we can see there is a gap between two sensors. When the user wears the armband, we can not guarantee the position is just one of the eight positions that we expanded and the sensor may cover the gap area when the rotation is less than one channel. However, the distance of this gap is much less than the width of a sensor, the impact of the gap on EmgAuth is in turn limited. From a mathematical point of view, rotation leads to infinite possibilities and we can only choose some representative

positions to train the model. There is a trade-off between the accuracy and the computation complexity.

In the task of image classification, flipping, cropping and scaling are the common data augmentation technique. After these operations, the label of an image does not change. Our labeling process is similar, after channel exchange, the expanded dataset still belongs to one person. In this way, the deep neural network can learn enough features and make reliable decisions, no matter how the user wears the Myo armband and whether the user rotates it or not.

D. Siamese Network

Siamese neural network is widely used in the tasks of verification and recognition. A standard CNN typically requires a large number of data to train a robust model. While we use our own dataset, the amount is limited. Besides, the users of authentication are dynamic as there are always new users joining and some users are leaving, we do not want to retrain the model when user group changes. Considering those requirements, we select the Siamese network as our deep learning model.

The Siamese network has two inputs and one output which value corresponds to the similarity between the two inputs. This network consists of two identical sub-networks with the same layers and weights. In addition, we add a layer to calculate the distance of the outputs of these two sub-networks. Here, we design a Convolutional Siamese Network and the architecture is shown in Figure 4. Our network architecture consists of three convolutional layers with different numbers of filters and one fully connected layer with 128 units.

Considering the Myo armband returns eight channels' signal at the same time, we need to mix them or find the relationship among them. A straightforward solution is using convolution. In the first convolutional layer, we set the kernel size to 8 by 1 to learn features among eight channels. We set the stride to one so that the first convolutional layer can focus on finding features among different channels. Since our input is 8 by 400 matrix, the output size of the first layer is 1 by 400, which achieves the combination of eight channels. Next, we set the kernel size to 1 by 3 to extract features during the process of picking up a smartphone and we get a size of 1 by 398 in this layer. To reduce the number of feature maps, we add a convolutional layer with 1 by 1 filters. These take all features from the previous layers into the next fully-connected layer. Besides, we add dropout layers after each convolutional layer to prevent overfitting. The dropout rate is increasing with the depth of the network from 0.1 to 0.2. As for activation, we use Rectified Linear Unit (Relu) for nonlinear transformation. Relu can reduce the likelihood of vanishing gradient, which results in faster learning.

In the last two layers of the network, we use a flatten layer and a fully-connected layer. The flatten layer is used to flat the output of previous convolutional layers so the features can be fed into a fully-connected layer. The fully connected layer takes every combination of features from the output of previous layers into account. Here we do not have a softmax

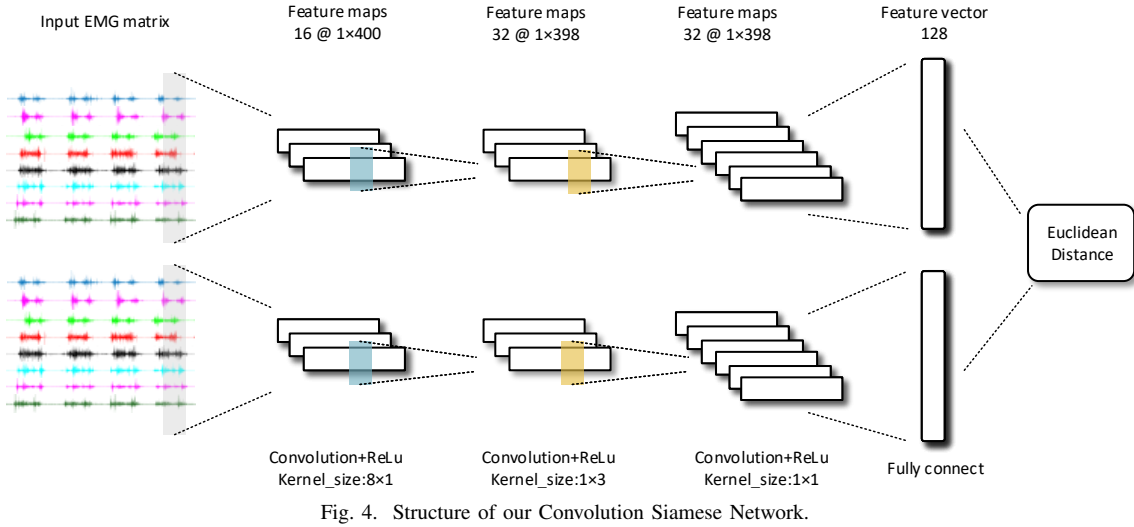


Fig. 4. Structure of our Convolution Siamese Network.

layer as usual since we prefer a vector that represents the original EMG input rather than a classification possibility. We define the number of units to 128, so we could get a 128-length vector as the map of the input EMG signal.

After defining the sub-network of our Siamese network, we need an extra layer to combine the outputs of them. We use the Euclidean distance to measure the difference between two output vectors from the last two fully connected layers. Loss function is used in supervised machine learning to minimize the differences between the predicted output of the model and the ground truth labels. In our task, we use the contrastive loss to train our model.

We train the network to make the distances of data from different participants are as far as possible, while from the same participant are as close as possible. In the long run, the network will learn to extract meaningful features and has the ability to distinguish different people. The input shape is (8, 400, 1), 8 means the data have eight different channels, 400 is the valid signal length, and 1 means each cell of the signal matrix has only one value.

E. Unlock Prototype System

We transfer the fine-tuned model into smartphones. The user interface of the unlocking simulation application is shown in Figure 6. In the first step, the user needs to input his or her name as the index of later EMG signal. In the second step, the user does four different actions to save their unique EMG signal in the database. We name these two steps as the enrollment stage. In the stage of authentication, the users open the simulation interface and take their smartphones as usual. The App saves the EMG signal from the Myo armband in real-time and implements segmentation in the time window of two seconds, so the processed signal can be fed into our model. The signal after segmentation is then paired with the previously stored four different types of EMG matrixes, respectively. These pairs are fed into the model, if one of the output is less than the threshold, the system considers the authentication successful; otherwise, the newcomer is rejected.

IV. EXPERIMENT AND EVALUATION

This section presents the experiments and implementation details of our EmgAuth system. We collect data from 40 participants and use this dataset to train a neural network. We then present the training progress and show the influence of related parameters. The performance of EmgAuth is tested in several experiments as well as the impact of different factors.

A. Dataset

In order to collect data to train a deep learning model, we invite 40 participants to help us build an EMG signal dataset. Participants include 28 males and 12 females, with the mean age of 24.6 and the range of 18 to 45, which considered to be typical user groups of smartphones.

The participants are required to wear the Myo armband in the forearm. The position of Myo armband is shown in Figure 7. This process is conducted in an academic environment and all participants are free from disturbing. An Android smartphone runs our data collection App and the EMG signal is recorded into csv files. We design a questionnaire to record

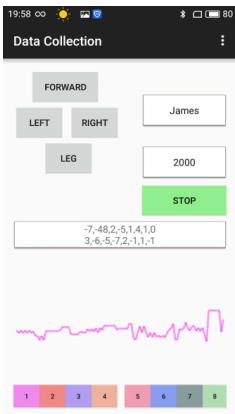


Fig. 5. Data collection App.

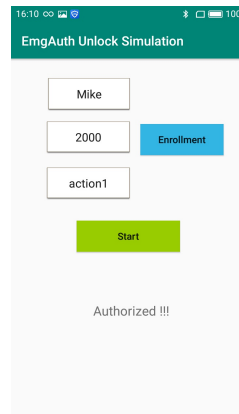


Fig. 6. Siamulation App.

smartphone positions when people are studying or working, the results are presented in Figure 11. As we can see, people usually put their smartphones on the left, forward and right, this helps us to design the data collect motions.



Fig. 7. The wearing position of Myo armband.

We require the participants to sit in front of a desk and put their smartphones on the desk. As shown in Figure 8, the motion they need to do is just picking up their smartphones and watching the screens as usual. Here, we design four scenes and the only difference among them is the location of the smartphone and hand. There are two initial positions, which are shown in the top two sub-figures. There is just one smartphone position corresponding to the former initial action while three positions for the latter initial action. The four smartphone positions are shown in Figure 11.

In position P1, P2, and P3, participants are required to repeat the motion for twenty times while position P4 for ten times. We decrease the times of the last scene because most people do not put the smartphone in such a position, as a result of our survey. The valid time starts from the user begins to move his or her hand to look at the screen and this time is normally less than two seconds. Therefore, we set the valid time to two seconds. To be specific, the sample rate is 200Hz and Myo armband has eight channels, so one piece of valid data is an 8×400 matrix. With 40 participants, we collect a dataset with 2800 valid data.

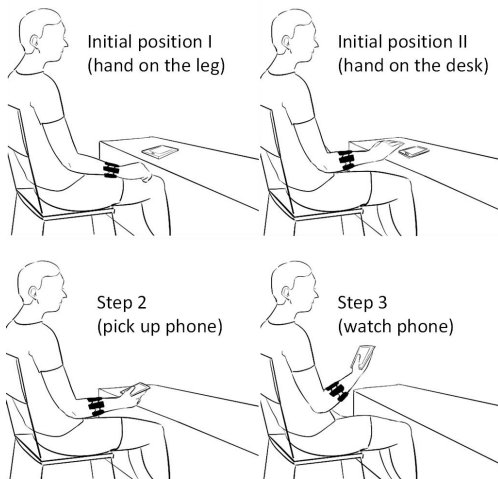


Fig. 8. Data collection steps.

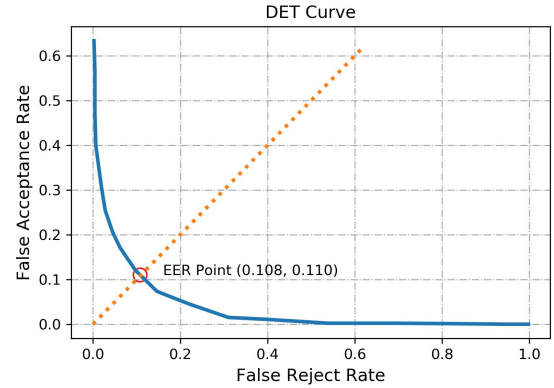


Fig. 9. DET curve with different thresholds

B. Model Training and System Evaluation

We implement our model using Keras, a Python-based deep learning platform. We train our model on a server machine equipped with an NVIDIA Tesla V100 GPU, 128GB memory, and an Intel Xeon E5 2560 processor. We use an Adam optimizer with a learning rate of 0.002 and a batch size of 32. For loss function, we choose contrastive loss rather than cross entropy loss. Contrastive loss runs over pairs of samples. During training, an EMG signal matrix pair is fed into the model with their ground truth relationship Y . If the two matrixes are similar, Y equals 0; otherwise, Y equals 1. The loss function is defined as (1), where d is the Euclidean distance between the two EMG feature vectors. The margin term is used to keep the loss within a valid range. For example, if two EMG signal matrixes in a pair are dissimilar, then their distance should be at least the value of margin, otherwise the loss will be 0.

$$Loss = (1 - Y)d^2 + Y \{ \max(0, \text{margin} - d) \}^2 \quad (1)$$

We evaluate the performance of EmgAuth using metrics that are commonly used in evaluating authentication systems. These metrics include accuracy, true acceptance rate (TAR), false acceptance rate (FAR) and false rejection rate (FRR). Equal error rate (EER) is also leveraged to find an appropriate classification threshold. With the threshold increases, FAR drops while FRR increases. EER is the point that FAR equals with FRR. Classifier has the best performance when the threshold corresponds to EER. Hence, we plot the Detection Error Tradeoff (DET) curve and the result is presented in Figure 9. The corresponding threshold of this EER point is 0.55, so we use it as the final threshold of our system.

We then study the effect of the different hyperparameters on the system performance, including the number of CNN layers, filter shape, learning rate, dropout rate, batch size, and the number of epochs. Table I shows the initial hyperparameters used in the evaluation section. We use the hold-out validation to check the performance and the ratio of train set and test set is 4:1. Also, we use accuracy as the only metrics of this evaluation.

TABLE I
TRAINING PARAMETERS

Parameter	Range	Initial value
Number of CNN layers	2 - 6	2
Filter number	16 - 64	16
Learning rate	0.001 - 0.005	0.001
Dropout rate	0.1 - 0.3	0.1
Batch size	16 - 256	16
Number of epochs	10 - 50	10

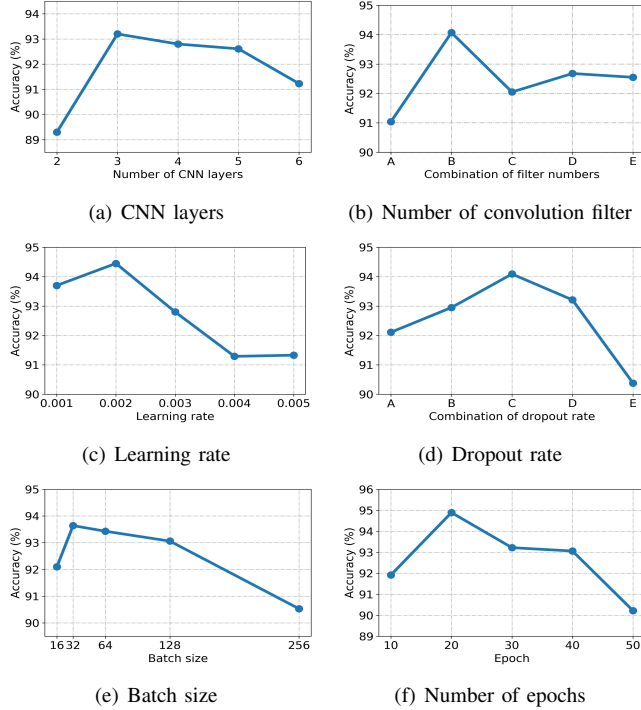


Fig. 10. Effect of different hyperparameters on authentication accuracy.

1) *Effect of the number of CNN layers*: Figure 10(a) presents the effect of changing the number of layers. The system reaches the highest accuracy when the number of layers is 3. Then with the number increases, the accuracy drops. Because the EMG signal is not as complicated as the images, too many layers may lead to overfitting, but only two layers cannot learn enough features.

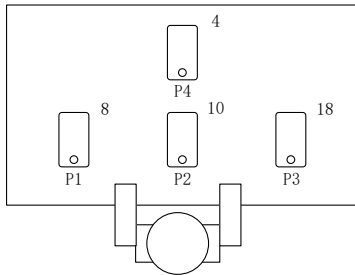


Fig. 11. Smartphone positions and corresponding frequencies.

2) *Effect of different filter shapes*: Figure 10(b) shows the effect of the number of convolution filters in each layer. A suitable number of convolution kernels can fully extract the features of the signal. We try five different number of filter combinations because there are three convolution layers. The A, B, C, D and E correspond to [16,16,16], [16,32,32], [16,32,64], [32,32,32] and [32,32,64], respectively. Combination B performs best, so we choose [16,32,32] as the filter number in the three convolution layers.

3) *Effect of different learning rates*: The learning rate is a hyperparameter that controls how much to change the model in response to the estimated error each time the model weights are updated. Too big or too small learning rates can both have negative affect on the learning result. Therefore, we use a series of values to choose the best one. Figure 10(c) shows that 0.002 is the most appropriate learning rate.

4) *Effect of different dropout rates*: Dropout is the simplest way to prevent neural network from overfitting. Considering that each convolution layer has a dropout layer, we try different dropout rate combinations and name them like Figure 10(b). Here, the A, B, C, D, and E correspond to [0.1,0.1,0.1], [0.1,0.2,0.2], [0.1,0.2,0.3], [0.2,0.2,0.2] and [0.2,0.2,0.3]. Figure 10(d) shows that there is an optimal combination for dropout rate at [0.1,0.2,0.2].

5) *Effect of different batch sizes*: Figure 10(e) shows how the batch size affects our system. The batch size is a hyperparameter that defines the number of samples to work through before updating the internal model parameters. Considering that the size of our dataset is not very large, a big batch size is not a good choice. From the figure, we can see that the accuracy reaches the highest when the batch size is 32.

6) *Effect of the number of epochs*: Figure 10(f) presents the effect of training epochs. The number of epochs is the number of complete passes through the training dataset. If the number of epochs is too high, the model is easily becoming overfitting as we have a small dataset. From the experiment, the model gets the best accuracy when we set the the number of epochs to 20.

7) *Effect of different distance functions*: As we mentioned above, we add a distance layer to combine the outputs of two identical sub-networks for measuring the similarity of them. Here, we evaluate three different functions including Manhattan distance [38], Euclidean distance and Cosine distance [39] to find a best option. To take a closer look at the system performance under different distance functions, we leverage the Receiver Operating Characteristic (ROC) curve in the study. ROC curve is an effective method to graphically reflect and compare the performance of different classifiers. Each point on an ROC curve corresponds to a certain detection threshold. Figure 12 presents the ROC curves of different distance functions, the Area Under the ROC Curve (AUC) of Euclidean distance is the biggest while Cosine distance performs worst. The AUC of Manhattan distance is between the other two distance functions. Therefore, we use Euclidean distance to measure the similarity of two feature vectors of the EMG signal.

TABLE II
HYPERPARAMETERS

Parameter	Value	Parameter	Value
Number of CNN layers	3	Filter number	[16, 32, 32]
Learning rate	0.002	Dropout rate	[0.1, 0.2, 0.2]
Batch size	32	Number of epoch	20

TABLE III
CROSS-VALIDATION RESULT

Train set	Test set	Accuracy	TAR	FAR	FRR
A, B, C, D	E	95.33%	95.83%	6.62%	4.67%
A, C, D, E	B	91.41%	90.22%	8.13%	9.78%
A, B, D, E	C	87.50%	84.70%	9.11%	15.29%
B, C, D, E	A	92.69%	95.77%	8.12%	4.23%
A, B, C, E	D	93.37%	92.50%	5.17%	7.50%
Average		92.06%	91.81%	7.43%	8.29%

After those experiments, we list the best parameters in Table II. We use our fine-tuned network to evaluate the performance of EmgAuth system. We divide our dataset into five subsets, which means each of them has 8 participants' EMG signal. These five subsets are marked as A, B, C, D, and E. Cross-validation is applied to handle the problem of insufficient data. We set the ratio of the train set and validation set to 4:1, as we can train the model for five times. The results of 5-fold cross-validation are listed in Table III. From the table, the average accuracy reaches 92.06% and the other three metrics are 91.81%, 7.43%, and 8.49%, respectively.

Except for the third group, the accuracies of the other four groups are more than 90%. In the third group, the accuracy is only 87.50%, severely lowering the average performance. The other three metrics are also terrible. We investigate the reasons behind it from the corresponding data. We find the EMG signal waves are different sometimes even they belong to the same motion of one person. Two reasons may lead to this situation. First, the user may not do the action in the sampling time, time drifting will lead to incorrect labeling, which misleads our system. Second, during the data collection step, some participants perform unnaturally, which may produce unqualified data and affect the performance of our system. These reasons also lead to the fluctuations of these metrics in the results of cross validation.

V. INFLUENCING FACTORS IN REAL SCENARIOS

The reliability of EmgAuth under various working conditions is critical for real-world deployment. In this section, we discuss the scenarios that EmgAuth might encounter in practice.

A. Impact of different enrollment actions

As we mentioned before, there are totally four different smartphone positions that we use during the data collection phase. However, in the real scenario, the position of the smartphone is random. In this section, we would like to investigate whether EmgAuth can handle other positions besides the above four locations. We invite five participants to do this experiment. First, they are required to put their smartphones

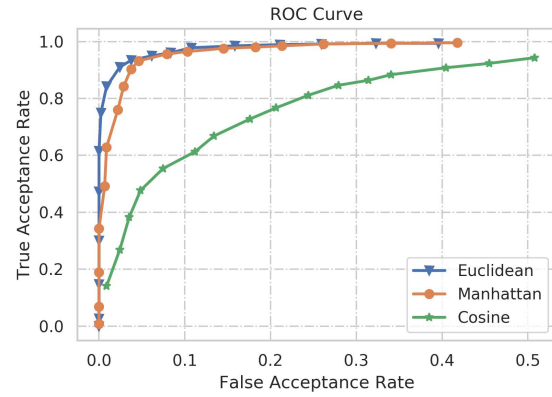


Fig. 12. ROC curves of different distance functions.

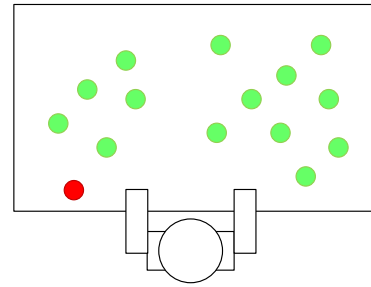


Fig. 13. Random positions and the authentication result.

as they want (except the above four positions) and we record the positions of them. Then they take the enrollment step to store their EMG signal. The results of the authentication are shown in Figure 13. The green marks are the positions that EmgAuth authenticates correctly while the reds are the positions that our system detects by mistake. There is just one position that EmgAuth fails and the accuracy reaches 93.33%. From this experiment, we can conclude that our deep learning model learns the features well and has an excellent possibility of generalization. We also find that no matter where the smartphone is put, the finger-level movements are similar when the user grabs his or her smartphone and it is the main reason why the system has a good generalization ability.

B. Impact of smartphone size and weight

To investigate the influence of types of smartphones, we invite four participants with four different smartphones. We design five sets of trials and every trial corresponds to a specific smartphone. In each trial, the four participants are asked to use the same smartphone to do the enrollment step and authentication step. Then in the next experiment, they change another smartphone at the same time. We do not set attackers in this experiment so TPR is the only metrics to measure the performance. The brands and parameters are listed in Table IV, as well as the experiment result. For the result of TAR, the overall performance is good except for Huawei Honor 10. The TAR of Honor is just 80% while the others are all more than 90%. The reason is that the size of Honor 10 is

TABLE IV
DIFFERENT SMARTPHONES AND THE CORRESPONDING TAR

Product	Weight(g)	Size(mm)	TAR
iPhone Xs	177	70.9×143.6×7.7	95%
iPhone 7	138	67.2×138.3×7.1	90%
Huawei Honor 10	230	85×177×7.65	80%
Xiaomi 8	175	74.8×154.9×7.6	90%

much bigger than the other three smartphones. A participant who gets used to the standard size smartphone is hard to adopt a bigger size one within a short time, which makes him does changeable motions in trials. In addition, we do not find the weight of the smartphone is an influencing factor of EmgAuth. Therefore, we conclude that EmgAuth is device-independent.

C. Performance on left hand

In real life, there are some left-handers inevitably, and this section is to evaluate whether EmgAuth can deal with this scenario. We invite two left-handers to help us do this experiment. The data collection process is identical with the right hand scenario except for the wearing position. They are required to do the enrollment first and repeat an action for five times during the authentication phase. From the experiment, our system fails just once among the ten times. Therefore, we can say that EmgAuth is also suitable for left-handers.

D. Performance when standing

To prove EmgAuth is robust, we recruit 10 extra participants to collect some EMG signal when they are standing and the smartphones are in the trouser pockets. They are required to take the smartphones out of their pockets and watch them. The duration of this action is also two seconds and every time the assistant says "start", the participant repeats the action five times. In this way, we get a tiny dataset for testing whether EmgAuth can unlock the smartphone when people are standing rather than sitting. First, we split the dataset for enrollment data and authentication data and the ratio of them is 1:4. Then we make pairs and put these pairs into the EmgAuth system. The TAR is 98.43% in the trial, which means EmgAuth can deal with the standing scenario. The reason why the TAR is higher than sitting scenarios is that comparing with four different actions of sitting, the action of standing is straightforward, which is easier for EmgAuth to find features and make the right decision.

E. Unlocking speed

In this section, we test the speed of EmgAuth both on the server and the Android smartphone. The whole unlocking process includes four steps: loading stored EMG metrics, data segmentation, making pairs and model calculation. Among the above four steps, loading data consumes the most time because it is an I/O operation. Therefore, we load the EMG metrics in memory after a user finishes the enrollment phase to accelerate the process. Then we test the time both on the server and a Xiaomi 8 smartphone. When the process runs in a server machine, the authentication latency is about 0.048s.

Due to the limited computing resource of the smartphone, the authentication latency of the simulation App run in a Xiaomi 8 is about 0.16s, which fulfills the requirement of real-time unlocking.

VI. CONCLUSION

We present EmgAuth, an EMG-based smartphone unlocking system, which leverages EMG signal and Siamese network to unlock smartphones. In particular, when training the Siamese network, we design a special data augmentation technique to make the system resilient to the rotation of the armband, which lets the system free of calibration. We conduct an experimentation with 40 participants for collecting a dataset and design a convolutional Siamese network for the EMG signal. Our system can authenticate users effectively with an average TAR of 91.81% while keeping the average FAR of 7.43%. Extensive experiments are conducted to demonstrate the usability of EmgAuth for smartphones with different sizes and at different locations, as well as users with different postures. To estimate the speed we measure the latency both on a server and an Android smartphone, the results fulfill the requirement of real-time smartphone unlocking.

Although the experimental results are promising, the limitation of EmgAuth also exists. First, our system may not work well in the humid environment as the EMG signal becomes unstable if the skin surface is wet. Second, the application scenarios are limited currently which just include sitting and standing. In the future, we will recruit more people to build a bigger dataset and evaluate the performance of EmgAuth in the long-term stability study. In particular, we will design a smaller EMG sensor to make the system easier to use and discover more application scenarios.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (61572060, 61772060, 61728210, 61572217, 61572218 and 61602024) and CERNET Innovation Project (NGII20170315).

REFERENCES

- [1] Y. Wang, K. Streff, and S. Raman, "Smartphone security challenges," *Computer*, vol. 45, no. 12, pp. 52–58, 2012.
- [2] J. Daugman, "How iris recognition works," in *The essential guide to image processing*. Elsevier, 2009, pp. 715–739.
- [3] N. L. Clarke and S. M. Furnell, "Authentication of users on mobile telephones—a survey of attitudes and practices," *Computers & Security*, vol. 24, no. 7, pp. 519–527, 2005.
- [4] Get your fingerprint. [Online]. Available: <https://www.instructables.com/id/Get-your-fingerprint/>
- [5] Fool apple faceid. [Online]. Available: <https://www.wired.co.uk/article/hackers-trick-apple-iphone-x-face-id-3d-mask-security>
- [6] S. K. Sahoo, T. Choubisa, and S. M. Prasanna, "Multimodal biometric person authentication: A review," *IETE Technical Review*, vol. 29, no. 1, pp. 54–75, 2012.
- [7] H. Yamaba, S. Nagatomo, K. Aburada, S. Kubota, T. Katayama, M. Park, and N. Okazaki, "An authentication method independent of tap operation on the touchscreen of a mobile device," *Journal of Robotics, Networking and Artificial Life*, vol. 2, no. 1, pp. 60–63, 2015.
- [8] H. Yamaba, A. Kurogi, S.-I. Kubota, T. Katayama, M. Park, and N. Okazaki, "Evaluation of feature values of surface electromyograms for user authentication on mobile devices," *Artificial Life and Robotics*, vol. 22, no. 1, pp. 108–112, 2017.

- [9] M. Sathiyarayanan and S. Rajan, "Myo armband for physiotherapy healthcare: A case study using gesture recognition application," in *2016 8th International Conference on Communication Systems and Networks (COMSNETS)*. IEEE, 2016, pp. 1–6.
- [10] Y. Xu, C. Yang, P. Liang, L. Zhao, and Z. Li, "Development of a hybrid motion capture method using myo armband with application to teleoperation," in *2016 IEEE International Conference on Mechatronics and Automation*. IEEE, 2016, pp. 1179–1184.
- [11] A. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 19, no. 4, pp. 302–314, 1997.
- [12] 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.
- [13] S. P. Banerjee and D. L. Woodard, "Biometric authentication and identification using keystroke dynamics: A survey," *Journal of Pattern Recognition Research*, vol. 7, no. 1, pp. 116–139, 2012.
- [14] F. Monrose and A. D. Rubin, "Keystroke dynamics as a biometric for authentication," *Future Generation computer systems*, vol. 16, no. 4, pp. 351–359, 2000.
- [15] D. Gafurov, K. Helkala, and T. Söndrol, "Biometric gait authentication using accelerometer sensor," *JCP*, vol. 1, no. 7, pp. 51–59, 2006.
- [16] B. Shrestha, N. Saxena, and J. Harrison, "Wave-to-access: Protecting sensitive mobile device services via a hand waving gesture," in *International Conference on Cryptology and Network Security*. Springer, 2013, pp. 199–217.
- [17] N. Sae-Bae and N. Memon, "Online signature verification on mobile devices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 6, pp. 933–947, 2014.
- [18] R. Merletti, A. Botter, C. Cescon, M. A. Minetto, and T. M. Vieira, "Advances in surface emg: recent progress in clinical research applications," *Critical Reviews™ in Biomedical Engineering*, vol. 38, no. 4, 2010.
- [19] E. J. Rechy-Ramirez and H. Hu, "Stages for developing control systems using emg and eeg signals: A survey," *School of Computer Science and Electronic Engineering, University of Essex*, pp. 1744–8050, 2011.
- [20] M. R. Ahsan, M. I. Ibrahimy, O. O. Khalifa *et al.*, "Emg signal classification for human computer interaction: a review," *European Journal of Scientific Research*, vol. 33, no. 3, pp. 480–501, 2009.
- [21] X. Zhang, X. Chen, W.-h. Wang, J.-h. Yang, V. Lantz, and K.-q. Wang, "Hand gesture recognition and virtual game control based on 3d accelerometer and emg sensors," in *Proceedings of the 14th international conference on Intelligent user interfaces*. ACM, 2009, pp. 401–406.
- [22] K. Kiguchi and Y. Hayashi, "An emg-based control for an upper-limb power-assist exoskeleton robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, pp. 1064–1071, 2012.
- [23] M. R. Ahsan, M. I. Ibrahimy, and O. O. Khalifa, "Electromyography (emg) signal based hand gesture recognition using artificial neural network (ann)," in *2011 4th International Conference on Mechatronics (ICOM)*. IEEE, 2011, pp. 1–6.
- [24] J. W. Yoo, D. R. Lee, Y. J. Sim, J. H. You, and C. J. Kim, "Effects of innovative virtual reality game and emg biofeedback on neuromotor control in cerebral palsy," *Bio-medical materials and engineering*, vol. 24, no. 6, pp. 3613–8, 2014.
- [25] V. Becker, P. Oldrati, L. Barrios, and G. Sörös, "Touchsense: classifying finger touches and measuring their force with an electromyography armband," in *Proceedings of the 2018 ACM International Symposium on Wearable Computers*. ACM, 2018, pp. 1–8.
- [26] M. E. Benalcázar, A. G. Jaramillo, A. Zea, A. Páez, V. H. Andaluz *et al.*, "Hand gesture recognition using machine learning and the myo armband," in *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, 2017, pp. 1040–1044.
- [27] J. G. Abreu, J. M. Teixeira, L. S. Figueiredo, and V. Teichrieb, "Evaluating sign language recognition using the myo armband," in *2016 XVIII Symposium on Virtual and Augmented Reality (SVR)*. IEEE, 2016, pp. 64–70.
- [28] S. Shin, J. Jung, and Y. T. Kim, "A study of an emg-based authentication algorithm using an artificial neural network," in *2017 IEEE SENSORS*. IEEE, 2017, pp. 1–3.
- [29] R. Shioji, S.-i. Ito, M. Ito, and M. Fukumi, "Personal authentication and hand motion recognition based on wrist emg analysis by a convolutional neural network," in *2018 IEEE International Conference on Internet of Things and Intelligence System (IOTAIS)*. IEEE, 2018, pp. 184–188.
- [30] G.-S. Xie, X.-Y. Zhang, W. Yang, M. Xu, S. Yan, and C.-L. Liu, "Lg-cnn: From local parts to global discrimination for fine-grained recognition," *Pattern Recognition*, vol. 71, pp. 118–131, 2017.
- [31] M. Xu, H. Fang, P. Lv, L. Cui, S. Zhang, and B. Zhou, "D-stc: Deep learning with spatio-temporal constraints for train drivers detection from videos," *Pattern Recognition Letters*, 2017.
- [32] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.
- [33] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a siamese time delay neural network," in *Advances in neural information processing systems*, 1994, pp. 737–744.
- [34] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *European conference on computer vision*. Springer, 2016, pp. 850–865.
- [35] C. Zhang, W. Liu, H. Ma, and H. Fu, "Siamese neural network based gait recognition for human identification," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 2832–2836.
- [36] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, vol. 2, 2015.
- [37] Tensorflow mobile. [Online]. Available: https://www.tensorflow.org/lite/guide/get_started
- [38] L. Greche, M. Jazouli, N. Es-Sbai, A. Majda, and A. Zarghili, "Comparison between euclidean and manhattan distance measure for facial expressions classification," in *2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS)*. IEEE, 2017, pp. 1–4.
- [39] K. K. George, C. S. Kumar, and A. Panda, "Cosine distance features for robust speaker verification," in *Sixteenth annual conference of the international speech communication association*, 2015.