M2FRED Analysis Using MobileNet and Siamese Neural Network

Riskie Annisa * and Benfano Soewito

Computer Science Department, BINUS Graduate Program Master of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia; Email: bsoewito@binus.edu (B.S.)
*Correspondence: riskie.annisa@binus.ac.id (R.A.)

***Abstract*—Mobile face recognition has become increasingly important, especially during the COVID-19 pandemic when the use of masks has become ubiquitous. The area that can be analyzed for face recognition is narrowed down only to the periocular area. This resulted in many studies of face recognition in the periocular area which require appropriate datasets, one of which is M2FRED, introduced by University of Salerno. Previous research on M2FRED using supervised learning algorithms such as Support Vector Machine (SVM), Multilayer Perceptron (MLP), Random Forest (RF), and Decision Tree (DT) shows promising results on M2FRED with accuracy at 95.4% using MLP. However, study on M2FRED using deep learning models has yet to be done. In this paper, we compare the performance of MobileNet and Siamese neural networks on M2FRED, a face dataset specifically designed for mobile face recognition that contains videos of 43 subjects with and without masks taken in uncontrolled environments using mobile devices. We employ this range of M2FRED to evaluate the performance of MobileNet and Siamese neural networks in handling challenges like face masks, various lighting conditions, and limited computational resources. MobileNet outperformed the Siamese neural network in every aspect with an average accuracy score of 99.77% for overall performance (99.85%), mask usage scenarios (100%), and lighting context (99.72% for indoor evaluation and 99.52% for outdoor evaluation). With its simple architecture, MobileNet also surpassed Siamese neural networks in terms of complexity.**

***Keywords*—face recognition, mobile, MobileNet, Siamese neural networks, M2FRED**

I. Introduction

Mobile face recognition has become increasingly popular due to its numerous advantages over traditional face recognition systems that typically run on desktop or server-class hardware [1, 2]. Some of the advantages are that it is easy, convenient, and enables real-time recognition. These benefits allow the application of mobile face recognition to be prevalent across several domains, for example, security, healthcare, law enforcement, entertainment, education, and marketing [3].

However, several challenges commonly emanate when implementing face recognition on mobile devices, such as limited computational resources for processing [4], varying lighting conditions [5], and changes in facial appearance due to aging [6–8], makeup [9–11], expression [11, 12], mask usage [9, 13], low-resolution image [13], or facial hair [11].

Moreover, anti-COVID regulations that started in 2020 globally made mask applications for any activity mandatory in almost any nation. This policy affected face recognition applications in many ways. The portion of the face usable for recognition had become restricted, leaving only the periocular area, therefore decreasing the accuracy in face recognition. Studies on mask detection and face-masked recognition are increasing rapidly to deal with this situation not only in algorithms but also in datasets. One of them was the Mobile Masked Face Recognition through Periocular Dynamics Analysis (M2FRED) dataset that was introduced by Cimmino *et al.* [14] to study the periocular dynamic influence on face recognition.

In addition, several popular deep learning algorithms have been developed to overcome those challenges, such as the Siamese neural networks and MobileNet. Siamese neural networks are a type of neural networks that can be very helpful in tasks where it is necessary to assess how similar or dissimilar two inputs are to each other. Therefore, it is commonly used to address challenges such as slight variations in the appearance [15] of the face due to various factors including aging, makeup, facial hair, and lighting conditions. Meanwhile, challenge in limited resources was emphasized in [16] by using an architecture called MobileNet which was specifically designed to be computationally efficient and small in size [17]. This is accomplished by using a combination of depth-wise separable convolutions followed by a pointwise convolution to reduce the number of computations required while maintaining accuracy. Siamese neural network and MobileNet can prove that deep learning algorithms can be modified to resolve frequently encountered obstacles in applying face recognition in mobile devices.

To gain a more comprehensive understanding of the strengths and weaknesses of MobileNet and Siamese neural network in dealing with challenges such as lighting conditions, mask usage, and device limitations, we utilized the M2FRED dataset. The dataset includes a set of short videos captured using mobile devices in both indoor and

Manuscript received May 30, 2023; revised July 20, 2023; accepted August 9, 2023; published December 1, 2023.

outdoor environments, with subjects wearing and not wearing masks [12].

One significant contribution of this study is the exploration of M2FRED dataset using deep learning techniques, specifically MobileNet and Siamese neural network. To the best of our knowledge, there has been a notable research gap in harnessing the potential of deep learning approaches, specifically MobileNet and Siamese neural network, for M2FRED analysis. Previous studies in this domain have predominantly relied on traditional machine learning models, including Multilayer Perceptron (MLP), Decision Tree (DT), Random Forest (RF), and Support Vector Machine (SVM). Therefore, our research presents a novel approach by utilizing deep learning models to address the challenges posed by M2FRED dataset.

II. Literature Review

This section delivers an overview of the literature regarding datasets and the architecture used for mobile face recognition, particularly to overcome challenges in mask utilization, light settings, and restricted computing resources.

A. Related Theories

Siamese neural networks are especially useful in applications that require determining the similarity or dissimilarity between two inputs. One unique feature of Siamese neural networks is that the network consists of two identical sub-networks that share the same architecture and weights. Each sub-network takes one input (in this case, data triplets) and processes it through several layers of convolutional, pooling, and activation functions to produce a feature vector or embedding. The output embeddings from each of the sub-networks are then compared using a distance metric, such as Euclidean distance or cosine similarity, to determine the similarity between them. Siamese neural networks are often used for tasks such as image comparison, object tracking, and facial recognition. They are especially useful in situations where there is limited training data available or where the input data is noisy or ambiguous. Siamese neural network can be simply described in Fig. 1 below.

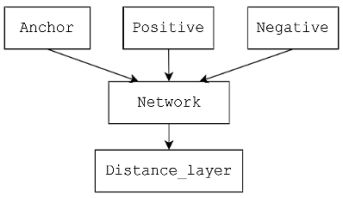

Figure 1. Data flow on Siamese neural network.

On the other hand, Siamese neural networks can require significant computational resources, particularly during training, due to their large number of parameters and the need to compute distances between pairs of images.

In the meantime, MobileNet architecture integrates the concept of depth-wise separable convolution, which splits the convolution operation into two separate operations: depth-wise convolution and point-wise convolution. The depth-wise convolution applies a single filter per input channel, while the point-wise convolution applies a 1×1 convolution to combine the results of the depth-wise convolution across channels. Depiction of these two kinds of convolutions is in Fig. 2.

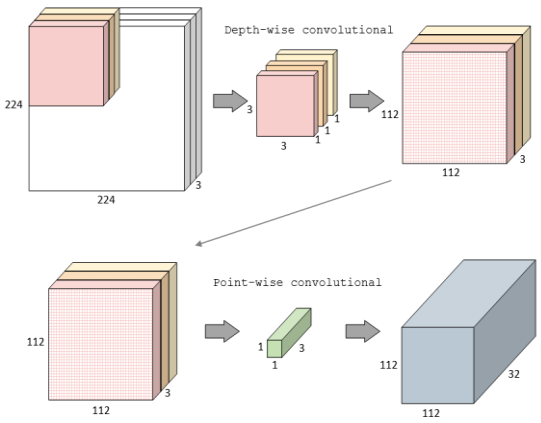

Figure 2. Depth-wise and point-wise convolutions on MobileNet.

This approach reduces the computational requirements and number of parameters needed for the convolutional layer, making it more efficient for applications on mobile and embedded devices.

B. Previous Researches

Siamese neural networks were initially introduced by Bromley *et al.* [18] to address the task of signature verification, treating it as an image-matching problem. Later, Taigman *et al.* [19] tried to implement the Siamese neural network to handle a slightly different task from signature verification, namely face recognition. Wolf *et al.* [19] proposed DeepFace-Siamese, a deep neural network architecture for face verification that uses a Siamese neural network to learn a similarity metric between pairs of face images. The network was trained on a large-scale dataset of face images (SFC) and then tested on LFW with an accuracy as high as 97.35%. Another research done by Song and Ji [20] can improve face recognition accuracy under non-restricted conditions, as demonstrated by simulation exploration using standard face datasets such as CASIA-WebFace [21], Yale-B [22], and LFW [23]. Further, Aufar and Sitanggang [24] tested implementing a Siamese neural network on a mobile application built using the Kivy framework using the LFW image dataset and achieved a classification accuracy rate of 98%.

In the meantime, MobileNet was proven to be well-suited for computer vision tasks for mobile devices such as object detection, fine-grain classification, facial attributes, and large-scale geo-localization [16]. A comparative study on four different CNN architectures: VGG16, Inception-V3, ResNet50, and MobileNet was conducted by Ahmed *et al.* [25], resulting in the finding that MobileNet performed better in test accuracy amongst all. Subsequently, an attempt to implement MobileNet on mobile attendance systems has been managed by

Brown [26] by also utilizing an IoT toolkit, namely OpenVINO. Evaluation is carried out by considering pose angles, and MobileNet can achieve a maximum accuracy of 100% at angles of +25º, 15º, –15º, and –25º, while the lowest accuracy is 89.25% at an angle of +60º.

As for datasets, according to the file type, datasets can be categorized into two types, image-based face datasets, and video-based face datasets. However, the availability of face datasets focusing on mask usage is relatively limited. For image-based face datasets, Ge *et al*. [27] introduced MAFA which contains only masked face images. Another mask detection dataset, MOXA [28] and MFDD [29], consists of both 'masked' and 'no mask' labels. A more comprehensive dataset on mask detection is FMD [25], which also has 'incorrect mask' label aside from 'mask' and 'no mask'. There are also RMFRD and SMFRD that were introduced by Huang *et al*. [29]. RMFRD was collected from public images available on the internet, with the corresponding masked faces. In the meantime, SMFRD was using the existing public face dataset with artificial masks [30].

Meanwhile for the video-based face datasets that consider mask usage, there is Youtube Faces Database (YFDB) [31] that is collected from Youtube. YFDB was created to investigate the challenges of face recognition in uncontrolled video settings. As for video-based face dataset in controlled settings, there is XM2VTS [32] that is ideal to use in an environment where it can be reasonably expected that the client will be cooperative. Yet video-based face dataset that focuses on mask usage is very limited. One we can find is M2FRED, which is a dataset that tries to answer the need for facial recognition during the pandemic era due to COVID-19. This dataset contains videos of the subject speaking short sentences both using and not using mask. However, there is only a small number of research on mobile face recognition utilizing M2FRED. In [14], M2FRED was tested using MLP, SVM, DT, and RF, resulting in better performance compared to XM2VTS. Summary of the previous researches can be seen in Table I.

The literature review reveals the noteworthy development of Siamese Neural Network and MobileNet, which have attained state-of-the-art status in face recognition. Therefore, based on their exceptional performance, the author selected these two models for the experiments conducted in this research paper.

Subsequently, from Table I, we notice that more exploration has yet to be done on M2FRED despite its potential. Therefore, we try to cultivate more on M2FRED using state-of-the-art deep learning models which are MobileNet and Siamese neural networks.

TABLE I. SUMMARY OF THE LITERATURE REVIEW

Ref.	Model	Dataset	Description	Features
Bromley *et al.* [18]	Siamese Neural Network	2190 up to 4380 signatures	Research done on signature verification with accuracy 97%	"pud" feature, x and y position, speed at each point, centripetal acceleration, tangential acceleration, direction cosine of the tangent, direction sine of the tangent, cosine of the local curvature, sine of the local curvature
Taigman *et al.* [19]	Siamese (DeepFace-Siamese)	Trained on SFC, tested on LFW	Accuracy is 97.35%	Simple edges, texture
Aufar Sitanggang [24]	Siamese CNN	LFW	Classification accuracy is 98%	N / A
Ahmed *et al.* [25]	MobileNet, VGG16, Inception-V3, ResNet50	130 images for each 10 celebrities obtained from internet	MobileNet performed best accuracy 84%	VGG16 was fine tuned to have 13.504.778 parameters, MobileNet was also fine tuned to 3.217.226 parameters, as for AlexNet N/A
Brown [26]	MobileNet	FERET B-series Face Dataset, CrowdHuman and Classroom Dataset (collected by researchers)	Experiment done by using IoT toolkit on mobile attendance system. Best MobileNet accuracy was 100%	N/A
Cimmino *et al.* [14]	MLP, SVM, DT, RF	M2FRED and XM2VTS	The quality of M2FRED is equivalent to XM2VTS despite M2FRED was taken under uncontrolled environments, and conducted by involving mask usage	Geometric features, facial landmarks, texture patterns, temporal information

III. MATERIALS AND METHODS

This section discusses datasets, processes, and the architecture used for the research. The aim of this study is to determine which method is more suitable to be implemented in a mobile-based face recognition system. To achieve this goal, we conducted experiments using the M2FRED dataset and two different algorithms, namely MobileNet and Siamese neural networks.

A. Dataset

This research employed the Mobile Masked Recognition Through Periocular Dynamics Analysis (M2FRED) dataset [14], which consists of short videos of 43 subjects of various genders and ages uttering predetermined sentences or phrases. The videos were captured in an uncontrolled environment with four different conditions: indoors without a mask, indoors with a mask, outdoors without a mask, and outdoors with a mask as seen in Fig. 3.

Figure 3. Samples of Subject 036 from M2FRED.

The structure of the dataset is as follows:

1) It consists of 43 subjects with ID numbers from 000 to 042.
2) The dataset is stored in two directories for each subject: one for videos without masks (i.e., 000_0), and the other for videos with masks (i.e., 000_1).
3) There are a total of 16 videos for each subject:
   - 4 masked videos taken indoors.
   - 4 masked videos taken outdoors.
   - 4 non-masked videos taken indoors.
   - 4 non-masked videos taken outdoors.

The best result for face recognition on the M2FRED dataset was achieved by using MLP with an accuracy of 95.4% [14].

This dataset was originally created to investigate face recognition during the pandemic, when masks are commonly used, leading to a shift in focus to the periocular area for facial recognition. But with its characteristics, M2FRED can also be used to analyze deep learning performance under various lighting conditions.

B. *Preprocessing*

We analyzed the M2FRED dataset thoroughly, and we found some inconsistencies in labeling. There is a data labeling mistake on Subject 15. Two videos (masked and indoor) of Subject 15 were mistakenly labeled as Subject 14. This causes the data on Subject 14 to be incomplete, as can be seen in Fig. 4. There is also one video with too short duration (only 1s) on Subject 22 (file 022_1_0_2.avi), resulting in only a few frames that can be extracted from that video.

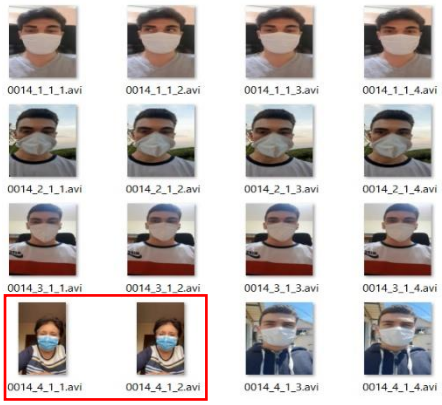

Figure 4. Samples of Subject 15 mistakenly labeled as subject 14.

There is also incomplete data for Subject 002, which only consists of 3 videos for each sub-directory when there should be 4 videos.

However, these inconsistencies do not significantly affect the total number of frames that can be generated from each subject considering that each subject does not only have one video sample. Therefore, we use a fixed number of frames to cope with these inconsistencies.

Tables II and III give the idea of the preprocessing result.

TABLE II. FRAME EXTRACTION FOR MOBILENET

Scheme	Frame of each subject		Total Num of Frames	
	Training	Testing	Training	Testing
MobileNet (overall)	138	32	5,934	1,376
MobileNet (mask)	138	13	4,644	559
MobileNet (indoor)	138	13	4,644	559
MobileNet (outdoor)	138	13	4,644	559

TABLE III. FRAME EXTRACTION SIAMESE NEURAL NETWORK

Scheme	Num of Subdirectories (each has 135 frames)		Total Num of Frames	
	Training	Testing	Training	Testing
Siamese (overall)	34	9	4,590	1.215
Siamese (mask)	34	9	4,590	1.215
Siamese (indoor)	34	9	4,590	1.215
Siamese (outdoor)	34	9	4,590	1.215

1) *Preprocessing result of mobileNet*

For MobileNet, frames are extracted from videos of each subject using the OpenCV (CV2) library. An extraction rate of 20 frames is used, meaning every 20th frame is extracted. For each subject, the first 6 extracted frames are used as training data, and the remaining frames are used as testing data.

The total number of frames collected for training data is 7,662 frames, and for testing data, it is 2,914 frames. However, it was observed that some subjects had fewer frames than others, with a minimum of 138 frames for training, 32 frames for testing the overall performance, and 13 frames for mask and lighting evaluation. Therefore, to ensure uniformity across all subjects, the fixed number of 138 frames per subject was used for both training and testing data.

After frame extraction, the frames are converted into arrays using Image Data Generator, allowing them to be read using TensorFlow. During this stage, data augmentation is applied, including rotation, horizontal and vertical shifting, shearing, zooming in, and horizontal flipping. Additionally, a validation set is created by taking 20% of the training data for each subject.

This data pipeline ensures that the extracted frames are processed, augmented, and divided into training and validation sets for each subject, making the data ready for training a MobileNet-based model with TensorFlow's Keras API.

2) Preprocessing result of Siamese neural network

A different approach is taken for data preprocessing in the Siamese neural network. Initially, videos are extracted using an extraction rate of 20 frames per video. After extraction, the directories are logically split into training sets and testing sets with an 80:20 ratio, resulting in 34 directories for training and 9 directories for testing. Each directory contains 135 images.

From these 135 images in each directory, triplets are created to form training and testing pairs. Each triplet is composed of an anchor, positive, and negative image, as illustrated in Fig. 5. The anchor image represents the reference image, the positive image is another image of the same identity as the anchor, and the negative image is an image of a different identity.

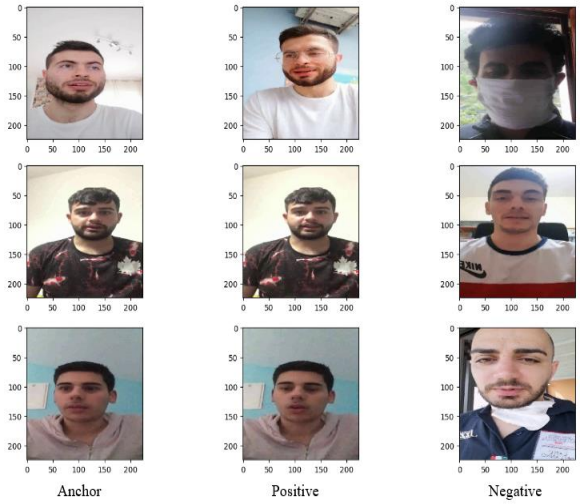

Figure 5. Triplets sample of M2FRED.

Using this setup, we generate a total of 4,590 training pairs and 1,215 testing pairs. Each pair consists of an anchor image paired with either a positive or a negative image.

In the Siamese neural network, these pairs of triplets (anchor + positive) and (anchor + negative) are fed into the model, producing vectors for each pair. By calculating the distance between these vectors, we can obtain a similarity score between the anchor, positive, and negative images. A specific threshold (e.g., 1.3) is used to classify whether a pair is considered a positive match, or a negative match based on the similarity score.

C. Experiments

The Experiments section of this research focuses on the utilization of two key deep learning models: MobileNet and Siamese Neural Network. These models are employed to investigate and analyze specific research objectives.

In the experiments, the MobileNet model is utilized for its efficient and lightweight architecture, which is particularly well-suited for mobile face recognition tasks. The Siamese Neural Network, on the other hand, is employed to capture and analyze the similarity between pairs of face images. By comparing these two models, we aim to determine their respective strengths and limitations in addressing the research problem.

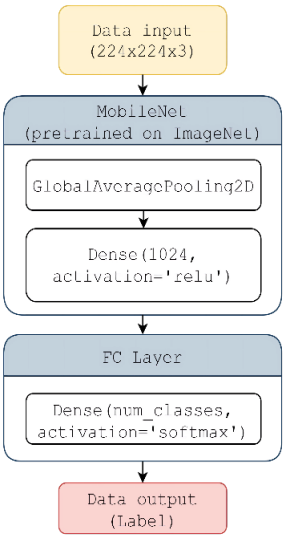

Figure 6. MobileNet architecture.

1) MobileNet

This study adopted the MobileNet architecture for face recognition tasks in resource-constrained environments such as mobile devices and embedded systems. To implement MobileNet, we use the Keras deep learning framework with a pre-trained MobileNet model, discard the top layer, and add two new layers to the model: a global average pooling layer and a dense layer with 1024 units and a ReLU activation function. The model was trained using the Adam optimizer with a learning rate of 0.001 and a batch size of 32 for 10 epochs. The softmax function was used as the activation function in the output layer to classify the images. Simply put, the representation of the model is as in Fig. 6, meanwhile the summary of the model is represented in Fig. 7.

```
Model: "MobileNet for M2FRED"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 mobilenet_1.00_224 (Functio  (None, 7, 7, 1024)       3228864
 nal)

 global_average_pooling2d (G  (None, 1024)             0
 lobalAveragePooling2D)

 dense (Dense)               (None, 1024)              1049600

 dense_1 (Dense)             (None, 43)                44075

=================================================================
Total params: 4,322,539
Trainable params: 4,300,651
Non-trainable params: 21,888
```

Figure 7. MobileNet summary.

2) Siamese neural network

For the Siamese neural network implementation, we first create sets of triplets that consist of an anchor image, a positive image (belonging to the same person as the anchor), and a negative image (belonging to a different person than the anchor).

Then the Siamese neural network architecture is defined, which consists of two identical CNN models, each consisting of a pretrained model, namely the Xception model, followed by dense layers and the L2 normalization layer. The two sub-networks share the same weights and

are fed with a pair of images as inputs. A Distance Layer was then applied to calculate the distance between (anchor, positive) pairs and (anchor, negative) pairs. Fig. 8 represents the Siamese neural network's architecture used in this research.

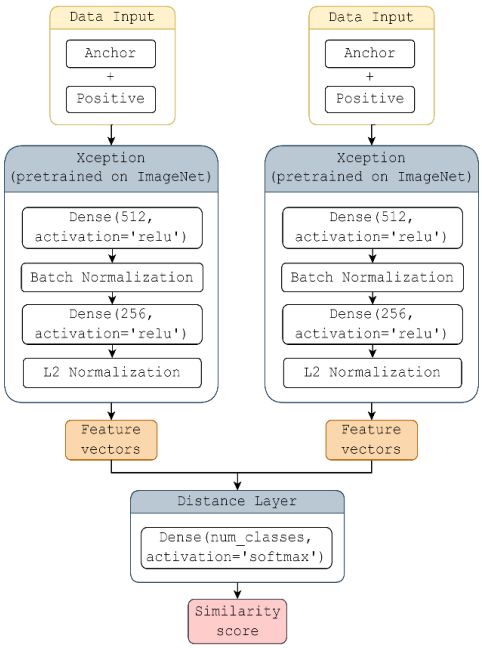

Figure 8. Siamese neural network architecture.

Concurrently, the model's overview is illustrated in Fig. 9.

```
Model: "Siamese_Network"
__________________________________________________________________
 Layer (type)                Output Shape              Param #
==================================================================
 Anchor_Input (InputLayer)   [(None, 224, 224, 3       0
                             )]

 Positive_Input (InputLayer) [(None, 224, 224, 3       0
                             )]

 Negative_Input (InputLayer) [(None, 224, 224, 3       0
                             )]

 Encode_Model (Sequential)   (None, 256)               22043944


 distance_layer (DistanceLayer)  ((None,),             0
                                 (None,))


==================================================================
Total params: 22,043,944
Trainable params: 9,583,800
Non-trainable params: 12,460,144
```

Figure 9. Siamese neural network architecture.

The subnetworks were trained to recognize the similarity between two images. The model was trained using the Adam optimizer with a learning rate of 0.001. We trained the model for 10 epochs with a batch size of 32 and a similarity threshold of 1.3. The purpose of the Siamese neural network was to learn a similarity metric between two input images, which can be useful in situations where face images may have occlusions or pose variations.

D. Evaluation Metrics

To evaluate the effectiveness of the deep learning models, this study utilizes accuracy, precision, recall, and F1-score, which can be generated using the metrics module from the scikit-learn library available in Python. Subsequently, to determine both models' performance as biometric systems, False Acceptance Rate (FAR), and False Rejection Rate (FRR) are obtained by calculating the total amount of true positives, false positives, true negatives, and false negatives from confusion matrix, as can be seen from Eqs. (1)–(2).

$$FAR = FP \div (FP + TN) \quad (1)$$

$$FRR = FN \div (FP + TP) \quad (2)$$

Meanwhile, to evaluate the resource requirements, we take notes on the total average time and parameters utilized for the models to do training.

IV. Result and Discussion

The focus of this research is on how MobileNet and Siamese neural networks perform under three major challenges: mask usage, lighting conditions, and limited computational resources. The M2FRED dataset provides an opportunity to evaluate the performance of the MobileNet and Siamese neural network algorithms under those conditions. In this section, we elaborate on the results of our experiments at every step along with the analysis.

The overall performance for each model in doing classification tasks overall M2FRED dataset can be seen in Table IV. This table also shows the distinction between models used in the previous research [14].

TABLE IV. Model's Performance

Model	Accuracy	Precision	F1-Score	Recall
MobileNet	99.85	99.85	99.85	99.85
Siamese	92.38	96.17	92.06	88.28
MLP	95.4	95.1	N/A	95.5
SVM	92.2	92.2	N/A	92.2
DT	28.9	27.6	N/A	29
RF	84.1	83.9	N/A	84

In general, MobileNet outperformed the Siamese neural network and models used in prior investigation for every metric. Simultaneously, to evaluate the performance of these models as biometric systems, metrics such as FAR and FRR are used, and the result is shown in Table V.

TABLE V. FAR and FRR Scores

Model	FAR (%)	FRR (%)
MobileNet (Avg)	0.0035	0.145
Siamese (Avg)	7.6	7.6

The FAR and FRR scores shown in Table V represent that MobileNet outperformed the Siamese neural network. FAR score indicates the probability that the system will incorrectly identify an unauthorized person as authorized. The lower the FAR, the better the system's security level. In this case, the average FAR is less than 0.0035%, which is a very low error rate and indicates that the system is

performing well. The Siamese neural networks, on the other hand, is very permissive and got a fairly high score for both FAR and FRR, indicating that the Siamese neural network gives poor performance as a biometric system.

There are also three major issues to be analyzed more thoroughly in this paper: mask usage, lighting variations, and algorithm complexity. Customized M2FRED subsets are prepared for each problem, and then processed to suit the characteristics of the networks used, both MobileNet and Siamese neural networks. All the evaluations are depicted in Fig. 10.

The Siamese neural network never shows better performance than MobileNet. And with the FAR and FRR scores that resulted, it can be concluded that the Siamese neural network is less suitable to be applied to M2FRED.

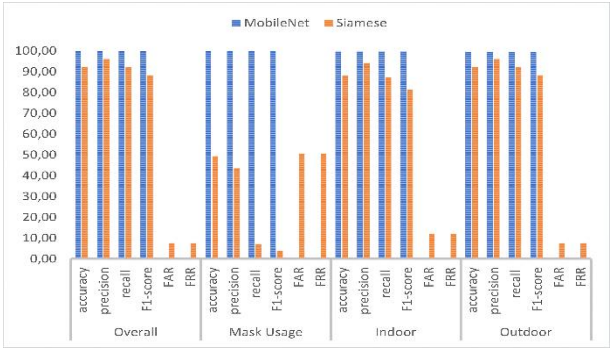

Figure 10. Bar graph of both model's performances.

Meanwhile, more thorough results of the mask and lighting evaluation can be observed in Table VI.

TABLE VI. MASK AND LIGHTING EVALUATION

Model		Accuracy (%)	Precision (%)	F1-Score (%)	Recall (%)	FAR (%)	FRR (%)
Mask Evaluation	MobileNet	100	100	100	100	4.98×10^{-5}	0.003
	Siamese	49.41	43.48	7.17	3.91	50.59	50.59
Indoor Evaluation	MobileNet	99.72	99.72	99.72	99.72	0.0065	0.205
	Siamese	88.09	94.12	87.21	81.25	11.91	11.91
Outdoor Evaluation	MobileNet	99.52	99.52	99.52	99.52	0.011	0,543
	Siamese	92.38	96.17	92.06	88.28	7.61	7.61

A. Results of Mask Usage Evaluation

We evaluate the performance of the models on masked faces, which can simulate real-world scenarios where people are wearing masks. A subset of the M2FRED dataset where the subjects are wearing masks was created, and we evaluated the models' performance on this subset.

The Siamese neural networks performed very poorly for this mask evaluation. This is due to the nature of the Siamese neural network, which works by measuring the similarity score of a pair of images. The testing set provides only masked images, causing the distance value to be too wide, resulting in only 49.41% accuracy and very poor FAR and FRR scores.

On the other hand, MobileNet exhibited robust performance even in the presence of occlusion due to the mask. It achieved an impressive accuracy of 99.79%, outperforming Siamese neural networks by a significant margin. This remarkable accuracy suggests that MobileNet's ability to extract informative features from the facial images was not severely affected by the presence of masks.

B. Results of Lighting Variations

M2FRED provides datasets with various lighting conditions, both indoor and outdoor. By utilizing this M2FRED benefit, we can evaluate MobileNet and Siamese neural network performance to handle illumination issues. For this scenario, we divided it into two sub-scenarios: indoor performance and outdoor performance. Both use a subset of the M2FRED dataset created specifically according to the purpose of testing each scenario.

Again, MobileNet gives us mostly superior performance than the Siamese neural network, though the Siamese neural network works better at handling lighting obstacles than mask usage. Both models perform better in outdoor circumstances, although for outdoor images, there are plenty of images that have backlight conditions.

C. Results on Computational Resource Evaluation

The models' performance under computational resource constraints is evaluated by calculating the usage of time and the number of parameters that need to be trained. Figs. 7 and 9 present the summary of MobileNet and Siamese neural networks, respectively.

It was quite clear from the number of parameters that need to be trained that the Siamese neural network is more complicated than MobileNet by having five times the number of parameters. This also influences the amount of time required to do training. MobileNet takes approximately 3,640 Seconds to do one training (10 epochs, batch_size = 32), whereas the Siamese neural network takes around 17,180 Seconds to perform the same training, not to mention the time needed for data preparation (creating the triplets).

The complexity is depicted in Table VII.

TABLE VII. COMPLEXITY COMPARISON

Model	Train Param	Train Time
MobileNet	4,322,539	3,640 s
Siamese	9,583,800	17,180 s

V. CONCLUSION

By far, MobileNet can demonstrate superior performance over the Siamese neural network in any evaluation metrics, be they accuracy, precision, recall, F1 score, FAR, FRR, and time efficiency. This happened in every testing scenario. The transfer of knowledge from the Xception module gives MobileNet a more comprehensive learning experience rather than the Siamese neural

network, which does the learning from scratch. And despite the complexity of the Siamese neural network, be it in the data preparation process or in the architecture itself, the Siamese neural network could not give better outcomes compared to MobileNet.

In our experiments, we applied different preprocessing methods for MobileNet and Siamese Neural Network. For MobileNet, we extracted frames in all directories and then divided them into training, validation, and test sets. Then for Siamese Neural Network, after the frame extraction, we divided the directories used for training and testing. These variations were necessary due to the specific requirements of each model and allowed us to conduct a fair comparison of their performance.

We also found that both MobileNet and Siamese neural networks exhibited the ability to accurately recognize faces in different lighting conditions and backgrounds, showcasing their robustness in real-world scenarios.

Additionally, it is important to note that the Siamese neural network performed significantly poorer in face recognition tasks that involved mask usage. Due to its nature of comparing image pairs, the Siamese network struggled when faced with masked images, resulting in lower accuracy and subpar FAR and FRR scores. This limitation emphasizes the need for further research and pre-processing techniques, such as focusing on the periocular area, to improve its performance in scenarios with occlusions like masks.

Overall, this research underscores the significance of selecting appropriate deep learning models based on specific use cases and requirements. Both MobileNet and Siamese neural networks showed promise in handling face recognition tasks in diverse environments, and while MobileNet emerged as the top performer, Siamese neural networks still displayed valuable capabilities.

Further research needs to be done to examine the model and architecture that best fit M2FRED, since M2FRED has the potential to be studied further, despite its drawbacks that have been mentioned in Section IV. It is also possible to combine MobileNet and Siamese neural networks by using MobileNet as a feature extractor and then continuing the training using Siamese neural networks to get the similarity score.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Riskie Annisa: initiated the challenge, studied, and evaluated the data, then composed the paper. Benfano Soewito: supervised, reviewed, and did editing in the whole writing process; all authors had approved the final version of this paper.

Acknowledgment

The authors wish to express appreciation towards the class of MTI PUPR and also the team of Binus Graduate Program.

References

[1] G. Dave, X. Chao, and K. Sriadibhatla. (2010). Face recognition in mobile phones. Department of Electrical Engineering, Stanford University. [Online]. pp. 7–23. Available: https://stacks.stanford.edu/file/druid:rz261ds9725/Sriadibhatla_Davo_Chao_FaceRecognition.pdf

[2] E. V. Fernandez and D. G. Jimenez, "Face recognition for authentication on mobile devices," *Image and Vision Computing*, vol. 55, pp. 31–33, 2016.

[3] M. Taskiran, N. Kahraman, and C. E. Erdem, "Face recognition: Past, present, and future (a review)," *Digital Signal Processing*, vol. 106, 2020.

[4] S. Chen, Y. Liu, X. Gao, and Z. Han, "Mobilefacenets: Efficient CNNs for accurate real-time face verification on mobile devices," in *Proc. 13th Chinese Conference on Biometric Recognition*, Urumqi, China, 2018, pp. 428–438.

[5] W. Yao, V. Varkarakis, G. Costache, J. Lemley, and P. Corcoran, "Toward robust facial authentication for low-power edge-ai consumer devices," *IEEE Access*, vol. 10, pp. 123661–123678, 2022.

[6] G. Mahalingam, K. Ricanek Jr., and A. M. Albert, "Investigating the periocular-based face recognition across gender transformation," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 12, pp. 2180–2192, 2014.

[7] M. M. Sawant and K. M. Bhurchandi, "Age invariant face recognition: A survey on facial aging databases, techniques and effect of aging," *Artificial Intelligence Review*, vol. 52, pp. 981–1008, 2019.

[8] D. Deb, L. B. Rowden, and A. K. Jain, "Face recognition performance under aging," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 46–54.

[9] M. S. Ejaz, M. R. Islam, M. Sifatullah, and A. Sarker, "Implementation of principal component analysis on masked and non-masked face recognition," in *Proc. International Conference on Advances in Science, Engineering and Robotics Technology*, 2019, pp. 1–5.

[10] S. Ueda and T. Koyama, "Influence of make-up on facial recognition," *Perception*, vol. 39, no. 2, pp. 260–264, 2010.

[11] G. Givens, J. R. Beveridge, B. A. Draper, P. Grother, and P. J. Phillips, "How features of the human face affect recognition: A statistical comparison of three face recognition," in *Proc. 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, 2004.

[12] P. Capasso, L. Cimmino, A. F. Abate, A. Bruno, and G. Cattaneo, "A PNU-based methodology to improve the reliability of biometric systems," *Sensors*, vol. 22, no. 16, 2022.

[13] M. Umer, S. Sadiq, R. M. Alhebshi, S. Alsubai, A. Al Hejaili, A. A. Eshmawi, M. Nappi, and I. Ashraf, "Face mask detection using deep convolutional neural network and multi-stage image processing," *Image and Vision Computing*, vol. 133, 2023.

[14] L. Cimmino, M. Nappi, and C. Pero, "M2FRED: Mobile masked face recognition through periocular dynamics analysis," *IEEE Access*, pp. 94388–94402, 2022.

[15] I. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *Proc. International Conference on Pattern Recognition*, Mexico, vol. 2, 2016.

[16] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyend, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *Computer Vision and Pattern Recognition*, vol. 33, 2017.

[17] Y. Zhou, "The efficient implementation of face mask detection using mobilenet," *Journal of Physics: Conference Series*, vol. 2181, no. 1, 2022.

[18] J. Bromley, I. Guyon, L. Yann, E. Sackinger, and S. Roopak, "Signature verification using 'Siamese' time delay neural network," *Advances in Neural Information Processing Systems*, vol. 6, pp. 737–744, 1993.

[19] Y. Taigman, M. Yang, M. A. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[20] C. Song and S. Ji, 'Face recognition method based on Siamese networks under non-restricted conditions," *IEEE Access*, vol. 10, pp. 40432–40444, 2022.

[21] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," arXiv preprint, arXiv:1411, vol. 7923, 2014.
[22] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.
[23] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," *Real-Life 'Images: Detection, Alignment, and Recognition*, pp. 1–14, 2008.
[24] Y. Aufar and I. S. Sitanggang, "Face recognition based on Siamese convolutional neural network using KIVY framework," *Indonesian Journal of Electrical Engineering and Computer Science*, pp. 764–772, 2022.
[25] T. Ahmed, P. Das, M. F. Ali, and M. F. Mahmud, "A comparative study on convolutional neural network based face recognition," in *Proc. 11th International Conference on Computing, Communication and Networking Technologies (ICCNT)*, 2020, pp. 1–5.
[26] D. Brown, "Mobile attendance based on face detection and recognition using OpenVINO," in *Proc. International Conference on Artificial Intelligence and Smart System*, 2021, pp. 25–27.
[27] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting masked faces in the wild with LLE-CNNs," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2682–2690.
[28] B. Roy, S. Nandy, D. Ghosh, D. Dutta, P. Biswas, and T. Das, "MOXA: A deep learning based unmanned approach for real-time monitoring of people wearing medical masks," *Transactions of the Indian National Academy of Engineering*, vol. 5, pp. 509–518, 2020.
[29] B. Huang, Z. Wang, G. Wang, K. Jiang, Z. He, H. Zou, and Q. Zou, "Masked face recognition datasets and validation," in *Proc. IEEE/CVF International Conference on Computer Vision*, pp. 1487–1491, 2021.
[30] Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Pei, H. Chen, Y. Miao, Z. Huang, and J. Liang, "Masked face recognition dataset and application" in *IEEE Transactions on Biometrics, Behaviour, and Identity Science*, vol. 5, no. 2, pp. 298–304, 2023.
[31] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Colorado, 2011.
[32] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre, "XM2VTSDB: The extended M2VTS database," in *Proc. Second International Conference on Audio and Video-Based Biometric Person Authentication (AVBPA '99)*, Washington DC, 1999.