# Facial Expression Recognition for Masked Faces

**Tina He**
Department of Engineering Science
University of Toronto
tina.he@mail.utoronto.ca

**Sviatoslav Leniuk**
Department of Engineering Science
University of Toronto
sviatoslav.leniuk@mail.utoronto.ca

## Abstract

Facial expression recognition (FER) in the wild is challenging, hence there is ongoing work to develop models that are robust to various environmental conditions. One challenge for FER is the partial occlusion of faces. Given the COVID-19 pandemic, mask-wearing in public spaces has become the norm, which makes FER for masked faces a new and relevant problem. The KDDI's research team has released a new dataset (M-LFW) containing synthetically masked faces. This dataset has only been evaluated on VGG19 and MobileNet by Yang et al. We evaluate this dataset with two other architectures, ResMaskingNet and Cbam_Resnet50, in order to further understand how real and synthetic face masks affect FER. Both architectures outperform the models reported by Yang et al. However, there remains a significant performance gap between synthetic and real masked faces.

## 1 Introduction

Facial expressions play a key role in how we understand and communicate with each other. In computer vision, automatic facial expression recognition (FER) is a promising area of research, with many applications including automatic counselling, operator fatigue detection, facial expression synthesis, etc. [1]. While deep learning-based FER models have made significant progress, partial occlusion of faces is a major challenge for FER systems [2]. In order to develop occlusion-tolerant systems, artificially superimposed occlusion has been used with some FER datasets [2], [3]. There are also FER datasets with occlusion under naturalistic conditions (e.g. caused by varying head poses, accessories, and interaction with objects) [2], [4], [5].

With the advent of the COVID-19 pandemic, wearing face masks has become the norm for interactions in public spaces [6]. Therefore, FER with face masks presents a relevant new challenge. However, there is very little work on this topic. Recently, Yang et al. developed a method to artificially add face masks to existing FER datasets, and they released the first masked FER dataset based on the Labeled Faces in the Wild (LFW) database [6], [7]. This new dataset (M-LFW [8]) has only been tested by Yang et al. using VGG19 and MobileNet models [6]. Our goal is to evaluate it on two other architectures, ResMaskingNet and Cbam_Resnet50 [9], which have shown good results on FER tasks, in order to further assess the impact of face masks on emotion recognition. The code for this project can be found at: `https://github.com/tinahe75/MaskedFER`.

## 2 Related Works

In recent years, deep learning methods have been predominantly used in improving automatic FER performance. The new models utilize attention layers in their network to create a relative mapping of the important features for classification [9], [11]. Cbam_Resnet50 and ResMaskingNet were designed based on this principle and as a result achieved validation accuracy of 73.39% and 74.14%, respectively, on the FER2013 dataset [9]. Even though these models currently outperform other

Preprint. Under review.

models on many datasets, there is limited amount of work conducted on applying them to occluded faces. Studies analysing FER models have shown that models tend to rely heavily on features near the mouth in order to accurately distinguish emotions especially among anger, happiness, fear, and sadness [5]. Various deep learning models have been proposed to combat occlusion, but it is not clear how well the Cbam_Resnet50 and ResMaskingNet architectures would fare when features near the mouth are not available.

Many forms of partially occluded FER have been studied. For example, Li et al. rendered random objects in front of faces, and trained CNNs with attention mechanism on these images [10]. Other studies utilize naturally occluded images caused by various conditions (pose, accessories, etc.) [2]. However, most studies in literature do not focus on a specific type of systematic occlusion.

Recently, Yang et al. released the new M-LFW dataset [8], shown in Figure 1, which systematically masks both the mouth and nose for faces with front or side views. This increases FER difficulty, but some previous works have shown that many people tend to rely on the eyes to discern a face's emotional expression rather than the mouth or nose [10]. Using MobileNet and VGG19, Yang et al. achieved 66% and 54% validation accuracy on M-LFW [6]. However, they observed much lower accuracies when the models were applied to real-world masked faces.


Figure 1: LFW images are annotated and synthetically masked to produce M-LFW [8]

## 3 Methods

### 3.1 Test dataset: real-world masked faces

An important observation made by Yang et al. is that there is a considerable performance gap between the model's accuracy on synthetic and real masked images [6]. Ideally, the FER models should be evaluated on a large number of masked faces in naturalistic settings. However, Yang et al. did not release their annotated test dataset described in [6], which contains real masked faces that were crawled online. The M-LFW dataset only has synthetic masked images. In order to gauge FER performance on real-world masked faces, we manually curated and annotated a small set of masked faces. The images were carefully selected to resemble the M-LFW data in the following ways: 1) the faces are mostly front-facing and placed in the middle of the image; 2) there is sufficient space surrounding the faces. Figure 2 shows samples of images in the test set.

Figure 2: Sample test images (with negative, neutral, positive labels)

Most of these images are found via Google Images and Freepik, which is a source for free stock photos. Similar to [6], keyword search was used (e.g. smiling + face mask + person). Our test dataset contains 45 images for each of the classes. We acknowledge that this is a very small dataset, so it can only provide a limited evaluation of the FER models. The small size is due to the highly time-consuming process of searching, cropping, and annotating images. There are a number of online-scraped masked faces datasets available on Kaggle and Github. We were not able to use these datasets, however, because many of these images are not sufficiently similar to the M-LFW dataset or difficult to annotate.

### 3.2 Evaluation of models

We adapted the open-source code developed by Pham et al. [9] for M-LFW. In the first set of experiments, we trained ResMaskingNet and Cbam_Resnet50 using M-LFW, in order to compare with the performance of VGG19 and MobileNet in [6]. We hypothesized that ResMaskingNet and Cbam_Resnet50 may achieve better results than VGG19, given that they outperformed VGG19 for FER2013 [6]. In the second set of experiments, we trained ResMaskingNet and Cbam_Resnet50 on unmasked data and investigated how well they apply to masked images. Yang et al. have found that masked faces significantly reduce performance of FER models that are trained on unmasked images [6]. Hence, we also expect a similar gap for our models.

After initial experiments, we observed significant overfitting for both ResMaskingNet and Cbam_Resnet50. To mitigate this, we applied data augmentation. Various augmentation techniques were used, including random horizontal flip, rotation, colour jitter, translation, and Gaussian noise. The original training set of M-LFW only contains 9307 images, and the augmented dataset is four times larger.

Additionally, we also examined different configurations for transfer learning. Pham et al. fine tuned ImageNet pre-trained models using FER2013 data [9]. They replaced the output layer of pretrained models with a fully connected layer. This setup leads to overfitting on our dataset, so we experimented with several alternatives (adding dropout, freezing layers, training from scratch). However, even with data augmentation, none of these produced satisfactory results. When selecting configurations, there is a trade-off between overfitting and accuracy. We chose to prioritize the latter. In the next section, we discuss our observations in greater detail and report the models with highest test accuracy.

## 4 Results and Discussion

### 4.1 Classification performance of models

Table 1 shows the train, validation and test accuracies obtained by the models on masked and unmasked faces. Similar to [9], the models are initialized with ImageNet pretrained weights. For both masked and unmasked faces, ResMaskingNet and Cbam_Resnet50 achieved better performance than the MobileNet and VGG19 models trained by Yang et al. For M-LFW, their MobileNet model achieved 72.13% and 66.14% train and validation accuracy; VGG19 produced 58.24% and 53.54% train and validation accuracy [6].

We also experimented with training on M-LFW after training on LFW. This is equivalent to pre-training on unmasked faces, and then fine tuning on masked faces. However, this produced lower performance, compared to directly fine tuning ImageNet pre-trained weights. Hence, we only use the latter setup when reporting results in Table 1. Model hyperparameters are included in the Appendix.

Table 1: Classification accuracy (%) of models on masked and unmasked faces

| **Train and validation dataset** | **ResMaskingNet** | | | **Cbam_Resnet50** | | |
|---|---|---|---|---|---|---|
| | Train | Validation | Test | Train | Validation | Test |
| M-LFW (masked faces) | 81.504 | 78.417 | 52.593 | 91.692 | 76.750 | 53.333 |
| LFW (unmasked faces) | 98.862 | 89.167 | 33.333 | 92.208 | 88.000 | 34.815 |

The model performance degrades when tested on real-world masked faces, as shown in Table 1. This is similar to the behaviour reported by Yang et al. [6]. If the models are trained on unmasked faces, they perform very poorly on masked faces. The test accuracy is largely equivalent to random guessing. If trained on synthetic masked faces, the test accuracy improves, but still falls short of the validation accuracy. One of the main causes for this performance gap may be the appearance of masks. In the synthetic dataset, all the masks are white. In contrast, masks in the test set have a wide variety of colours, textures, and shapes. This likely makes it very challenging for the model to generalize.

Figure 3 shows the confusion matrices for the M-LFW models on the test set. For both ResMaskingNet and Cbam_Resnet50, the most common mistakes involve misclassifying negative as neutral, or misclassifying neutral as positive. Both architectures have the lowest error for positive labels and highest error for negative labels. The class imbalance of M-LFW is likely contributing to these

mistakes. In the training set, the percentages of negative, neutral, and positive labels are 8.2%, 36.0%, and 55.8% respectively. Due to this imbalance, the models have less information about the negative class, and are therefore more likely to bias towards the two other classes. The Appendix contains additional confusion matrices for the validation set and for the LFW models.

During training, we observed overfitting for both ResMaskingNet and Cbam_Resnet50. After a number of epochs, the training accuracy approaches 100%, while validation accuracy remains in the 60-80% range. The highest validation accuracy usually appears after only a few epochs of training. Data augmentation did not resolve this issue, but it did slightly increase the accuracy of all models. Adding dropout before the output layer also did not eliminate overfitting. Furthermore, even if pre-trained weights are not used, the models still overfit. The only strategy that curbed overfitting is freezing pre-trained layers. However, this led to a significant drop in both train and validation accuracy.

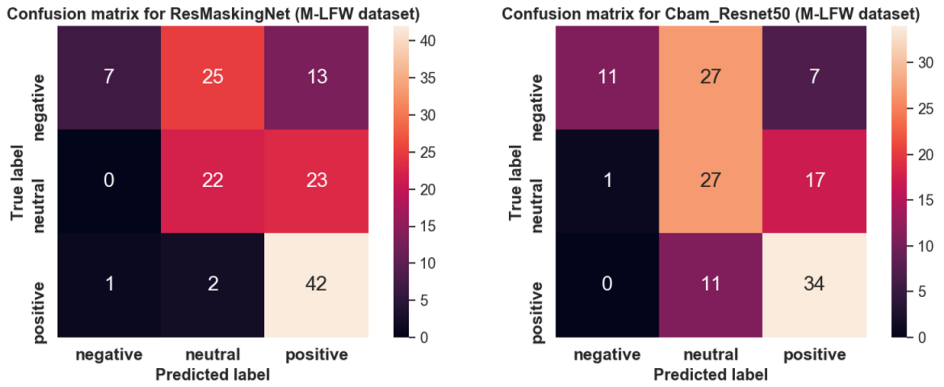


Figure 3: Confusion matrices for real-world test images, after training on M-LFW

Given that none of the above strategies resolved overfitting, this indicates that the architectures may be too complex given the dataset size and task difficulty. The M-LFW dataset only has less than 11,000 images and 3 classes. Furthermore, ResMaskingNet and Cbam_Resnet50 are deeper architectures than VGG19 and MobileNet, which may be one of the reasons that we observe more overfitting than Yang et al. [6].

### 4.2 Limitations and suggestions for future work

Given the limited size of the M-LFW dataset, it is difficult to ensure adequate performance without overfitting. Other architectures and techniques for masked FER are worth further investigation, in order to improve validation and test accuracy.

In the field of FER and facial recognition, most of the synthetic masked datasets to date are rendered with the same type of mask [8], [12], [13]. Given that real-world face masks can vary significantly in colour, shape, fit, and texture, one suggestion for future work is to add variation to synthetic masks. For example, random textures and patterns can be rendered onto face masks, which would increase the diversity of images in the dataset. Models that are trained with such datasets may be more robust and generalize better to real-world masked faces.

## 5 Conclusion

For both masked and unmasked data, ResMaskingNet and Cbam_Resnet50 outperform the MobileNet and VGG19 models trained by Yang et al. [6]. Models trained on unmasked data perform very poorly on real-world masked faces. Training on synthetic masked faces increases the accuracy. However, we still observe a performance gap between synthetic and real masked faces. Improving masked FER is a topic worth further research, and we recommend diversifying synthetic masks as a potential next step forward.

## References


[1] Kumari, R. Rajesh & K. Pooja, "Facial Expression Recognition: A Survey", *Procedia Computer Science*, vol. 58, pp. 486-491, 2015. Available: 10.1016/j.procs.2015.08.011. https://doi.org/10.1016/j.procs.2015.08.011

[2] L. Zhang, B. Verma, D. Tjondronegoro and V. Chandran, "Facial Expression Analysis under Partial Occlusion", *ACM Computing Surveys*, vol. 51, no. 2, pp. 1-49, 2018. Available: 10.1145/3158369. https://arxiv.org/pdf/1802.08784.pdf

[3] M. Lyons, M. Kamachi, and J. Gyoba, *The Japanese Female Facial Expression (JAFFE) Dataset*, Zenodo, 1998. [Online]. Available: http://doi.org/10.5281/zenodo.3451524.

[4] A. Colombo, C. Cusano and R. Schettini, "UMB-DB: A database of partially occluded 3D faces," *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, Barcelona, Spain, 2011, pp. 2113-2119, doi: 10.1109/ICCVW.2011.6130509.

[5] X. Burgos-Artizzu, P. Perona and P. Dollar, "Robust face landmark estimation under occlusion", *ICCV*, vol. 2013, 2013. http://www.vision.caltech.edu/xpburgos/ICCV13/#cite

[6] B. Yang, J. Wu and G. Hattori, "Facial Expression Recognition with the advent of face masks", in *19th International Conference on Mobile and Ubiquitous Multimedia*, Essen, Germany, 2020, pp. 335-337 [Online]. Available: https://dl.acm.org/doi/10.1145/3428361.3432075.

[7] G. Huang, M. Ramesh, T. Berg and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments", 2007. http://vis-www.cs.umass.edu/lfw/

[8] B. Yang, J. Wu, and G. Hattori, *LFW emotion dataset*, ACM MUM, 2020. [Online]. Available: https://github.com/KDDI-AI-Center/LFW-emotion-dataset

[9] L. Pham, H. Vu and T. Tran, "Facial Expression Recognition Using Residual Masking Network", *International Conference on Pattern Recognition*, no. 15, 2021. https://github.com/phamquiluan/ResidualMaskingNetwork

[10] Y. Li, J. Zeng, S. Shan and X. Chen, "Occlusion Aware Facial Expression Recognition Using CNN With Attention Mechanism", IEEE Transactions on Image Processing, vol. 28, no. 5, pp. 2439-2450, 2019. Available: 10.1109/tip.2018.2886767.

[11] S. Woo, J. Park, J. Lee and I. Kweon, "CBAM: Convolutional Block Attention Module", Computer Vision Foundation, no. 15, 2018.

[12] A. Cabani, K. Hammoudi, H. Benhabiles, and M. Melkemi, "MaskedFace-Net – A dataset of correctly/incorrectly masked face images in the context of COVID-19," *Smart Health*, vol. 19, no. 100144, Nov. 2020. Available: 10.1016/j.smhl.2020.100144.

[13] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic," M *Measurement*, vol. 167, no. 108288, p. 108288, Jul. 2020. Available: 10.1016/j.measurement.2020.108288.

## Appendix

ResMaskingNet and Cbam_Resnet50 models were trained on Google Colab. ResMaskingNet uses the ImageNet pretrained weights for Resnet34. The output layer is replaced by a dropout layer (p=0.4) and a fully connected layer. Cbam_Resnet50 uses pretrained weights for Cbam_Resnet50, with a modified fully connected layer at the end. The original image size for M-LFW is 250x250, and all data is resized to 224x224 with 3 channels input. For both ResMaskingNet and Cbam_Resnet50, learning rate of 0.0001, momentum of 0.9, weight decay of 0.001, and batch size of 48 are used. Number of epochs is capped at 50, and the learning rate reduces if plateauing occurs for 2 epochs. Early stopping occurs if there is plateauing for 8 epochs.

Figure 4 shows the confusion matrices for ResMaskingNet and Cbam_Resnet50 on the validation set. For both models, common mistakes include misclassifying positive as neutral, or misclassifying negative as neutral. Similar to the test data, there is poor performance for the negative class. Most of the negative labels are misclassified.

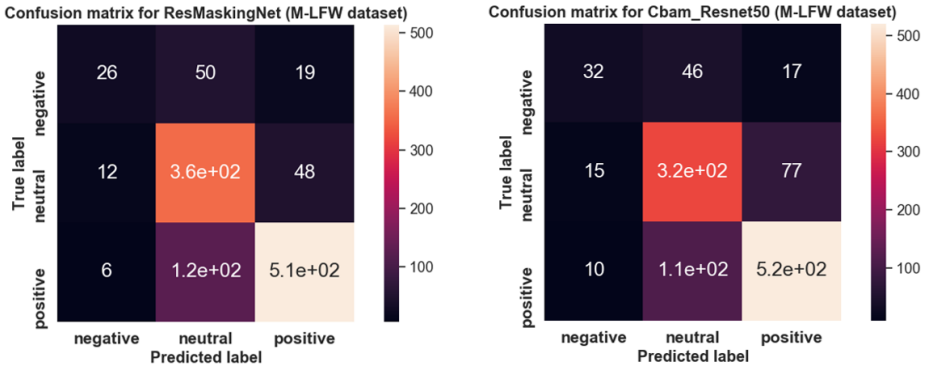


Figure 4: Confusion matrices for validation data (M-LFW)

Figure 5 shows the confusion matrices for ResMaskingNet and Cbam_Resnet50 on the test set, after training on unmasked faces. Both models fail to recognize most of the negative labels. There is also a lot of confusion between neutral and positive classes. This performance is considerably worse than Figure 3.

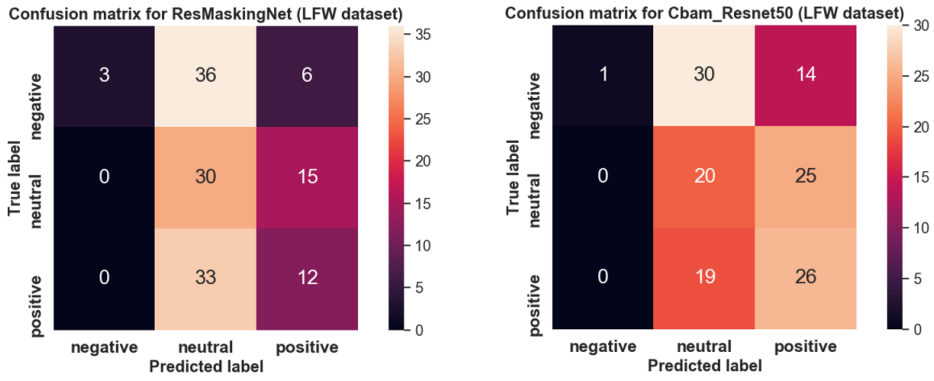


Figure 5: Confusion matrices for test data (LFW)

Figure 6 shows the confusion matrices for ResMaskingNet and Cbam_Resnet50 on the validation set, after training on unmasked faces. Both models make a lot of mistakes for the negative class. However, the models rarely misclassify positive as neutral, compared to Figure 4.

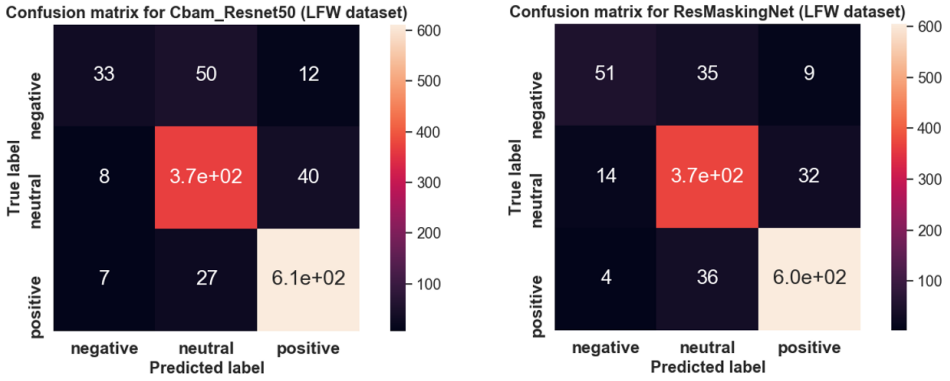


Figure 6: Confusion matrices for validation data (LFW)

Figure 7 shows misclassified samples for ResMaskingNet and Cbam_Resnet50 on the test set, after training on M-LFW. Figure 8 shows misclassified samples on the validation set. The samples included are misclassfied by both architectures. It should be noted that for the validation set, annotations are based on unmasked faces. Hence, the distinction between classes can be more subtle. Even to the human eye, some of the images are difficult to classify after masking. In contrast, for the test set, the annotations are based on masked faces. If the emotion appears ambiguous for an image, we did not annotate and include it in the test set. As a result, especially for the positive and negative classes, expressions tend to be more exaggerated and also easier for humans to recognize.

Figure 7: M-LFW: Negative misclassified as neutral (top); neutral misclassified as positive (bottom)

Figure 8: M-LFW: Negative misclassified as neutral (top); positive misclassified as neutral (bottom)

Figure 9 shows misclassified samples for ResMaskingNet and Cbam_Resnet50 on the test set, after training on unmasked faces. Figure 10 shows misclassified samples for ResMaskingNet and Cbam_Resnet50 on the validation set, after training on unmasked faces. The samples included are misclassfied by both architectures.

Figure 9: LFW: Negative misclassified as neutral (top); positive misclassified as neutral (bottom)

Figure 10: LFW: Negative misclassified as neutral (top); positive misclassified as neutral (bottom)