Facial Expression Recognition Using Attention Technique

Adarsh Ghimire, Selina Shrestha, Youssef Ibrahim

November 30, 2020

Abstract

In this project, we approach the problem of facial expression recognition in images by using a deep learning framework along with an attention mechanism that is able to put focus on regions of the face that are more useful in identifying the person's expression. Previous works on this have used mechanisms like region attention [1] and spatial affine transformation [2] in order to add attention to their convolutional networks. However, there are many other attention networks for image classification tasks that have not been applied to facial expression recognition yet. In this project, we propose the addition of a lightweight Convolutional Block Attention Module (CBAM) to the convolutional network so as to classify facial expressions more accurately, and at little added computational expense. In the proposed system, facial features are first extracted from input image using the VGG16 convolutional base, which has been pre-trained on VGG face dataset, and has its last 3 layers fine-tuned on the training dataset. These features are then passed to the CBAM module, which applies channel-based and spatial attention to generate refined features that are then used by the fully connected classifier to predict the person's expression. The results demonstrate that the addition of the CBAM attention module is able to increase the system accuracy at a very little computational overhead. Additionally, the proposed model has been successful in outperforming the top model in Kaggle Leaderboard for this task.

Introduction

Human emotions play an important role in how we function and the analysis of these emotions can be used in several areas like psychology, marketing, security, event management, etc. Thus, automatic emotion recognition has been an ongoing research topic in the field of Artificial Intelligence. While there are many other sources for emotion recognition like speech and gestures, facial expression is a primary gateway of emotions. Thus, an area called Facial Expression Recognition (FER) has emerged in which the expression of a person captured in images is detected and classified as happy, sad, angry, etc. Identifying facial expressions as such can be of great help in detecting underlying human emotions.

Most of the works on Facial Expression Recognition [1–10] make use of Convolutional Neural Networks owing to their wide popularity and effectiveness

with image data. However, majority of these works do not take into account the fact that expression of emotions affect certain regions of the face more than the others, i.e. certain parts of the face play a bigger role in identifying emotions. To benefit from this, the concept of attention network was introduced [1] whereby the model focuses on features in more important regions of the face while classifying the person's emotion. Aside from the benefits of predicting output using significant regions of the face, the attention technique may also help in tackling the problem of occlusion in real-life images. Since the model does not equally rely on each part of the face for classification, it will be able to sustain its performance even when some of the lesser important facial parts are hidden. [1] proposed a region attention network which focuses on regions of the face by first cropping the image multiple times while [2] made use of the spatial transformer network to integrate attention into the model.

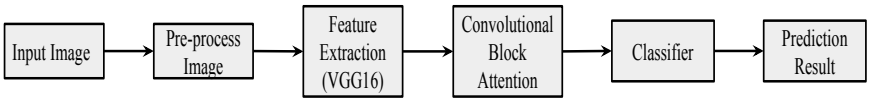

Figure 1: Overview of proposed system

In this work, we have used the Convolutional Block Attention Module (CBAM) to add channel and spatial attention to our convolutional neural network in order to increase its accuracy in expression recognition. Our deep learning framework, as shown in figure (1) consists of the VGG16 convolutional base pre-trained on the VGG face dataset to extract features from facial images, followed by the CBAM attention module that emphasizes the more relevant channels and spatial pixels, followed by a fully connected classifier that classifies the input image into one of the seven emotions: happy, sad, anger, fear, surprise, disgust and neutral. The framework has been trained on the FER2013 dataset. The use of a convolutional base pre-trained on face images provided a good baseline performance for the model, and the model accuracy was further boosted by the addition of the attention network.

Related works

In the literature, facial expression recognition systems have been implemented in various ways ranging from the use of hand-crafted features, prior knowledge features, geometric features, to the use of convolutional networks to extract the features of the image in order to identify the expression.

Earlier works on facial expression recognition used a traditional machine learning approach consisting of two steps: 1. Feature extraction from images, 2. Classification. These approaches used hand-crafted features such as histogram of oriented gradients (HOG) [11], local binary patterns (LBP) [12, 13], Gabor wavelets [13] and Haar features [12, 14]. These features were then used by classifiers such as SVM [11, 13], neural network [15] and random forest [12] in order to find the best emotion for the image. While these methods worked fairly well with simple datasets, they failed to do so in more challenging datasets with

partial and occluded faces. The limitations of these methods lie in the fact that they use prior facial knowledge for feature extraction, which only works well in the availability of complete faces. These techniques thus failed in images with partially visible faces.

Several recent works related to emotion recognition use Deep Neural Networks and most of them rely on Convolutional Neural Networks (CNN) [1–10]. [3] proposed a model that incorporates CNN network along with the concept of residual blocks. [4] used the ensemble technique where multiple convolutional network was trained in parallel, each network responsible for the abstract learning of feature map. The network shared a common loss function at the end, such that the parameter updates allowed it to have view of shared representations learning. [5] did not include either pre or post processing steps. However, they used two Parallel Feature Extraction blocks inspired by GoogleNet [9], which used different sized filters to extract facial features, thereby compensating for the various scales, dimensionality, and feature abstractions in input images. [8] used a model of a two-level CNN framework where the first level was responsible for background removal from an image and the second level used a conventional CNN network to make prediction of emotion based on the primary expression vectors (EV) where the expressional vector (EV) is generated by tracking down relevant facial points of importance and is directly related to changes in expression.

All of the above works using deep neural networks achieved significantly greater accuracy than the earlier works that made use of hand-crafted features. However, these works missed to integrate the fact that some facial regions play a greater role in expressing emotions than the others. To incorporate this, the concept of attention network was introduced in [1] whereby the model attended more to the important regions of the face while classifying the emotion. Attention is one of the most powerful concepts in the deep learning field nowadays. It is like attending to certain key sections when processing huge amounts of information, which in fact reduces the resource requirement alongside improves the performance. [1] proposed a region attention network which focuses on regions of the face by first cropping the image multiple times so as to generate different regions and then pass them through two attention modules: self-attention module and relation attention module. The self-attention module learns coarse attention weights while the relation attention module refines them to the global context of the image. In addition, the region biased loss is introduced which allows high attention to the important regions and areas of interest. [2] on the other hand uses the concept of attentional convolutional network to attend to the specific regions of the face through the use of spatial transformer networks [10]. The spatial transformer network uses affine transformations of the images to find the region of attention. [16] proposed a Convolutional Block Attention Module which uses the concept of adding channel attention and spatial attention successively to focus on important features and suppress the unnecessary ones in the given feature map. However, this attention technique has not been tested on facial expression recognition tasks. So, in this project, we have explored the addition of the CBAM attention network [16] to a deep convolutional neural network for the task of facial expression recognition and noted the improvement in the performance.

Methodology

In the proposed method, the images are first scaled and normalized and then sent into the FER model consisting of:

1. VGG16 convolutional base
2. Convolutional Block Attention Module (CBAM)
3. Fully connected classifier

During training, as shown in figure (2), the image generator generates images from the training set, which after being pre-processed are sent into the model for classification. The loss function then computes the categorical loss between the predicted result and true label which is then used by the Adam optimizer to update weights in the network. The in depth flow of the network architecture is shown in figure (3).

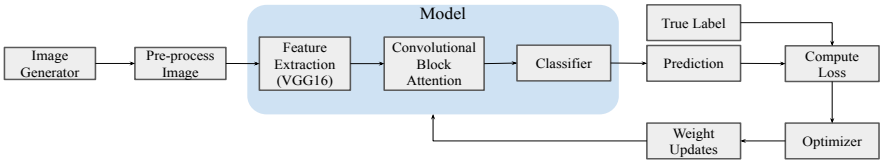

Figure 2: Training flow diagram

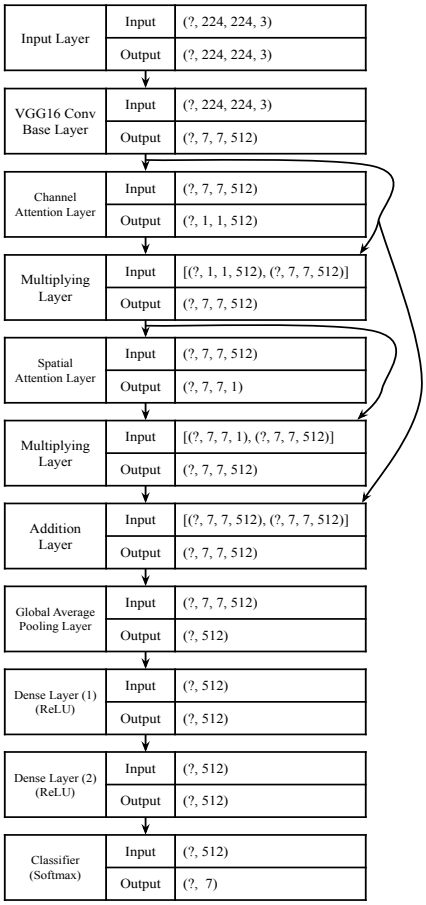

Figure 3: In-depth flow diagram of proposed model

Initially, the VGG16 convolutional layers were frozen and only the attention and classifier weights were trained. After, that the last 3 layers of the VGG16 convolutional base were fine-tuned. During the fine-tuning process the learning rate was reduced by factor of 10 in each of the three layers.

Details of proposed system methodology have been briefed in the sections below.

1. VGG16

   For feature extraction from facial images, we have used a VGG16 model that was pre-trained on VGG Face dataset [17]. The pretrained VGG16 model takes a 224x224 RGB image and the model architecture comprises 13 linearly stacked convolutional layers. The convolution filters are of size 3x3 and are applied with a stride of 1. Max pooling layers are inserted between convolutional blocks as shown in figure (4). The output of the convolutional feature extraction section of size 7x7x512 is flattened and sent to a fully connected classifier consisting of 3 dense layers and softmax output activation. Figure (4) shows the detailed architecture.

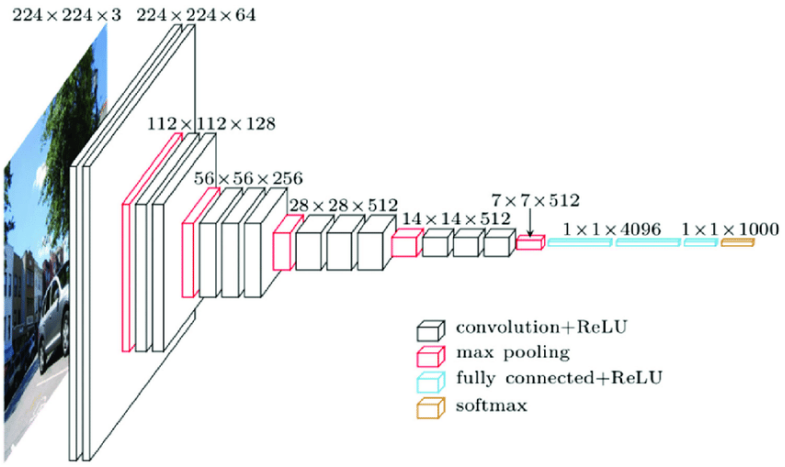

Figure 4: VGG16 Model Architecture [18]

While training our model, we first used the convolutional base of VGG16 (pre-trained on VGGFACE) to train the proposed attention layer and classifier network. After training for some epochs, we unfreezed the last layer of the VGG16 convolutional base and trained the whole model again. This fine-tuning process was repeated for up to the third last layer, and during each re-training step, the learning rate was reduced by a factor of 10.

2. Convolutional Block Attention Module (CBAM) [16]

   CBAM provides a generic and lightweight solution that can fit properly with any Convolution Neural Network with minor and insignificant overhead. Its main target is to refine the features of a given intermediate feature map using its attention maps. Attention maps are deduced on two discrete dimensions: spatial (M_s) and channel (M_c). Using equations (1) and (2), these attention maps are then multiplied by the intermediate feature maps $F^{'}$ and F respectively, which are the input to the respective attention module, in order to produce adjustable feature improvement.

$$F^{'} = M_c(F) \times F \tag{1}$$

$$F^{''} = M_s(F^{'}) \times F^{'} \tag{2}$$

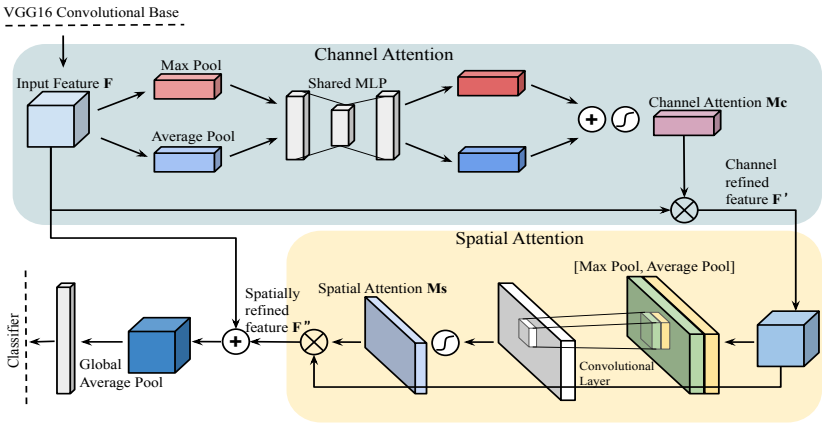

Figure 5: Convolutional Block Attention Module Implementation

(a) Channel sub-module

In this module a map is produced from the inter-channel association of features. This module looks for the meaningful channel of the input feature map. The spatial dimensions of the given image's feature map (F) is compressed so that effective channel attention can be generated. For channel attention map, the average pooling is used to accumulate all the spatial information and max pooling is used find distinctive features in each channel. The output of these two operators enters a Multi-Layer Perceptron (MLP) to provoke the channel attention map (M_c). Then the output is multiplied with input feature map to generate the channel refined feature map ($F^{'}$). Equation (3) shows the computations performed by the channel attention sub-module to generate the channel attention map.

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{3}$$

(b) Spatial sub-module

This module is looking for the location of the meaningful parts of the input feature map ($F^{'}$), which is approving the task of the channel attention module. It works by creating two features maps which are obtained by performing max pooling and average pooling across the channels of the input feature map ($F^{'}$) which are used to describe the features of the image in an efficient manner. After that, a spatial attention map is computed by performing convolution operations across those feature maps using a convolution filter of size 3. After convolving both, a 2D spatial attention map is produced. Then the input feature map ($F^{'}$) is multiplied with the computed spatial attention map (M_s) which results in giving spatially refined features. Equation (4) shows the computations performed by the spatial attention sub-module to generate the spatial attention map.

$$M_s(F) = \sigma(f^{3x3}([AvgPool(F); MaxPool(F)])) \tag{4}$$

3. Classifier

   The Attention network output was given to the global average pooling layer, after which the two fully connected layers with 512 neurons with relu activation in both are used to learn the necessary features. The drop out of 0.5 was kept before both of the hidden layers such that model can generalize more. And lastly, a dense layer with 7 neurons that is softmax activated has been used to classify the image.

4. Loss Function

   Categorical cross-entropy is used to compute the loss of the network because it is a multi-class classification problem. This type of loss function is the best one to be used with this type of task. In our case, we have two probability distributions one of them is of the probability distribution of the correct labels of the data (which are the one-hot encoded labels of 7 categories) and the other is the probability distribution predicted by the model, which the categorical cross-entropy loss measures such that difference between those two is decreased by the optimizer in order to reach the similar distributions.

5. Optimizer

   The Adaptive Momentum (Adam) optimizer has been used because of its momentum and faster convergence feature. It is very efficient in terms of computations and adjusts an adaptive learning rate for each of the parameters in order to minimize the overall loss rate. Moreover, it does not need lots of memory and works well with noisy data [State-of-the-Art CNN Optimizer for Brain Tumor Segmentation in Magnetic Resonance Images]. During transfer learning, the learning rate was set to 0.001, however during fine tuning each of the last 3 convolutional layers of the VGG16 was re-trained one at a time by reducing the learning rate by a factor of 10 each time.

Experiments and Results

Dataset

The Facial Expression Recognition 2013 (FER-2013) dataset, which was first used in ICML 2013 Challenges in Representation Learning, was used for training and validation. This dataset is composed of 48x48 grayscale face images, which we have rescaled to 224x224 before passing into the system. The dataset contains a total of 35,887 images divided into three sets: 28,709 (80%) training set images, 3,589 (10%) validation set images, and 3,589 (10%) test set images. The public dataset has been used as the validation set and the private dataset as test set.

What differentiates FER-2013 dataset from other datasets available for the task of Facial Expression Recognition is that it contains a wide range of variations concerning partial faces, face occlusion by hand, sunglasses, etc [2]. The face images are labeled with one of the seven categories of primary emotions: Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral. The images are not equally distributed among all seven categories as shown in figure (6).

Set/Emotion	Angry	Disgust	Fear	Happy	Neutral	Sad	Surprise
Training Set	3995	436	4097	7215	4965	4830	3171
Validation Set	467	56	496	895	607	653	415
Test Set	491	55	528	879	626	594	416

Figure 6: Dataset distribution in each category

To combat the heavy imbalance of samples in each categories of the dataset, pre-computed weights for each category is given the model prior to training. Equation(5) has been used to compute weights for each category, where higher weight is given to categories with fewer images and vice versa, due to which adverse effects of class imbalance are moderated.

$$CategoryWeight = \frac{TotalNumberOfTrainingSamples}{NumberOfClasses \times NumberOfSamplesInTheCategory} \tag{5}$$

Experiments

In order to choose our model architecture and in order to evaluate it, the following was done:

1. Choosing the best convolutional base

   Different convolutional base architectures along with a fully connected classifier, as shown below, were trained and evaluated on the validation set to choose the best convolutional base.

   (a) Small Convolutional Network + Classifier
   (b) Resnet50 pre-trained on Imagenet Dataset + Classifier
   (c) Resnet50 pre-trained on VGGFace Dataset + Classifier
   (d) VGG16 pre-trained on Imagenet + Flatten + Classifier
   (e) InceptionV3 pre-trained on Imagenet Dataset + Global Average Pooling + Classifier
   (f) VGG16 pre-trained on VGGFace Dataset + Flatten + Classifier
   (g) VGG16 pre-trained on VGGFace Dataset + Global Average Pooling + Classifier

   The pre-trained CNN models were additionally fine-tuned on the training set. Among all these networks, the VGG16 pre-trained on VGGFace dataset, followed by global average pooling and classifier was able to perform the best. Thus, this convolutional base was selected for the addition of the attention network.

2. Adding the CBAM attention network to the best convolutional base

   CBAM network was added after the convolutional base of VGG16 pre-trained on VGGFace. The output of the attention network was then fed to a global average pooling layer and then to the classifier. The training and fine tuning was again repeated for the whole model.

Results

Models	Validation Accuracy (Public dataset)	Test Accuracy (Private Dataset)
Small Convolutional Network + Classifier	20.70 %	20.28 %
Resnet50 pre-trained on Imagenet Dataset + Classifier	25.24 %	25.01 %
Resnet50 pre-trained on VGGFace Dataset + Classifier	25.80 %	26.45 %
VGG16 pre-trained on Imagenet Dataset + Flatten + Classifier	33.80 %	32.20 %
InceptionV3 pre-trained on Imagenet Dataset + Global Average Pooling + Classifier	58.76 %	60.78 %
VGG16 pre-trained on VGGFace Dataset + Flatten + Classifier	60.25 %	61.01 %
VGG16 pre-trained on VGGFace Dataset + Global Average Pooling + Classifier	68.78 %	69.67 %
VGG16 pre-trained on VGGFace Dataset + Attention + Global Average Pooling + Classifier	**70.78** %	**71.43** %

Figure 7: Experiments Results Table

Many models have been tested to reach the maximum accuracy possible. Resnet50 did not give high accuracy either the one pre-trained on ImageNet or VGGFace datasets. The VGG16 pre-trained on Imagenet also did not give promising results. InceptionV3 pre-trained on Imagenet dataset gave better results concerning validation accuracy, i.e. almost 59%. VGG16 pre-trained on VGGFace with Flatten gave valdiation accuracy of 60.25% however VGG16 pre-trained on VGGFace with Global Average Pooling gave the best accuracy of 68.78% among all. Knowing the fact that the attention technique improves the performance of the model, the VGG16 pretrained on VGGFace model was used as benchmark, then the attention technique was added after the VGG16 pre-trained on VGGFace dataset and before the Global Average Pooling which gave validation and test accuracy of 70.78% and 71.43% respectively by just addition of 66K extra parameters. The proposed model result was also compared against Kaggle Facial Expression Recognition Challenge leaderboard results. The leading team, who is at the top of the leaderboard, got 69.76% and 71.16% on the Public dataset and private dataset respectively which resembles the validation and and test set of our consideration, and our proposed model results seem to take a lead over them.

Conclusion and Future Works

In this paper, a deep learning framework using the VGG16 convolutional base, CBAM attention mechanism and a fully connected classifier is proposed to perform the task of facial expression recognition in images. Various convolutional base architectures were examined and the VGG16 trained on VGG Face dataset

was found to perform the best. Addition of the attention network to this was able to improve the system accuracy at the cost of very minimal added computations by attending to important channel and spatial parts of the feature map. The proposed system was also able to achieve higher accuracy than the top model in Kaggle Leaderboard for this task.

Even though the proposed system was able to outperform the best model available in Kaggle, the accuracy was still not very high. This might be because of some problems associated with the dataset such as: watermarks, mis-labeled images, absence of face in images, etc. So, the model accuracy could be improved if this dataset could be refined before training. Additionally, placing CBAM attention after each convolution layer of VGG16 convolutional base can be explored. Moreover, transformer network can be explored for giving attention to the specific facial regions.

References

[1] K. Wang, X. Peng, J. Yang, D. Meng, and Y. Qiao, "Region attention networks for pose and occlusion robust facial expression recognition," *IEEE Transactions on Image Processing*, vol. 29, pp. 4057–4069, 2020.

[2] S. Minaee and A. Abdolrashidi, "Deep-emotion: Facial expression recognition using attentional convolutional network," 2019.

[3] S. Samsani and V. A. Gottala, "A real-time automatic human facial expression recognition system using deep neural networks," in *Information and Communication Technology for Sustainable Development* (M. Tuba, S. Akashe, and A. Joshi, eds.), (Singapore), pp. 431–441, Springer Singapore, 2020.

[4] H. Siqueira, S. Magg, and S. Wermter, "Efficient facial feature learning with wide ensemble-based convolutional neural networks," *arXiv preprint arXiv:2001.06338*, 2020.

[5] P. Burkert, F. Trier, M. Z. Afzal, A. Dengel, and M. Liwicki, "Dexpression: Deep convolutional neural network for expression recognition," 2016.

[6] H. Zeng, X. Shu, Y. Wang, Y. Wang, L. Zhang, T. C. Pong, and H. Qu, "Emotioncues: Emotion-oriented visual summarization of classroom videos," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2019.

[7] P. Tzirakis, G. Trigeorgis, M. A. Nicolaou, B. W. Schuller, and S. Zafeiriou, "End-to-end multimodal emotion recognition using deep neural networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, p. 1301–1309, Dec 2017.

[8] N. Mehendale, "Facial emotion recognition using convolutional neural networks (ferc)," *SN Applied Sciences*, vol. 2, p. 446, Feb 2020.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," 2014.

[10] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," 2016.

[11] M. Usman, S. Latif, and J. Qadir, "Using deep autoencoders for facial expression recognition," in *2017 13th International Conference on Emerging Technologies (ICET)*, pp. 1–6, 2017.

[12] F. Farooq, J. Ahmed, and L. Zheng, "Facial expression recognition using hybrid features and self-organizing maps," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 409–414, 2017.

[13] S. Bellamkonda and N. Gopalan, "An enhanced facial expression recognition model using local feature fusion of gabor wavelets and local directionality patterns," *International Journal of Ambient Computing and Intelligence*, vol. 11, pp. 48–70, 11 2019.

[14] C. Gacav, B. Benligiray, and C. Topal, "Greedy search for descriptive spatial face features," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1497–1501, 2017.

[15] C. Masum Refat and N. Azlan, "Deep learning methods for facial expression recognition," 10 2019.

[16] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," 2018.

[17] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," 2015.

[18] W. Nash, T. Drummond, and N. Birbilis, "A review of deep learning in the study of materials degradation," *npj Materials Degradation*, vol. 2, 12 2018.