MONITOR THE EMPLOYEE'S EMOTIONAL WELL-BEING WITH PRE-TRAINED NEURAL NETWORK GOOGLENET USING THE FER2013 DATASET

CN7023 Coursework-Artificial Intelligence & Machine Vision

Module Leader
Dr Shaheen Khatoon

Brintha Thirunavukkarasu
Student ID 2555470

Table of Contents

1. Introduction

1.1. Objective ……………………………………………………………3

1.2. Overview……………………………………………………………3

2. Simulations

2.1 FER2013 (Facial Expression Recognition 2013) Dataset……………………4

2.2 Encode the dataset …………………………………………………...5

2.3 GoogleNet Architecture……………………………………………....6

2.3.1. Transfer Learning ………………………………………………7

2.3.2. Data Preprocessing ……………………………………………...8

3. Model Evaluation

3.1. Accuracy ……………………………………………………………...10

3.2. Accuracy curve ……………………………………………………….10

3.3. Confusion Matrix ……………………………………………………..12

4. Critical Analysis of results ………………………………………………...13

5. Conclusions ………………………………………………………………15

5.1. key takeaways…………………………………………………………...15

6. References……………………………………………………………………...16

7. MATLAB Certificates

7.1. Course 1: MATLAB Onramp……………………………………….......17

7.2. Course 2: Machine Learning Onramp ……………………………......17

7.3. Course 3: Deep Learning Onramp………………………………........18

7.4. Course 4: Image Processing Onramp……………………………...18

List of Figures

1. **Sample image of the FER2013 dataset……………………………………………………..5**
2. **Distribution of Images among five classes…………………………………………………5**
3. **GoogleNet Network Architecture (Szegedy et al., 2014)……………………………………6**
4. **Features of GoogleNet……………………………………………………………………7**
5. **Replaced Layer in GoogleNet……………………………………………………………8**
6. **Validation………………………………………………………………………………..10**
7. **Accuracy Curve and Loss Curve function………………………………………………11**
8. **Confusion matrix………………………………………………………………………..12**
9. **Test Labelling……………………………………………………………………………13**
10. **Fine Tuning Parameters and its value…………………………………………...14**

List of Tables

1. **FER2013 dataset information……………………………………………………4**

1. Introduction

In today's world, stress may seem like a normal part of daily living, but when it gets out of hand and negatively impacts a person's life over time, it can cause several significant health concerns. The Workplace Health Report indicates that 76% of employees report moderate-to-high or high levels of stress. According to Statistica data, the UK's inpatient hospital admissions due to illnesses related to stress cost approximately £8.13 billion (Pindar, 2022). Numerous companies have started stress-reduction initiatives for their staff members, and AI is developing new techniques that are simple to use and track the well-being of employees. Artificial intelligence combined with computer vision techniques can automatically and quickly identify different emotional states in an individual.

1.1 Objective

Convolutional neural networks CNNs), a deep learning technique used widely in the field of computer vision because of their capacity to autonomously acquire hierarchical features from images, renders them exceptionally well-suited for image classification. Among these characteristics (neural network accuracy, speed, and size), I chose GoogleNet architecture to monitor the symptoms of stress and depression state of an employee. Transfer learning is a technique that utilizes pre-trained neural networks such as GoogleNet for image classification to leverage learned features for novel tasks (Mustafa et al., 2017). This streamlines and expedites model training in comparison to the process of training from the beginning. The FER2013 dataset, also referred to as the Facial Expression Recognition 2013 dataset, is frequently used to train a model for classifying people's emotions. The objective of this strategy is to improve the accuracy of emotion identification, which could lead to the precise recognition of employee emotions.

1.2 Overview

A convolutional neural network will be trained in GoogleNet to classify images of emotions into their corresponding classes (anger, disgust, fear, happy, sad, surprise, neutral) via transfer learning and data augmentation. The FER2013 dataset (Amal et al., 2021), primarily focused on the development and evaluation of machine learning algorithms for problems related to facial expression recognition. A dataset is divided at random into 20% for testing and 80% for training. The images should be preprocessed, which includes scaling them to the proper input size for the GoogleNet model and converting them from grayscale to RGB format. Using the GoogleNet function in MATLAB's Deep Learning Toolbox, load the pre-trained GoogLeNet model. Adjust the GoogLeNet network to the specific requirement of images and establish the settings for the training set, such as the learning rate, optimizer, mini-batch size, and any other pertinent information. Use the train network function to train the modified GoogLeNet model. Finally, to evaluate the final trained model's performance in real-world circumstances, test it on untested data.

2. Simulations

2.1 FER2013 (Facial Expression Recognition 2013) Dataset

TABLE 1: FER2013 dataset information

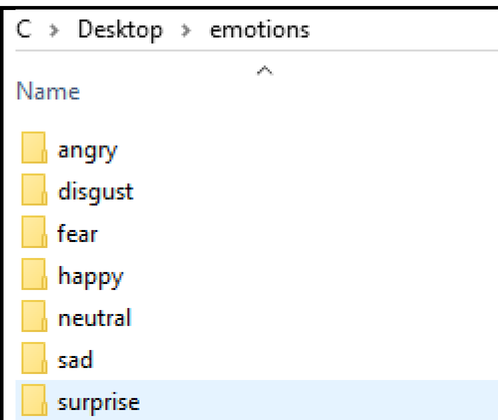
Link	[Emotion Detection (kaggle.com)]()	
Description	It contains 30,000 grayscale facial images of seven expressions. Due to automatic face registration, each image's face is roughly in the center and takes up the same amount of space.	
Size	48 x 48 pixel	
Categories/Classes	Angry Disgust Fear Happy Sad Surprise Neutral	
Total images Training set Validation set	30,000 28,709 3,589	

Fig.1.Sample image of FER2013 dataset

2.2 Encode the dataset

The FER2013 contains 30,000 images, it is hard to run in machines without higher computational power, preferably with GPU. I have taken 20% of the total which is 5899 images. Every class in the chosen dataset has almost the same number of images, except for the ‘disgust’ class. The input image size used for the training and validation is 243×243×3.

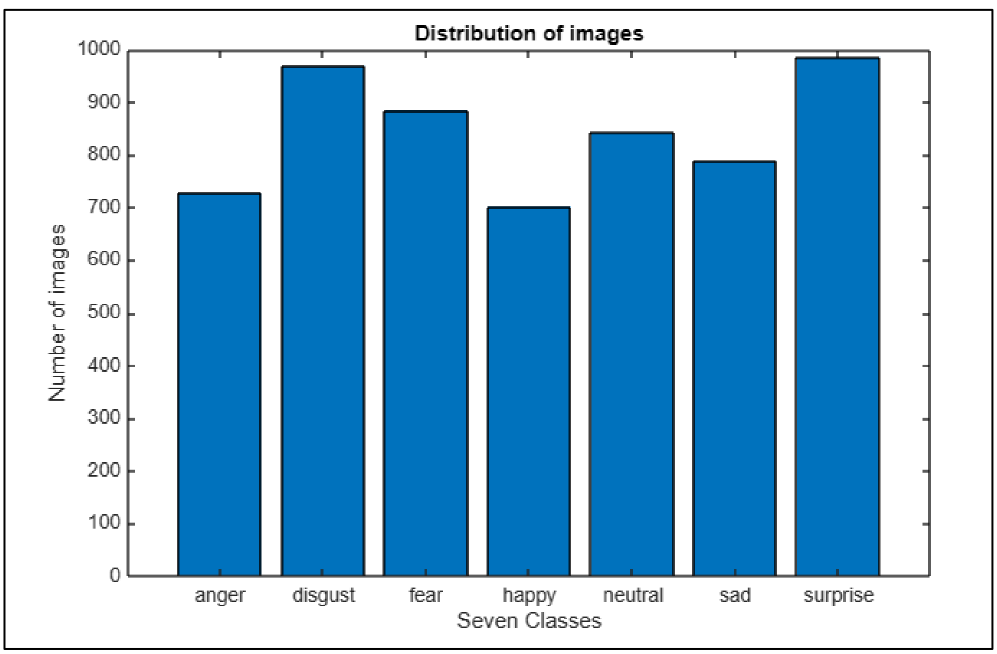


Fig.2 Distribution of Images among five classes

2.3 GoogleNet Architecture

Google Net, also known as Inception V1, is a 22-layer deep convolutional neural network that was developed by Google Research and won the 2014 ILSVRC image classification challenge. When compared to previous champions AlexNet, it has produced a considerably lower error rate. Having been trained on more than a million photos, GoogLeNet is capable of classifying photos into 1000 different object categories, including fruits, mats, houses, and a variety of expressions.

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Fig. 3. GoogleNet Network Architecture (Szegedy et al., 2014)

The network's design prioritizes computing efficiency and practicality, enabling inference to be executed on individual devices, even ones with restricted computational resources. This architecture uses RGB color channels of 224 by 224 image size and employs Rectified Linear Units (ReLU) act as the activation function for every convolution.

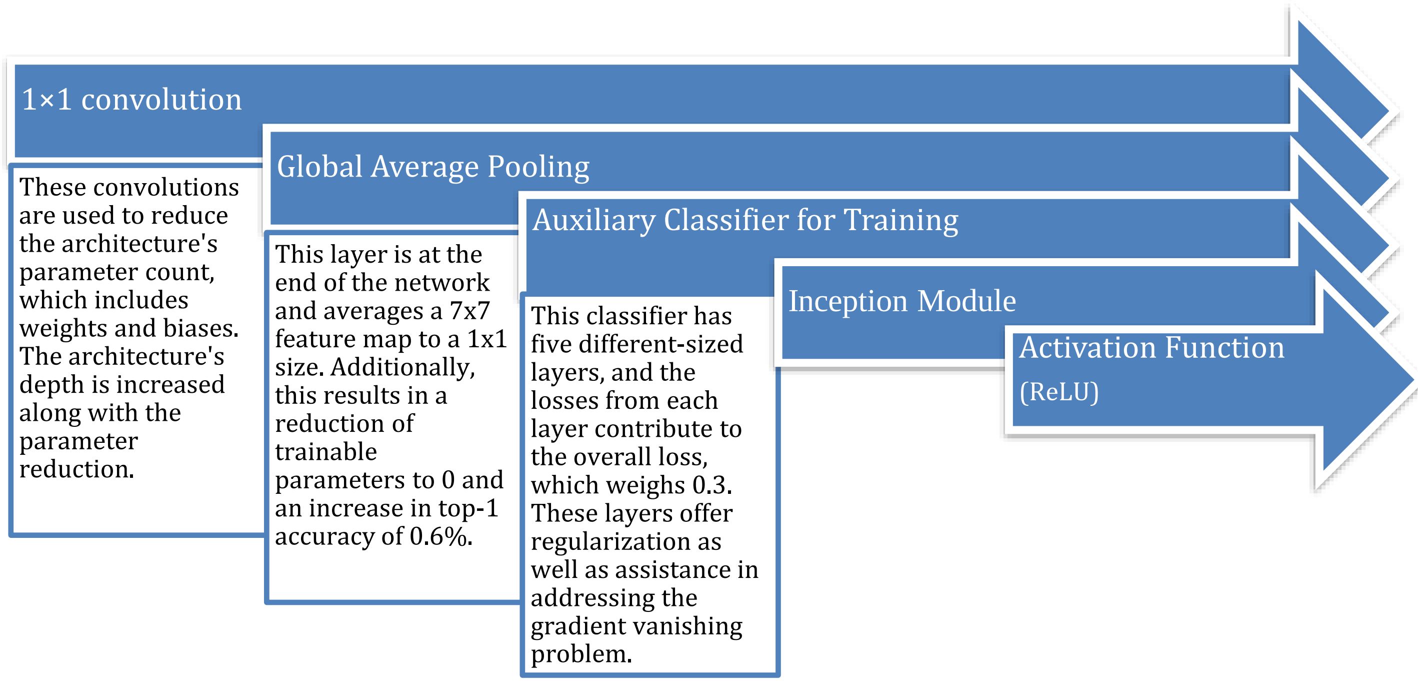


Fig.4.Features of GoogleNet

2.3.1. Transfer Learning

GoogleNet is a pre-trained network that is used to begin learning a new task. In general, transfer learning enables network fine-tuning faster and easier than starting from zero and training a network with randomly initialized weights (Shaees et al., 2020). Using fewer training images, you may rapidly apply learnt features to a new task. The input image is classified by the final classification layer and the learnable layer of the network using features extracted by the convolutional layers. 'loss3 classifier' and 'output' are the two layers of GoogleNet. Make a layer graph utilising the network that was trained to replace these layers.

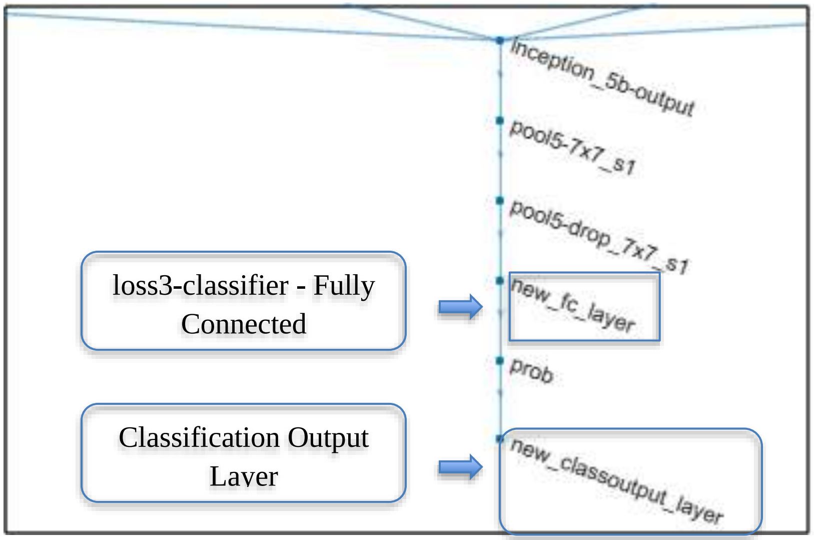


Fig.5. Replaced Layer in GoogleNet
(During training, the train network automatically configures the layer's output classes.)

2.3.2. Data Preprocessing

Freeze Initial Layers

To avoid the network's older layers from overfitting to the new dataset, freezing the weights of those levels for transfer learning and setting their learning rates to zero is the best option. Freeze the network's weights while leaving the final learnable layer unfrozen. In GoogleNet, the first 10 layers are frozen using *freezeWeights* function and using *createLgraphUsingConnections* to reconnect each layer to its initial position.

Function: *freezeWeights(layers(1:10);*

Normalize Pixel Values

Adjust the image pixel values so that they fall between [-1, 1] and [0, 1]. Better convergence occurs throughout training as a result.

Function: *pixelRange = [-30 30];*

Data Augmentation

This network needs input image size of 227×227×3, resizing the training and validation images using an augmented image datastore. Additionally, data augmentation keeps the network from overfitting and from learning every detail of the training images. To enhance the diversity of the training dataset and boost the model's capacity for generalization, such as random rotation, horizontal flipping, shifting, and zooming is used in image augmenter.

Function: *imageDataAugmenter*
RandXReflection
RandXTranslation, RandYTranslation,
RandXScale, RandYScale,
RandRotation,
RandXShear, RandYShear

Color Preprocessing

All the images in the FER2013 dataset are grayscale, to change into RGB images using the colour preprocessing function *gray2rgb.*
Function: *gray2rgb*

Training Options

Setting the starting learning rate to a low number to impede learning in the transferred layers.
To speed up learning in the new final layers, In the previous phase, boost the learning rate variables for the fully connected layer. Only the new layers learn quickly from this set-up of learning rate parameters, whereas the other layers learn more slowly. In transfer learning, fewer number epochs are enough to attain accuracy. Using Sgdm with momentum (stochastic gradient descent), lower the learning rate by a factor of 0.2 for every five epochs.
Function: *MiniBatchSize* = 10   *MaxEpochs* = 6

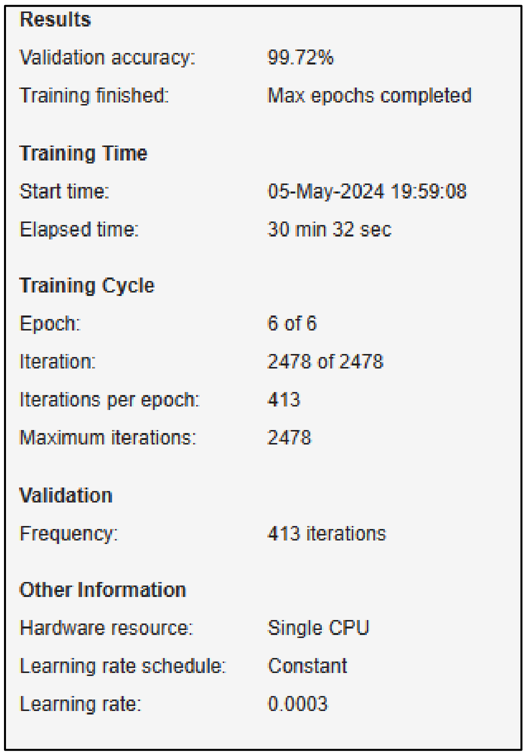
Implementation

Training the network with augmented training images and layers.
net = trainNetwork(augument_Train_image, replaced layer of lgraph, training_options);

3. Model Evaluation

3.1. Accuracy

The validation dataset achieved 99.72% accuracy and a log loss of 0.28% (Bodapati et al., 2021). In the experiment, the learning rate weighting factor is 0.003, the epoch is 6, the batch size is 10, the momentum (Momentum) is set to 0.9, and the SGD optimizer is used to optimize the loss with 2478 iterations.



Results	
Validation accuracy:	99.72%
Training finished:	Max epochs completed
Training Time	
Start time:	05-May-2024 19:59:08
Elapsed time:	30 min 32 sec
Training Cycle	
Epoch:	6 of 6
Iteration:	2478 of 2478
Iterations per epoch:	413
Maximum iterations:	2478
Validation	
Frequency:	413 iterations
Other Information	
Hardware resource:	Single CPU
Learning rate schedule:	Constant
Learning rate:	0.0003

Fig.6. Validation

3.2. Accuracy curve

Plotting the accuracy data acquired during training and validation epochs in MATLAB allowed for the visualization of the accuracy curve. This can assist you in determining any problems like overfitting or underfitting as well as how your model's accuracy changes over time.

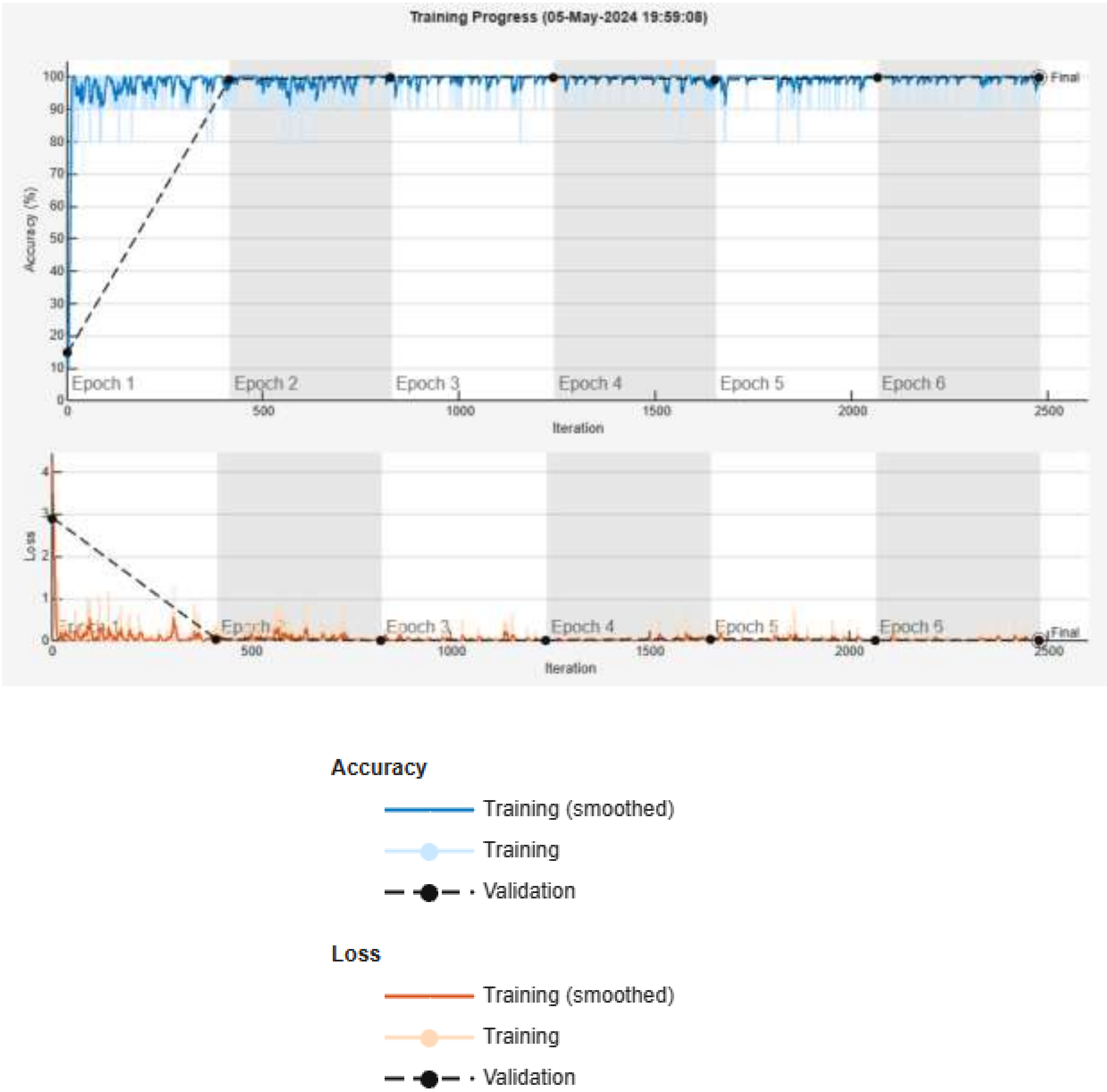


Fig.7 Accuracy Curve and Loss Curve function

Initially accuracy is 44 % and after tunning, accuracy is 99.72%, changed epochs and batch size for better results. Many articles suggest that accuracy for FER2013 tends to be less due to the image nature and but with GoogleNet network achieved higher accuracy than other CNN networks. After the first epoch, peaked at more than 90% accuracy.

3.3. Confusion matrix

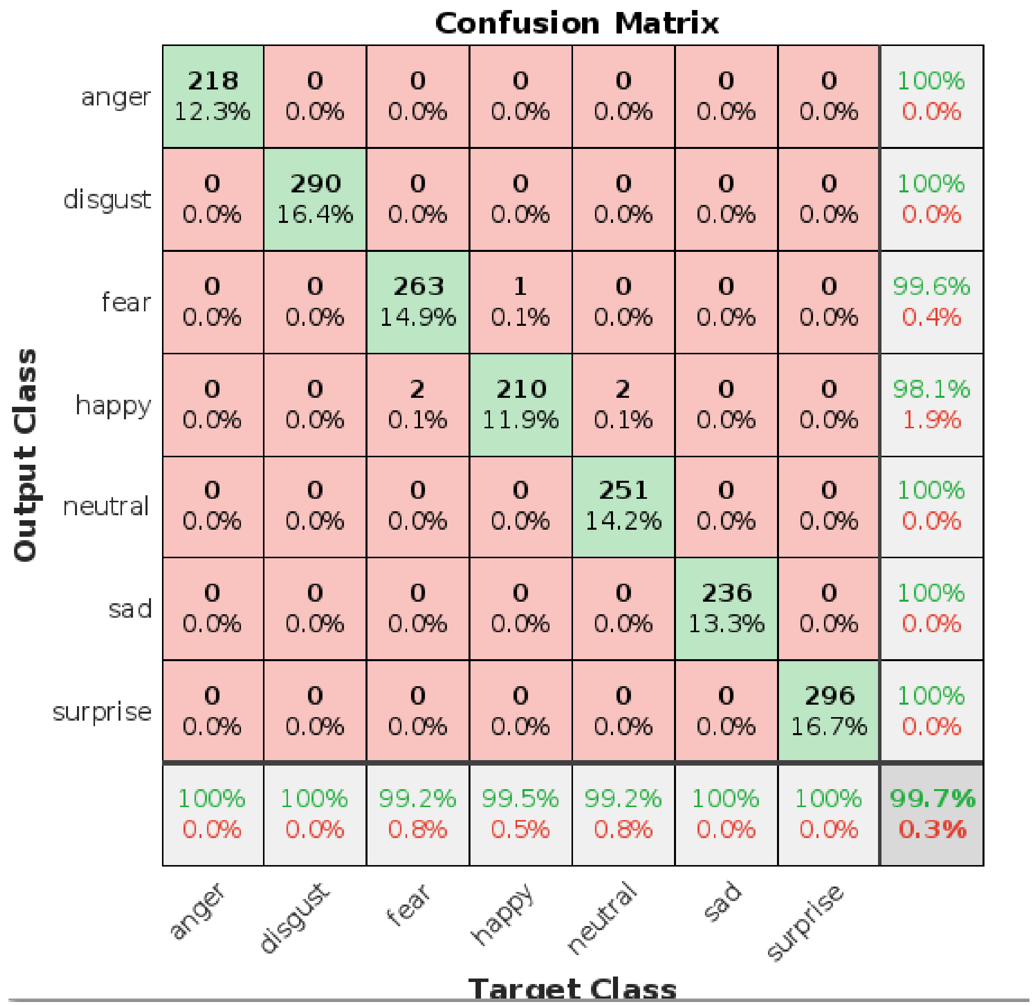


Fig.8. Confusion matrix

This confusion matrix appears to be showing the results of different emotions' classifications. The predicted emotion is represented by each column, and the actual emotion is indicated by each row. The count and percentage of instances where the actual emotion matches the predicted emotion are shown by the numbers in the matrix. For instance, the actual emotion "angry" is shown in the first row, and the greatest percentage in that row is 100%, meaning that 100% of the cases in which the actual emotion was "angry" were accurately identified as such. Off-diagonal elements indicate misclassifications, whereas diagonal elements typically indicate accurate predictions. The accuracy percentages on the right may represent overall accuracy or accuracy by class.

4. Critical Analysis of results

To achieve the desired objective of accurately classifying emotions using the GoogleNet architecture with transfer learning on the FER2013 dataset, the critical examination of the results entails evaluating the effectiveness of the deployed approaches and methods.

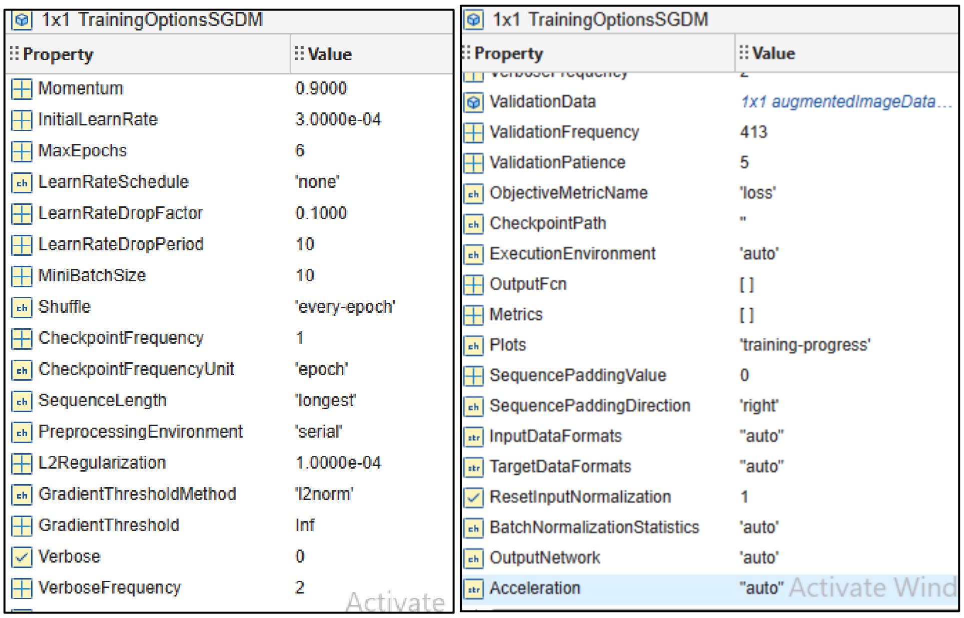
Fig.9. Test Labelling

- Firstly, it makes sense to use transfer learning from the pre-trained GoogleNet model to increase the accuracy of emotion identification by utilizing the features that have been learned from a large amount of data. The model can be adjusted to the particular goal of emotion classification by swapping out the final classification layers and fine-tuning the network. This increases efficiency and decreases the requirement for a large amount of training data and labeled correctly as expected.

Fine Tuning

- Additionally, the robustness and generalization capacity of the model is enhanced by data preprocessing methods such as pixel value normalization, data augmentation, and grayscale to RGB conversion. Random rotation, flipping, shifting, and zooming are examples of augmentation techniques that serve to diversify the training dataset, lower overfitting, and enhance the model's capacity to adjust to new data.

- By adjusting learning rate from 0.0001 to 0.0003, adding regularization techniques like L2, limiting the size of gradients, with all these changes achieved higher accuracy, batch normalization or dropout is not used.



1x1 TrainingOptionsSGDM

Property	Value
Momentum	0.9000
InitialLearnRate	3.0000e-04
MaxEpochs	6
LearnRateSchedule	'none'
LearnRateDropFactor	0.1000
LearnRateDropPeriod	10
MiniBatchSize	10
Shuffle	'every-epoch'
CheckpointFrequency	1
CheckpointFrequencyUnit	'epoch'
SequenceLength	'longest'
PreprocessingEnvironment	'serial'
L2Regularization	1.0000e-04
GradientThresholdMethod	'l2norm'
GradientThreshold	Inf
Verbose	0
VerboseFrequency	2

1x1 TrainingOptionsSGDM

Property	Value
ValidationData	1x1 augmentedImageData...
ValidationFrequency	413
ValidationPatience	5
ObjectiveMetricName	'loss'
CheckpointPath	''
ExecutionEnvironment	'auto'
OutputFcn	[ ]
Metrics	[ ]
Plots	'training-progress'
SequencePaddingValue	0
SequencePaddingDirection	'right'
InputDataFormats	"auto"
TargetDataFormats	"auto"
ResetInputNormalization	1
BatchNormalizationStatistics	'auto'
OutputNetwork	'auto'
Acceleration	"auto"

Fig.10. Fine Tuning Parameters and its value

- The success rate is 99.72% on the validation dataset, however, shows that there is still space for improvement despite these efforts. This limitation could be caused by several things, such as the FER2013 dataset's intrinsic difficulties, which include incorrectly labelled photos and non-face shots that can negatively impact model performance. Furthermore, the network architecture and hyperparameters (epoch, batch size, learning rate) selected might not be the best for this application.
- Through systematic experimentation, these settings could be changed to get better outcomes. To improve the model's convergence to a better solution, try increasing the number of epochs or modifying the learning rate schedule. Enhancing performance could also come from assembling several models or investigating different network topologies. Furthermore, using data augmentation methods or adding more preprocessing stages could help the model capture more nuanced facial emotions and increase classification accuracy.

5. Conclusions

The goal of this study was to track the emotional well-being of employees by classifying emotions using GoogleNet architecture with transfer learning on the FER2013 dataset. Even with the validation dataset's accuracy of 99.72%, issues like incorrect labelling and dataset diversity prevented better accuracy. Although data preprocessing methods and transfer learning were successful in improving model performance, more network architecture and hyperparameter optimization are required. The accuracy and resilience of the model can be enhanced for more accurate emotion recognition in real-world circumstances through systematic testing and refinement, including modifying epochs and learning rates and investigating alternate network topologies.

5.1. Key Takeaways

1. Transfer learning from trained models, such as GoogleNet, improves performance and decreases the amount of training data required.
2. Models for classifying people's emotions are often trained using the FER2013 dataset. It's a great tool for raising the accuracy of emotion recognition in work pertaining to worker wellbeing.
3. To increase the level of accuracy of identifying emotions in work environments, network design and hyperparameters must be continuously experimented with and refined.
4. The main goal of this approach is to increase the precision of employee emotion detection by improving the accuracy of emotion identification. This may have important ramifications for comprehending and managing mental health problems in the workplace.

6. Reference

1. Amal, V. S., Suresh, S., & Deepa, G. (2021). Real-Time Emotion Recognition from Facial Expressions Using Convolutional Neural Network with Fer2013 Dataset. *Smart Innovation, Systems and Technologies*, 541–551. https://doi.org/10.1007/978-981-16-3675-2_41.
2. Bodapati, J. D., Srilakshmi, U., & Veeranjaneyulu, N. (2021). FERNet: A Deep CNN Architecture for Facial Expression Recognition in the Wild. *Journal of the Institution of Engineers (India): Series B*. https://doi.org/10.1007/s40031-021-00681-8.
3. Mustafa, Ahmed Abdalazeem Ahmed, & Tarig Ahmed Khalid. (2017). Benchmark analysis of popular ImageNet classification deep CNN architectures. 2017 *International Conference on Smart Technologies for Smart Nation (SmartTechCon)*. https://doi.org/10.1109/smarttechcon.2017.8358502.
4. Pindar, J. (2022, June 17). Stress Statistics UK | 2022 Data. Champion Health. https://championhealth.co.uk/insights/stress-statistics/.
5. Shaees, S., Naeem, H., Arslan, M., Naeem, M. R., Ali, S. H., & Aldabbas, H. (2020, September 1). *Facial Emotion Recognition Using Transfer Learning*. IEEE Xplore. https://doi.org/10.1109/ICCIT-144147971.2020.9213757.
6. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Dumitru Erhan, Vanhoucke, V., & Rabinovich, A. (2014). Going Deeper with Convolutions. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.1409.4842.

7. MATLAB Certificates

Course 1: MATLAB Onramp

MathWorks | Training Services

Course Completion Certificate

Brintha Thirunavukkarasu

has successfully completed **100%** of the self-paced training course

MATLAB Onramp

DIRECTOR, TRAINING SERVICES

21 February 2024

Course 2: Machine Learning Onramp

MathWorks | Training Services

Course Completion Certificate

Brintha Thirunavukkarasu

has successfully completed **100%** of the self-paced training course

Machine Learning Onramp

DIRECTOR, TRAINING SERVICES

17 March 2024

Course 3: Deep Learning Onramp

MathWorks | Training Services

Course Completion Certificate

Brintha Thirunavukkarasu

has successfully completed **100%** of the self-paced training course

Deep Learning Onramp

DIRECTOR, TRAINING SERVICES

17 March 2024

Course 4: Image Processing Onramp

MathWorks | Training Services

Course Completion Certificate

Brintha Thirunavukkarasu

has successfully completed **100%** of the self-paced training course

Image Processing Onramp

DIRECTOR, TRAINING SERVICES

19 March 2024