Registration No: 230118234

Contents

Registration No: 230118234

Introduction

(NOTE: Functions for Bayesian Optimisation and Spectrogram Augmentation are at the end)

The goal of this project is to design, implement, and evaluate an ASR system utilizing deep learning methodologies. Starting with a basic CNN model, the project iteratively incorporates advanced techniques such as grid search, model averaging, and Bayesian optimization to improve accuracy and robustness. The dataset comprises spectrogram images representing different speech commands, which the model learns to classify. The process involves preprocessing the data, defining and training the model, and systematically tuning the architecture and hyperparameters to meet performance benchmarks.

Preparing the environment and preprocessing the spectrogram images for training and validation.

```
clear all;

% Define paths to your dataset
dataFolder = 'speechImageData';
trainDataFolder = fullfile(dataFolder, 'TrainData');
valDataFolder = fullfile(dataFolder, 'ValData');

% Create imageDatastores for training and validation datasets
adsTrain = imageDatastore(trainDataFolder, 'IncludeSubfolders', true, 'LabelSource',
'foldernames');
adsValidation = imageDatastore(valDataFolder, 'IncludeSubfolders', true, 'LabelSource',
'foldernames');

% Define input size and resize the images in the datastores
inputSize = [98 50 1];
adsTrain.ReadFcn = @(x)imresize(imread(x), inputSize(1:2));
adsValidation.ReadFcn = @(x)imresize(imread(x), inputSize(1:2));
```

Registration No: 230118234

Task 1: Baseline Model

Objective: Establish a baseline model with a deep convolutional network architecture.

Description: The baseline model is the starting point for the ASR system. It includes fundamental layers such as convolutional, batch normalization, max pooling, and dropout layers arranged in a sequential manner to process the input spectrogram images.

5 convolutional layers and 16 filters (which is doubled for the second and quadrupled for the remaining layers) are used. Alongside, maxpooling in the first 3 layers is used to downsample the feature maps 'spatially' and a final maxpooling layer with timepooling pools the input feature map globally over time. An accuracy of around 75% is obtained, which is satisfactory as much over 60%.

```
fprintf('Starting Basic Approach...\n');
```

```
Starting Basic Approach...
```

```
% Define Network Architecture
numClasses = numel(categories(adsTrain.Labels)); % Use adsTrain to determine the number of classes dynamically
timePoolSize = ceil(inputSize(1)/8);
numF = 16; % Number of filters for the convolutional layers
dropoutProb = 0.2; % Dropout probability

layers = [
    imageInputLayer(inputSize)

    convolution2dLayer(3, numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

    convolution2dLayer(3, 2*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

    convolution2dLayer(3, 4*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

    convolution2dLayer(3, 4*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer

    convolution2dLayer(3, 4*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer([timePoolSize, 1])
    dropoutLayer(dropoutProb)
```

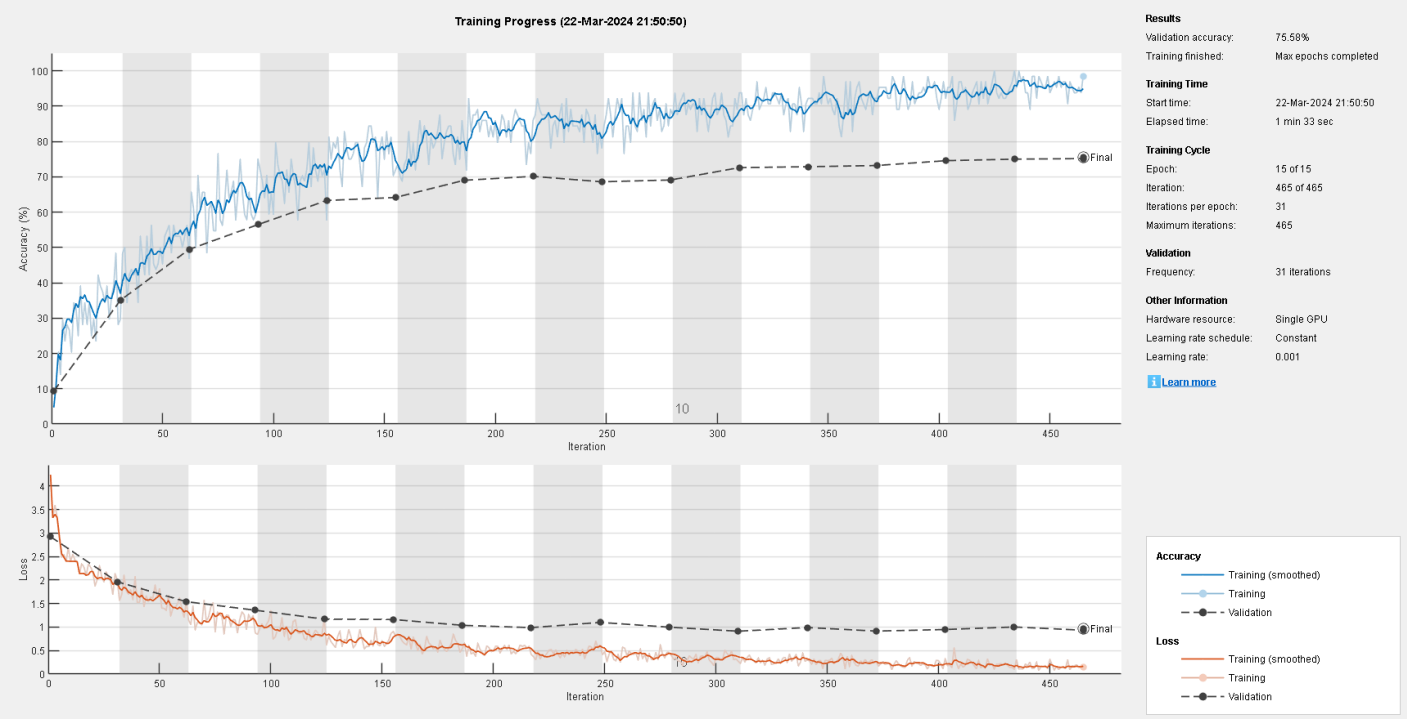
Registration No: 230118234

```
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];

% Specify Training Options
minibatchsize = 64;
options = trainingOptions('adam', ...
    'InitialLearnRate', 0.001, ...
    'MaxEpochs', 15, ...
    'MiniBatchSize', minibatchsize, ...
    'Shuffle', 'every-epoch', ...
    'Plots', 'training-progress', ...
    'Verbose', true, ...
    'ValidationData', adsValidation, ...
    'ValidationFrequency', floor(numel(adsTrain.Files)/minibatchsize));

% Train Network
trainedNet = trainNetwork(adsTrain, layers, options);
```

```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        4.69% |        9.31% |       4.2394 |       2.9211 |
0.0010 |
|       1 |          31 |       00:00:11 |       29.69% |       35.01% |       1.9933 |       1.9623 |
0.0010 |
|       2 |          50 |       00:00:13 |       45.31% |              |       1.8169 |              |
0.0010 |
|       2 |          62 |       00:00:17 |       46.88% |       49.44% |       1.5456 |       1.5394 |
0.0010 |
|       3 |          93 |       00:00:22 |       75.00% |       56.45% |       0.8759 |       1.3575 |
0.0010 |
|       4 |         100 |       00:00:23 |       59.38% |              |       1.3508 |              |
0.0010 |
|       4 |         124 |       00:00:28 |       57.81% |       63.28% |       1.1751 |       1.1674 |
0.0010 |
|       5 |         150 |       00:00:31 |       75.00% |              |       0.7099 |              |
0.0010 |
|       5 |         155 |       00:00:33 |       71.88% |       64.13% |       0.9039 |       1.1600 |
0.0010 |
|       6 |         186 |       00:00:39 |       79.69% |       69.00% |       0.5393 |       1.0366 |
0.0010 |
|       7 |         200 |       00:00:41 |       85.94% |              |       0.4945 |              |
0.0010 |
|       7 |         217 |       00:00:45 |       85.94% |       70.11% |       0.4144 |       0.9876 |
0.0010 |
```

Registration No: 230118234

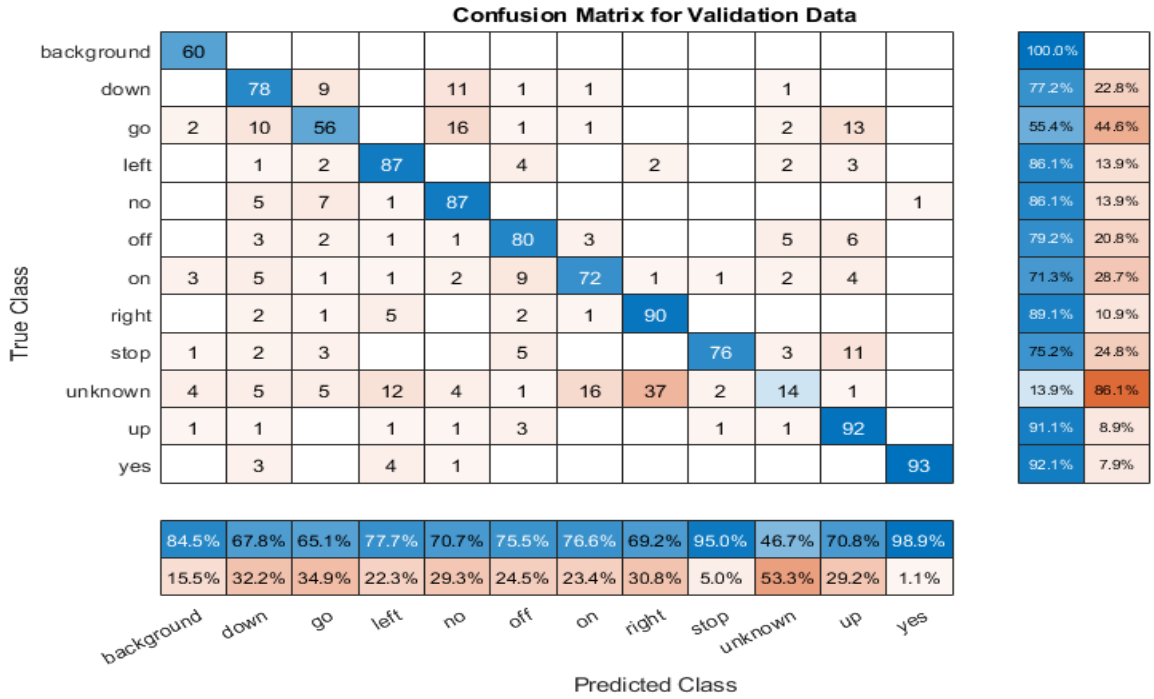
```
|       8 |         248 |       00:00:51 |       78.12% |       68.57% |       0.6147 |       1.0983 |
0.0010 |
|       9 |         250 |       00:00:51 |       87.50% |              |       0.4782 |              |
0.0010 |
|       9 |         279 |       00:00:57 |       89.06% |       69.09% |       0.4600 |       0.9968 |
0.0010 |
|      10 |         300 |       00:01:00 |       87.50% |              |       0.3499 |              |
0.0010 |
|      10 |         310 |       00:01:03 |       93.75% |       72.59% |       0.2723 |       0.9111 |
0.0010 |
|      11 |         341 |       00:01:08 |       92.19% |       72.84% |       0.3020 |       0.9894 |
0.0010 |
|      12 |         350 |       00:01:10 |       90.62% |              |       0.2866 |              |
0.0010 |
|      12 |         372 |       00:01:14 |       92.19% |       73.19% |       0.2706 |       0.9140 |
0.0010 |
|      13 |         400 |       00:01:18 |       89.06% |              |       0.2839 |              |
0.0010 |
|      13 |         403 |       00:01:20 |       92.19% |       74.55% |       0.3275 |       0.9472 |
0.0010 |
|      14 |         434 |       00:01:25 |      100.00% |       74.98% |       0.1006 |       1.0011 |
0.0010 |
|      15 |         450 |       00:01:27 |       93.75% |              |       0.3145 |              |
0.0010 |
|      15 |         465 |       00:01:31 |       98.44% |       75.15% |       0.1415 |       0.9322 |
0.0010 |
|=========================================================================================================
=============|
Training finished: Max epochs completed.
```




```
% Plot Confusion Matrix for Validation Set
YValidation = classify(trainedNet, adsValidation);
TValidation = adsValidation.Labels;
figure('Units', 'normalized', 'Position', [0.2 0.2 0.5 0.5]);
```

Registration No: 230118234

```
cm = confusionchart(TValidation, YValidation, ...
    'Title', 'Confusion Matrix for Validation Data', ...
    'ColumnSummary', 'column-normalized', 'RowSummary', 'row-normalized');
sortClasses(cm, categories(adsTrain.Labels));
```



Registration No: 230118234

Task 2: Grid Search

Objective: Utilize Grid Search to explore and optimize the model's architecture, focusing on the number of convolutional layers and filters per layer, to enhance validation accuracy.

Description: In this task, the model undergoes a systematic optimization process, exploring combinations of 3, 4, or 5 convolutional layers and 16, 24, or 32 filters per layer. Each configuration is constructed and evaluated for performance using Adam optimizer, targeting settings like a 0.001 learning rate, 64 mini-batch size, and up to 20 training epochs. The evaluation focuses on validation accuracy to determine the most effective architecture.

The grid search concludes by identifying and reporting the architecture that achieves the highest accuracy on the validation set. A confusion matrix of this optimal model provides insights into its classification capabilities across different speech commands. This concise approach aims to find the best balance between model complexity and generalization, enhancing performance on new data.

Early stopping is used to enhance computational time; the training stops if the validation loss does not improve for 5 epochs. The accuracy is higher by a percentage or two *without* early stopping, which is expected.

```
fprintf('Starting Grid Search Approach...\n');
```

```
Starting Grid Search Approach...
```

```
% Grid Search Parameters
numLayersOptions = [3, 4, 5];
numFiltersOptions = [16, 24, 32];

% Specify Training Options
minibatchsize = 64;
options = trainingOptions('adam', ...
    'InitialLearnRate', 0.001, ...
    'MaxEpochs', 20, ...
    'MiniBatchSize', minibatchsize, ...
    'Shuffle', 'every-epoch', ...
    'Plots', 'training-progress', ...
    'Verbose', true, ...
    'ValidationData', adsValidation, ...
    'ValidationFrequency', floor(numel(adsTrain.Files)/minibatchsize), ...
    'ValidationPatience', 5, ... % Early Stopping
    'ExecutionEnvironment', 'auto'); % Utilize GPU if available

% Initialize variables to store the best model's details
bestAccuracy = 0;
bestModel = [];
bestNumLayers = 0;
bestNumFilters = 0;

% Define Network Architecture and Train Models
for numLayers = numLayersOptions
    for numFilters = numFiltersOptions
        numClasses = numel(categories(adsTrain.Labels));
        timePoolSize = ceil(inputSize(1)/8);
```

Registration No: 230118234

```
        dropoutProb = 0.2;

        layers = [
            imageInputLayer(inputSize)
        ];

        for l = 1:numLayers
            filters = numFilters * (2^(l-1));
            layers = [
                layers;
                convolution2dLayer(3, filters, 'Padding', 'same')
                batchNormalizationLayer
                reluLayer
                maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')
            ];
        end

        layers = [
            layers;
            maxPooling2dLayer([timePoolSize, 1], 'Padding', 'same')
            dropoutLayer(dropoutProb)
            fullyConnectedLayer(numClasses)
            softmaxLayer
            classificationLayer
        ];

        % Train Network
        trainedNet = trainNetwork(adsTrain, layers, options);

        % Evaluate Model
        YValidation = classify(trainedNet, adsValidation);
        TValidation = adsValidation.Labels;
        accuracy = sum(YValidation == TValidation) / numel(TValidation);
        fprintf('Layers: %d, Initial Filters: %d, Validation Accuracy: %.2f%%\n', numLayers,
numFilters, accuracy * 100);

        % Update best model if current model is better
        if accuracy > bestAccuracy
            bestAccuracy = accuracy;
            bestModel = trainedNet;
            bestNumLayers = numLayers;
            bestNumFilters = numFilters;
        end
    end
end
```

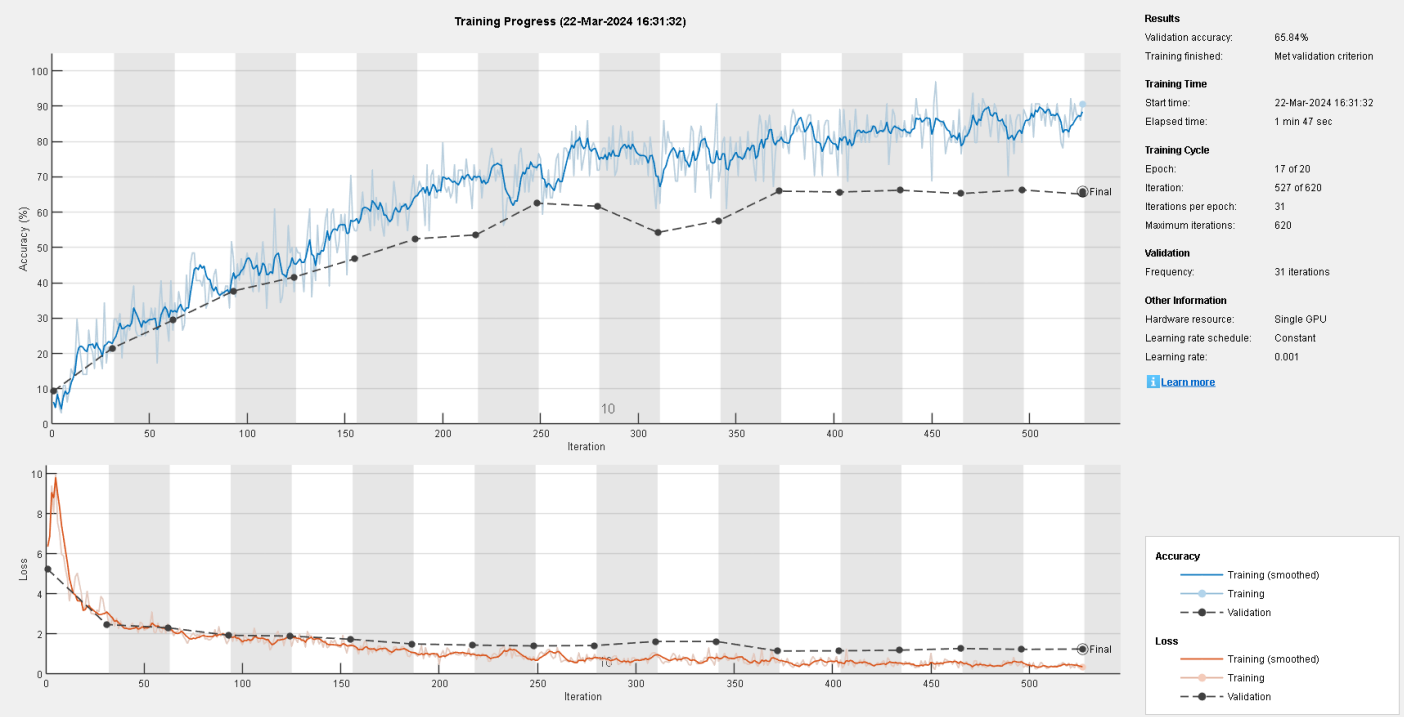
```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
============|
```

Registration No: 230118234

Epoch	Iteration	Time Elapsed	Mini-batch	Validation	Mini-batch	Validation	Base Learning
		(hh:mm:ss)	Accuracy	Accuracy	Loss	Loss	Rate
1	1	00:00:05	6.25%	9.31%	6.3612	5.2250	0.0010
1	31	00:00:11	21.88%	21.43%	3.0927	2.4720	0.0010
2	50	00:00:14	28.12%		2.1765		0.0010
2	62	00:00:17	26.56%	29.38%	2.5110	2.3124	0.0010
3	93	00:00:22	46.88%	37.66%	1.6895	1.9353	0.0010
4	100	00:00:24	48.44%		1.4639		0.0010
4	124	00:00:29	37.50%	41.59%	1.9622	1.8995	0.0010
5	150	00:00:32	59.38%		1.3326		0.0010
5	155	00:00:34	54.69%	46.80%	1.4621	1.7375	0.0010
6	186	00:00:40	64.06%	52.43%	1.0661	1.5047	0.0010
7	200	00:00:42	79.69%		0.4939		0.0010
7	217	00:00:46	68.75%	53.54%	1.0252	1.4460	0.0010
8	248	00:00:52	71.88%	62.51%	0.8728	1.4067	0.0010
9	250	00:00:53	75.00%		0.8177		0.0010
9	279	00:00:59	71.88%	61.66%	0.9827	1.4249	0.0010
10	300	00:01:02	73.44%		0.7459		0.0010
10	310	00:01:04	70.31%	54.23%	1.0466	1.6271	0.0010
11	341	00:01:10	68.75%	57.56%	1.0223	1.6276	0.0010
12	350	00:01:12	79.69%		0.7825		0.0010
12	372	00:01:16	81.25%	65.93%	0.6290	1.1514	0.0010
13	400	00:01:20	82.81%		0.5326		0.0010
13	403	00:01:22	89.06%	65.67%	0.3437	1.1600	0.0010
14	434	00:01:28	81.25%	66.18%	0.5759	1.1925	0.0010
15	450	00:01:30	68.75%		1.0251		0.0010
15	465	00:01:33	76.56%	65.24%	0.6448	1.2745	0.0010
16	496	00:01:39	79.69%	66.27%	0.6739	1.2332	0.0010

Registration No: 230118234

```
|      17 |         500 |       00:01:40 |       84.38% |              |       0.5613 |              |
0.0010 |
|      17 |         527 |       00:01:45 |       90.62% |       65.07% |       0.3647 |       1.2499 |
0.0010 |
|=========================================================================================================
=============|
Training finished: Met validation criterion.
```



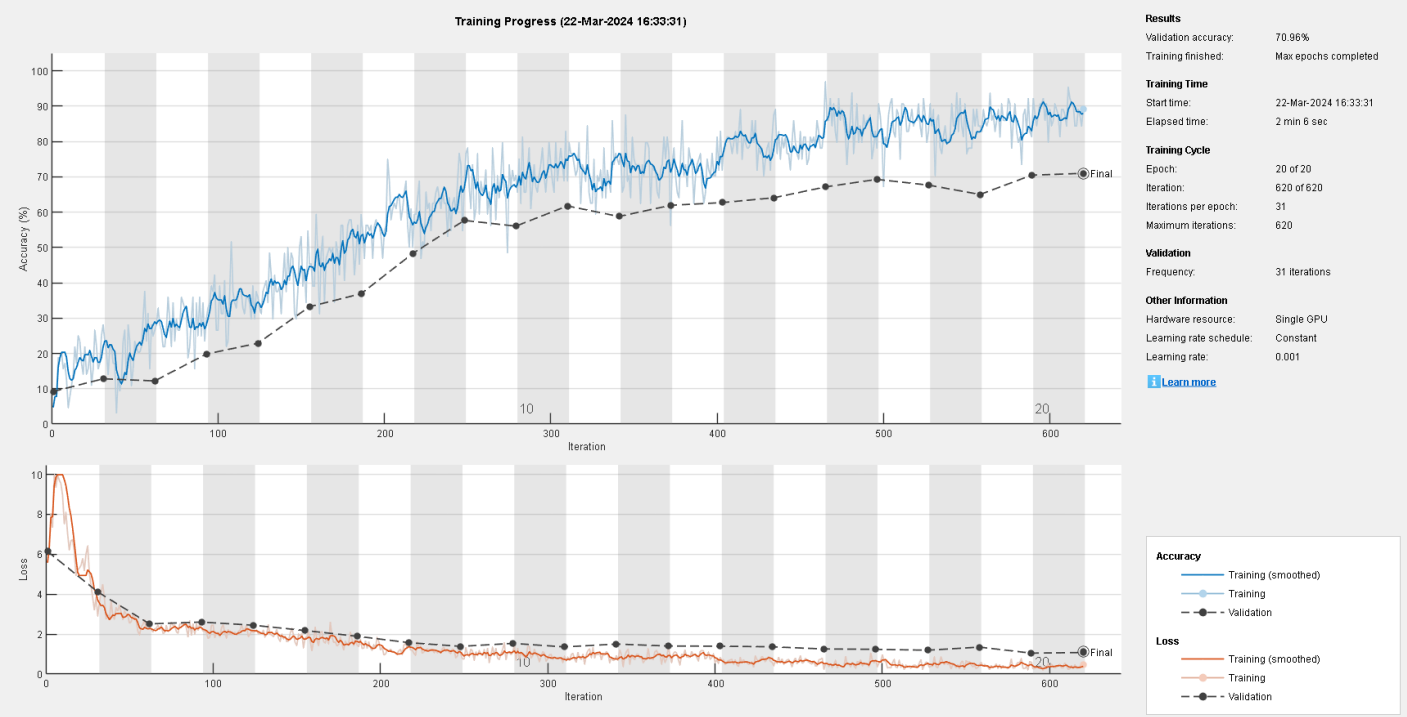

```
Layers: 3, Initial Filters: 16, Validation Accuracy: 65.84%
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        4.69% |        9.22% |       5.5707 |       6.1418 |
0.0010 |
|       1 |          31 |       00:00:09 |       28.12% |       12.81% |       2.9134 |       4.1143 |
0.0010 |
|       2 |          50 |       00:00:12 |       23.44% |              |       2.5481 |              |
0.0010 |
|       2 |          62 |       00:00:15 |       26.56% |       12.21% |       2.4002 |       2.5134 |
0.0010 |
|       3 |          93 |       00:00:21 |       29.69% |       19.81% |       2.3265 |       2.6042 |
0.0010 |
|       4 |         100 |       00:00:23 |       26.56% |              |       2.4442 |              |
0.0010 |
|       4 |         124 |       00:00:28 |       35.94% |       22.89% |       2.0781 |       2.4456 |
0.0010 |
|       5 |         150 |       00:00:31 |       53.12% |              |       1.4216 |              |
0.0010 |
```

Registration No: 230118234

```
|       5 |         155 |       00:00:33 |       45.31% |       33.22% |       1.8215 |       2.1796 |
0.0010 |
|       6 |         186 |       00:00:39 |       59.38% |       36.89% |       1.4540 |       1.8933 |
0.0010 |
|       7 |         200 |       00:00:41 |       54.69% |              |       1.2079 |              |
0.0010 |
|       7 |         217 |       00:00:45 |       50.00% |       48.25% |       1.4934 |       1.5733 |
0.0010 |
|       8 |         248 |       00:00:51 |       67.19% |       57.64% |       0.9797 |       1.3872 |
0.0010 |
|       9 |         250 |       00:00:52 |       78.12% |              |       0.7516 |              |
0.0010 |
|       9 |         279 |       00:00:57 |       70.31% |       56.02% |       1.0667 |       1.5377 |
0.0010 |
|      10 |         300 |       00:01:00 |       73.44% |              |       0.7763 |              |
0.0010 |
|      10 |         310 |       00:01:04 |       68.75% |       61.74% |       0.9005 |       1.3586 |
0.0010 |
|      11 |         341 |       00:01:09 |       73.44% |       58.75% |       0.9239 |       1.5030 |
0.0010 |
|      12 |         350 |       00:01:11 |       75.00% |              |       0.9450 |              |
0.0010 |
|      12 |         372 |       00:01:15 |       56.25% |       61.91% |       1.3380 |       1.4043 |
0.0010 |
|      13 |         400 |       00:01:19 |       81.25% |              |       0.5729 |              |
0.0010 |
|      13 |         403 |       00:01:21 |       71.88% |       62.68% |       0.8313 |       1.3954 |
0.0010 |
|      14 |         434 |       00:01:27 |       87.50% |       64.05% |       0.5957 |       1.3649 |
0.0010 |
|      15 |         450 |       00:01:30 |       82.81% |              |       0.6972 |              |
0.0010 |
|      15 |         465 |       00:01:33 |       96.88% |       67.12% |       0.2318 |       1.2575 |
0.0010 |
|      16 |         496 |       00:01:40 |       78.12% |       69.26% |       0.6910 |       1.2387 |
0.0010 |
|      17 |         500 |       00:01:41 |       67.19% |              |       0.9677 |              |
0.0010 |
|      17 |         527 |       00:01:46 |       82.81% |       67.63% |       0.5649 |       1.1942 |
0.0010 |
|      18 |         550 |       00:01:49 |       76.56% |              |       0.6630 |              |
0.0010 |
|      18 |         558 |       00:01:52 |       84.38% |       64.99% |       0.4813 |       1.3500 |
0.0010 |
|      19 |         589 |       00:01:58 |       81.25% |       70.45% |       0.6017 |       1.0481 |
0.0010 |
|      20 |         600 |       00:02:00 |       89.06% |              |       0.3885 |              |
0.0010 |
|      20 |         620 |       00:02:04 |       89.06% |       70.96% |       0.4786 |       1.0796 |
0.0010 |
|=============================================================================================================
=============|
```

Registration No: 230118234

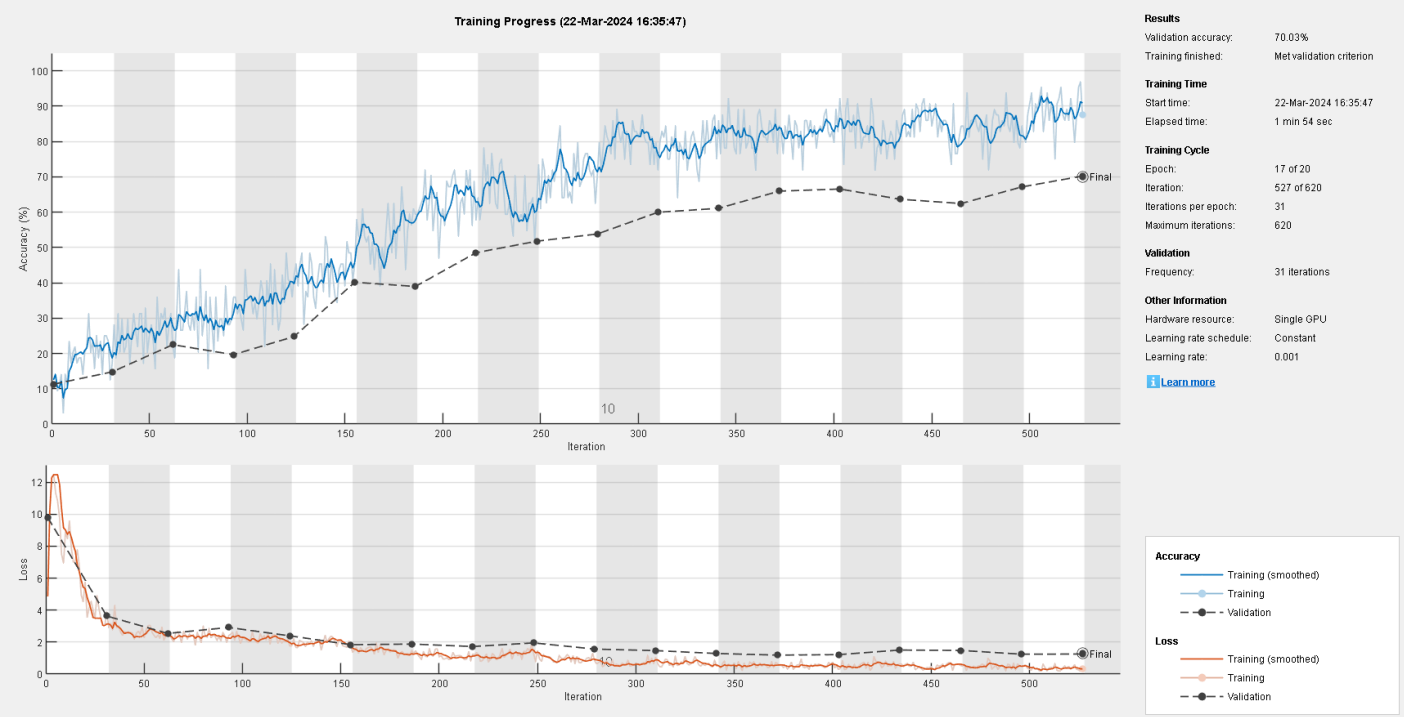
```
Training finished: Max epochs completed.
```




```
Layers: 3, Initial Filters: 24, Validation Accuracy: 70.96%
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:03 |       12.50% |       11.27% |       4.8558 |       9.7921 |
0.0010 |
|       1 |          31 |       00:00:10 |       17.19% |       14.77% |       3.8850 |       3.6288 |
0.0010 |
|       2 |          50 |       00:00:12 |       25.00% |              |       2.2700 |              |
0.0010 |
|       2 |          62 |       00:00:16 |       21.88% |       22.46% |       2.6720 |       2.5438 |
0.0010 |
|       3 |          93 |       00:00:23 |       35.94% |       19.64% |       2.4092 |       2.9134 |
0.0010 |
|       4 |         100 |       00:00:24 |       35.94% |              |       2.2328 |              |
0.0010 |
|       4 |         124 |       00:00:29 |       42.19% |       24.77% |       1.7444 |       2.3923 |
0.0010 |
|       5 |         150 |       00:00:33 |       39.06% |              |       2.0413 |              |
0.0010 |
|       5 |         155 |       00:00:35 |       45.31% |       40.14% |       1.7789 |       1.8363 |
0.0010 |
|       6 |         186 |       00:00:41 |       59.38% |       38.94% |       1.4794 |       1.8770 |
0.0010 |
|       7 |         200 |       00:00:44 |       60.94% |              |       1.0817 |              |
0.0010 |
```

```
|       7 |         217 |       00:00:48 |       57.81% |       48.51% |       1.2561 |       1.7229 |
0.0010 |
|       8 |         248 |       00:00:54 |       62.50% |       51.75% |       1.2605 |       1.9501 |
0.0010 |
|       9 |         250 |       00:00:55 |       60.94% |              |       1.4646 |              |
0.0010 |
|       9 |         279 |       00:01:01 |       70.31% |       53.80% |       0.8345 |       1.5625 |
0.0010 |
|      10 |         300 |       00:01:04 |       81.25% |              |       0.6547 |              |
0.0010 |
|      10 |         310 |       00:01:07 |       71.88% |       59.95% |       1.0436 |       1.4663 |
0.0010 |
|      11 |         341 |       00:01:13 |       85.94% |       61.06% |       0.5078 |       1.2868 |
0.0010 |
|      12 |         350 |       00:01:15 |       76.56% |              |       0.6538 |              |
0.0010 |
|      12 |         372 |       00:01:20 |       84.38% |       65.93% |       0.6514 |       1.1848 |
0.0010 |
|      13 |         400 |       00:01:24 |       79.69% |              |       0.7499 |              |
0.0010 |
|      13 |         403 |       00:01:26 |       92.19% |       66.52% |       0.2808 |       1.2150 |
0.0010 |
|      14 |         434 |       00:01:33 |       81.25% |       63.62% |       0.6215 |       1.5010 |
0.0010 |
|      15 |         450 |       00:01:35 |       84.38% |              |       0.4636 |              |
0.0010 |
|      15 |         465 |       00:01:39 |       84.38% |       62.43% |       0.5344 |       1.4794 |
0.0010 |
|      16 |         496 |       00:01:45 |       84.38% |       67.12% |       0.4452 |       1.2436 |
0.0010 |
|      17 |         500 |       00:01:46 |       90.62% |              |       0.2917 |              |
0.0010 |
|      17 |         527 |       00:01:52 |       87.50% |       70.20% |       0.3393 |       1.2522 |
0.0010 |
|=============================================================================================================
=============|
Training finished: Met validation criterion.
```

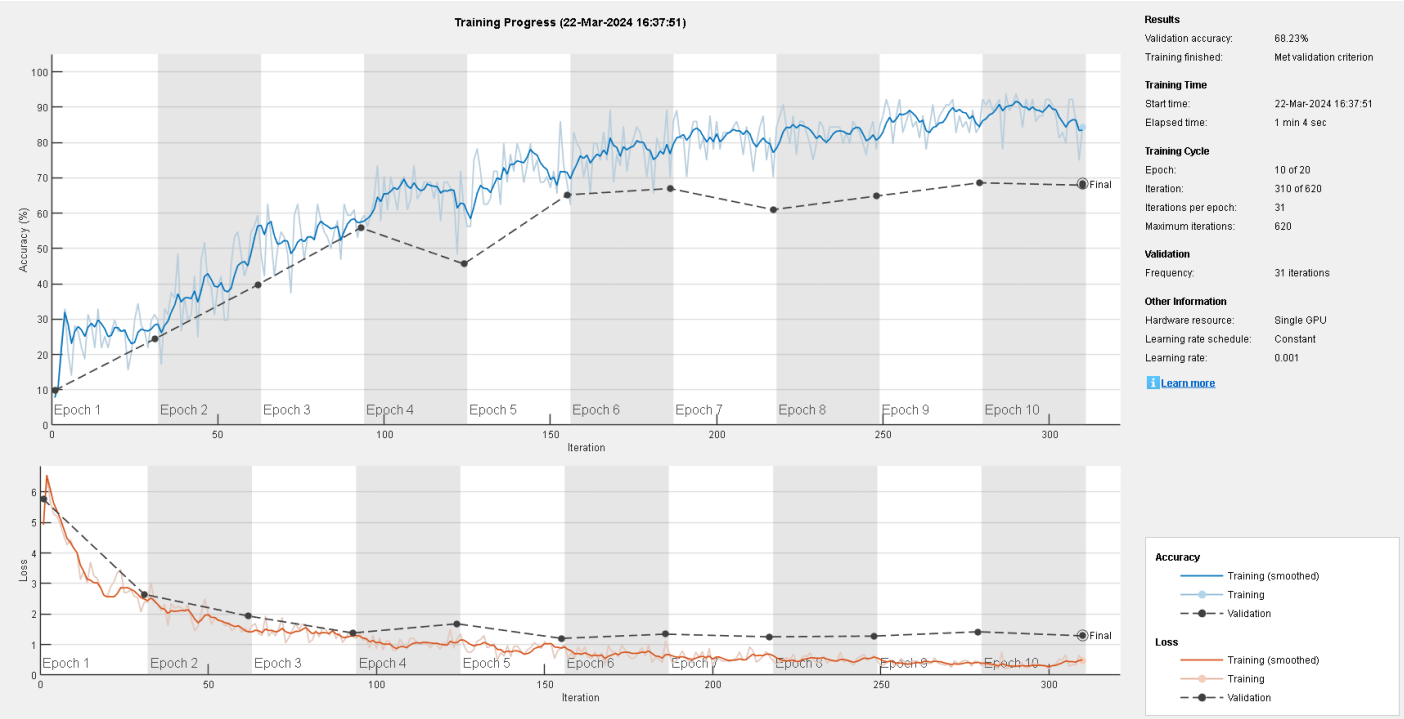
Registration No: 230118234




```
Layers: 3, Initial Filters: 32, Validation Accuracy: 70.03%
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:03 |        7.81% |        9.82% |       4.9239 |       5.7623 |
0.0010 |
|       1 |          31 |       00:00:10 |       31.25% |       24.34% |       2.4027 |       2.6456 |
0.0010 |
|       2 |          50 |       00:00:12 |       39.06% |              |       2.0269 |              |
0.0010 |
|       2 |          62 |       00:00:15 |       59.38% |       39.71% |       1.4195 |       1.9341 |
0.0010 |
|       3 |          93 |       00:00:21 |       57.81% |       55.85% |       1.2593 |       1.3802 |
0.0010 |
|       4 |         100 |       00:00:22 |       70.31% |              |       0.8658 |              |
0.0010 |
|       4 |         124 |       00:00:27 |       60.94% |       45.60% |       1.1797 |       1.6816 |
0.0010 |
|       5 |         150 |       00:00:30 |       67.19% |              |       1.0922 |              |
0.0010 |
|       5 |         155 |       00:00:33 |       68.75% |       65.16% |       0.8972 |       1.2030 |
0.0010 |
|       6 |         186 |       00:00:38 |       70.31% |       66.95% |       1.1085 |       1.3413 |
0.0010 |
|       7 |         200 |       00:00:41 |       85.94% |              |       0.3127 |              |
0.0010 |
```

Registration No: 230118234

```
|       7 |         217 |       00:00:44 |       70.31% |       60.97% |       0.6048 |       1.2533 |
0.0010 |
|       8 |         248 |       00:00:50 |       78.12% |       64.82% |       0.6340 |       1.2781 |
0.0010 |
|       9 |         250 |       00:00:51 |       87.50% |              |       0.3159 |              |
0.0010 |
|       9 |         279 |       00:00:56 |       84.38% |       68.57% |       0.4332 |       1.4195 |
0.0010 |
|      10 |         300 |       00:01:00 |       92.19% |              |       0.2190 |              |
0.0010 |
|      10 |         310 |       00:01:03 |       84.38% |       67.89% |       0.4796 |       1.2863 |
0.0010 |
|=========================================================================================================
=============|
Training finished: Met validation criterion.
```



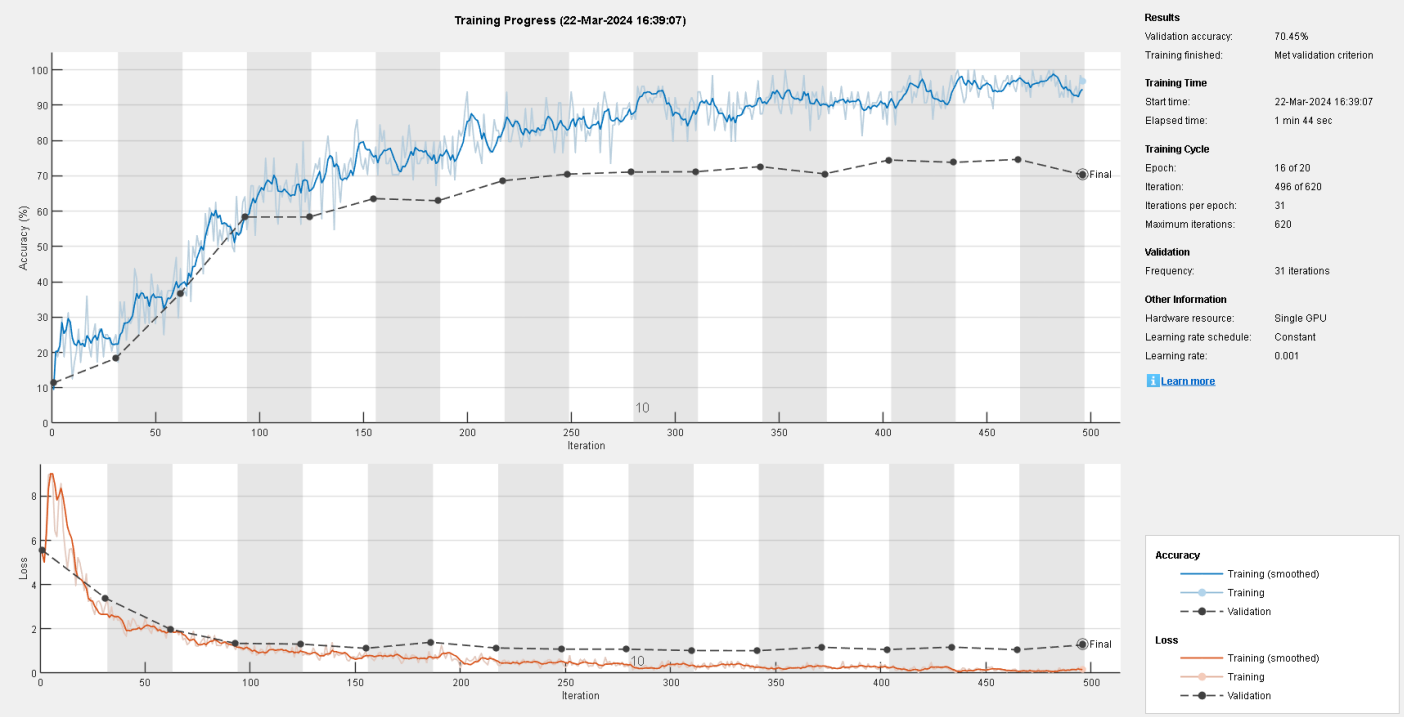

```
Layers: 4, Initial Filters: 16, Validation Accuracy: 68.23%
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        9.38% |       11.36% |       5.4208 |       5.5571 |
0.0010 |
|       1 |          31 |       00:00:10 |       25.00% |       18.36% |       3.0396 |       3.3912 |
0.0010 |
|       2 |          50 |       00:00:12 |       28.12% |              |       2.4502 |              |
0.0010 |
|       2 |          62 |       00:00:16 |       43.75% |       36.72% |       2.0235 |       1.9815 |
0.0010 |
```

Registration No: 230118234

```
|       3 |          93 |       00:00:22 |       64.06% |       58.33% |       1.0624 |       1.3427 |
0.0010 |
|       4 |         100 |       00:00:23 |       67.19% |              |       0.9917 |              |
0.0010 |
|       4 |         124 |       00:00:28 |       59.38% |       58.33% |       1.1249 |       1.3098 |
0.0010 |
|       5 |         150 |       00:00:32 |       71.88% |              |       0.7007 |              |
0.0010 |
|       5 |         155 |       00:00:34 |       73.44% |       63.54% |       0.9823 |       1.1190 |
0.0010 |
|       6 |         186 |       00:00:40 |       76.56% |       62.94% |       0.6930 |       1.3914 |
0.0010 |
|       7 |         200 |       00:00:42 |       93.75% |              |       0.2329 |              |
0.0010 |
|       7 |         217 |       00:00:46 |       82.81% |       68.57% |       0.5225 |       1.1349 |
0.0010 |
|       8 |         248 |       00:00:53 |       81.25% |       70.45% |       0.5610 |       1.0806 |
0.0010 |
|       9 |         250 |       00:00:53 |       93.75% |              |       0.2431 |              |
0.0010 |
|       9 |         279 |       00:00:59 |       85.94% |       71.05% |       0.3385 |       1.0780 |
0.0010 |
|      10 |         300 |       00:01:02 |       92.19% |              |       0.3044 |              |
0.0010 |
|      10 |         310 |       00:01:05 |       82.81% |       71.14% |       0.4873 |       1.0188 |
0.0010 |
|      11 |         341 |       00:01:12 |       87.50% |       72.59% |       0.3834 |       1.0095 |
0.0010 |
|      12 |         350 |       00:01:13 |       89.06% |              |       0.2836 |              |
0.0010 |
|      12 |         372 |       00:01:18 |       90.62% |       70.54% |       0.2975 |       1.1659 |
0.0010 |
|      13 |         400 |       00:01:22 |       85.94% |              |       0.4292 |              |
0.0010 |
|      13 |         403 |       00:01:24 |       93.75% |       74.47% |       0.1929 |       1.0619 |
0.0010 |
|      14 |         434 |       00:01:30 |       95.31% |       73.78% |       0.1148 |       1.1735 |
0.0010 |
|      15 |         450 |       00:01:33 |       96.88% |              |       0.1085 |              |
0.0010 |
|      15 |         465 |       00:01:36 |       96.88% |       74.64% |       0.0923 |       1.0537 |
0.0010 |
|      16 |         496 |       00:01:43 |       96.88% |       70.28% |       0.1593 |       1.2691 |
0.0010 |
|=============================================================================================================
=============|
```

Registration No: 230118234

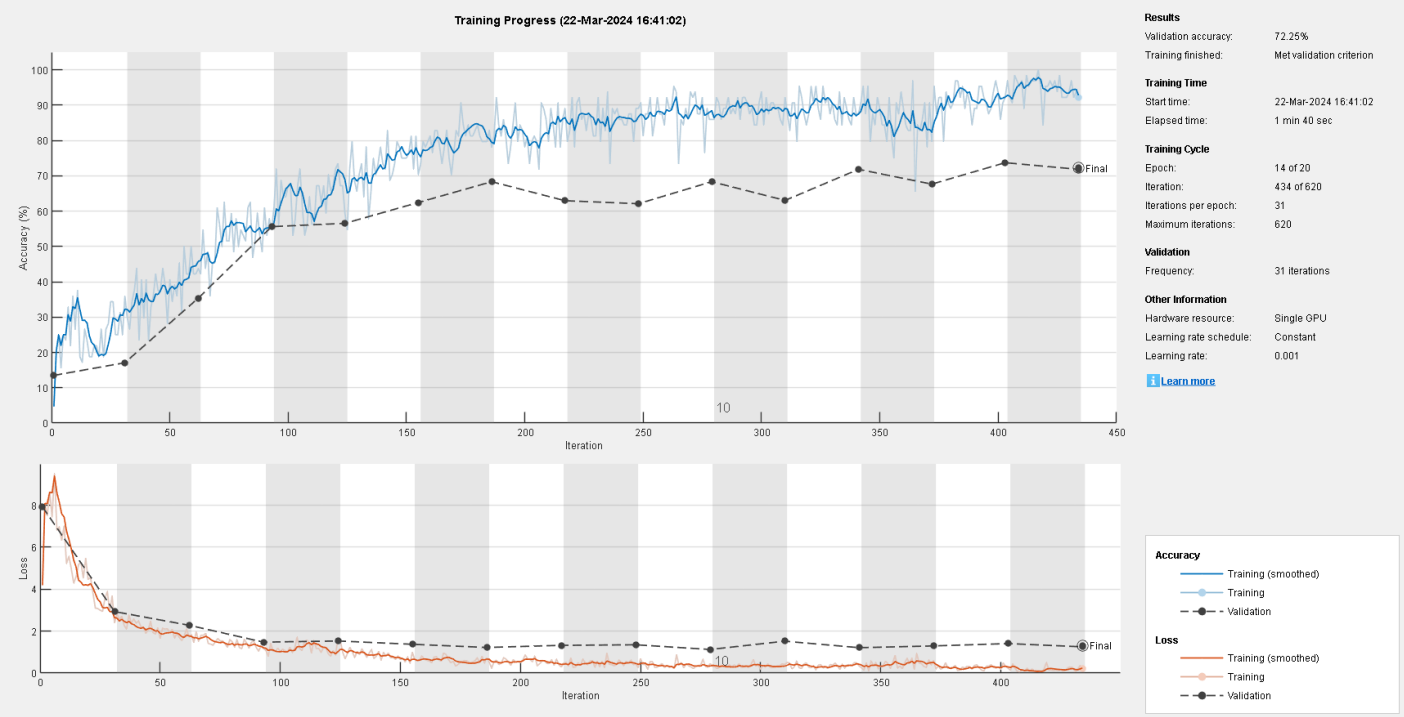
```
Training finished: Met validation criterion.
```




```
Layers: 4, Initial Filters: 24, Validation Accuracy: 70.45%
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        4.69% |       13.49% |       4.1832 |       7.9168 |
0.0010 |
|       1 |          31 |       00:00:11 |       35.94% |       17.08% |       2.3915 |       2.9291 |
0.0010 |
|       2 |          50 |       00:00:14 |       40.62% |              |       1.6640 |              |
0.0010 |
|       2 |          62 |       00:00:17 |       43.75% |       35.35% |       1.7185 |       2.2791 |
0.0010 |
|       3 |          93 |       00:00:24 |       54.69% |       55.59% |       1.1707 |       1.4730 |
0.0010 |
|       4 |         100 |       00:00:25 |       67.19% |              |       1.0132 |              |
0.0010 |
|       4 |         124 |       00:00:30 |       59.38% |       56.53% |       1.6257 |       1.5288 |
0.0010 |
|       5 |         150 |       00:00:34 |       75.00% |              |       0.6368 |              |
0.0010 |
|       5 |         155 |       00:00:37 |       70.31% |       62.43% |       0.9853 |       1.3712 |
0.0010 |
|       6 |         186 |       00:00:44 |       78.12% |       68.32% |       0.7242 |       1.2201 |
0.0010 |
|       7 |         200 |       00:00:46 |       78.12% |              |       0.4997 |              |
0.0010 |
```

Registration No: 230118234

```
|       7 |         217 |       00:00:50 |       89.06% |       62.94% |       0.2864 |       1.3208 |
0.0010 |
|       8 |         248 |       00:00:57 |       89.06% |       62.08% |       0.4307 |       1.3480 |
0.0010 |
|       9 |         250 |       00:00:57 |       81.25% |              |       0.6904 |              |
0.0010 |
|       9 |         279 |       00:01:04 |       85.94% |       68.32% |       0.4542 |       1.1238 |
0.0010 |
|      10 |         300 |       00:01:07 |       93.75% |              |       0.1924 |              |
0.0010 |
|      10 |         310 |       00:01:10 |       92.19% |       63.02% |       0.4010 |       1.5194 |
0.0010 |
|      11 |         341 |       00:01:17 |       87.50% |       71.82% |       0.3402 |       1.2100 |
0.0010 |
|      12 |         350 |       00:01:18 |       89.06% |              |       0.3385 |              |
0.0010 |
|      12 |         372 |       00:01:23 |       79.69% |       67.63% |       0.6006 |       1.2925 |
0.0010 |
|      13 |         400 |       00:01:28 |       93.75% |              |       0.1837 |              |
0.0010 |
|      13 |         403 |       00:01:30 |       92.19% |       73.70% |       0.2381 |       1.4052 |
0.0010 |
|      14 |         434 |       00:01:38 |       92.19% |       71.82% |       0.1954 |       1.2496 |
0.0010 |
|========================================================================================================
=============|
Training finished: Met validation criterion.
```



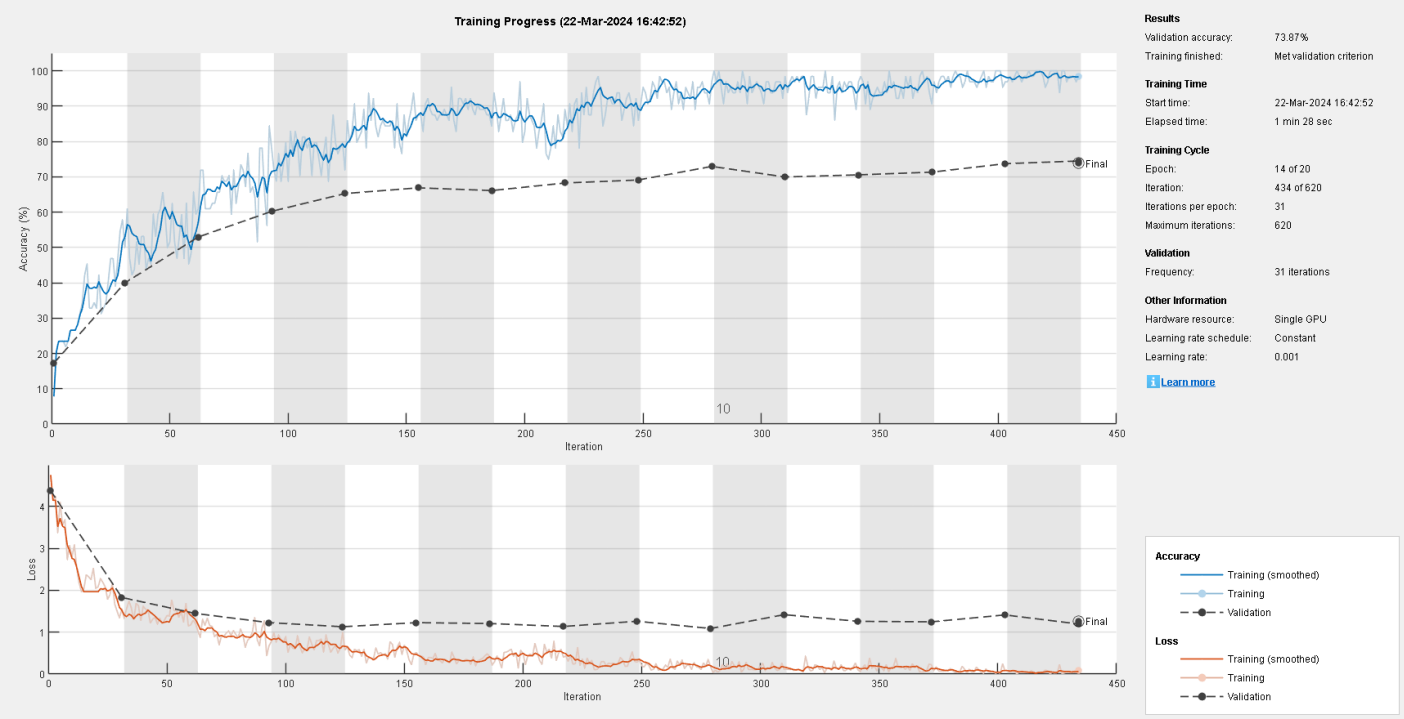

```
Layers: 4, Initial Filters: 32, Validation Accuracy: 72.25%
Training on single GPU.
Initializing input data normalization.
|========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
```

Registration No: 230118234

1	1	00:00:03	7.81%	17.25%	4.7506	4.3753	0.0010
1	31	00:00:09	50.00%	39.97%	1.5561	1.8262	0.0010
2	50	00:00:12	51.56%		1.3941		0.0010
2	62	00:00:16	60.94%	52.95%	1.2087	1.4455	0.0010
3	93	00:00:21	78.12%	60.20%	0.6742	1.2267	0.0010
4	100	00:00:23	68.75%		0.8923		0.0010
4	124	00:00:27	71.88%	65.24%	0.9909	1.1285	0.0010
5	150	00:00:31	81.25%		0.6905		0.0010
5	155	00:00:33	87.50%	66.87%	0.5148	1.2263	0.0010
6	186	00:00:39	84.38%	66.10%	0.4077	1.2050	0.0010
7	200	00:00:41	82.81%		0.4895		0.0010
7	217	00:00:45	89.06%	68.23%	0.4039	1.1350	0.0010
8	248	00:00:51	85.94%	69.09%	0.2820	1.2595	0.0010
9	250	00:00:51	95.31%		0.1996		0.0010
9	279	00:00:57	96.88%	72.93%	0.1780	1.0862	0.0010
10	300	00:01:00	93.75%		0.1889		0.0010
10	310	00:01:03	98.44%	69.94%	0.1115	1.4167	0.0010
11	341	00:01:09	96.88%	70.54%	0.1051	1.2577	0.0010
12	350	00:01:11	93.75%		0.1845		0.0010
12	372	00:01:15	90.62%	71.31%	0.2231	1.2454	0.0010
13	400	00:01:19	100.00%		0.0394		0.0010
13	403	00:01:21	96.88%	73.70%	0.0872	1.4102	0.0010
14	434	00:01:27	98.44%	74.47%	0.0947	1.1941	0.0010

Registration No: 230118234

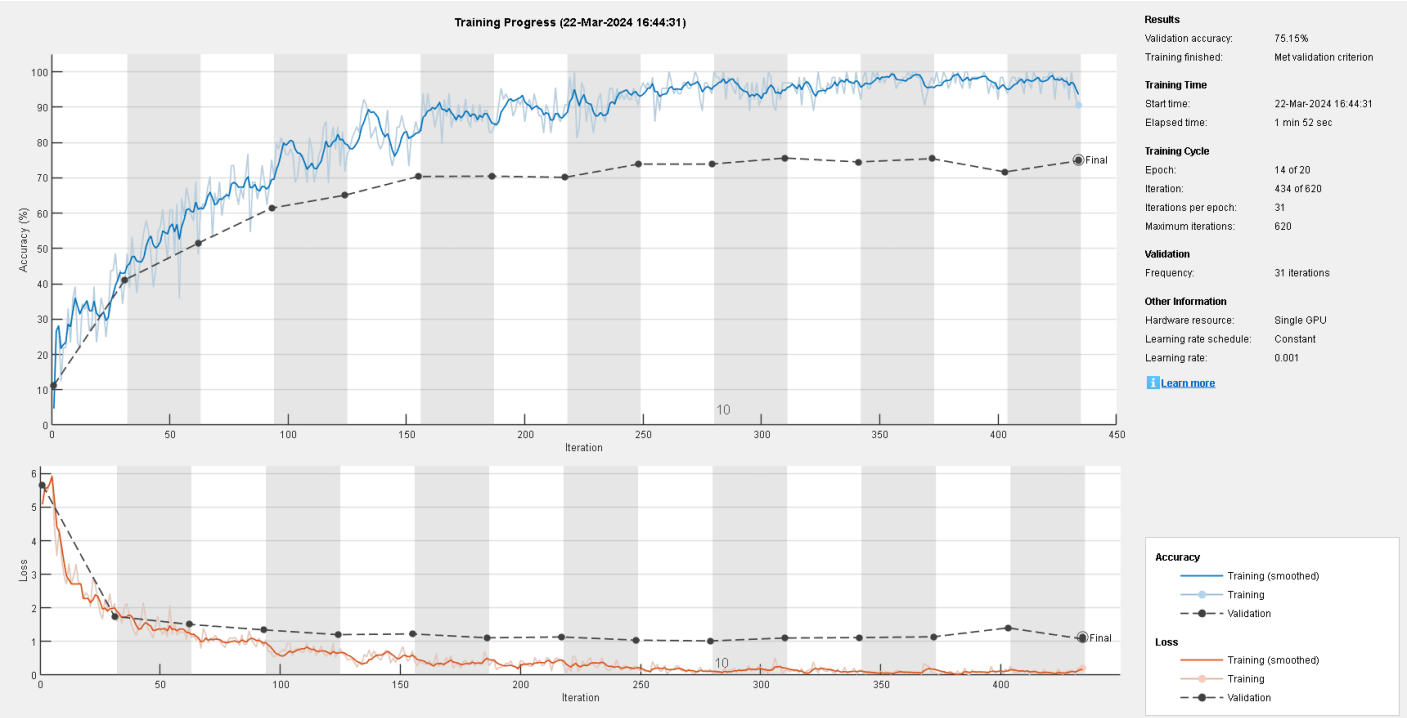
```
Training finished: Met validation criterion.
```




```
Layers: 5, Initial Filters: 16, Validation Accuracy: 73.87%
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        4.69% |       11.27% |       5.0795 |       5.6616 |
0.0010 |
|       1 |          31 |       00:00:10 |       40.62% |       41.08% |       2.1185 |       1.7476 |
0.0010 |
|       2 |          50 |       00:00:14 |       60.94% |              |       1.3643 |              |
0.0010 |
|       2 |          62 |       00:00:17 |       48.44% |       51.49% |       1.4873 |       1.5099 |
0.0010 |
|       3 |          93 |       00:00:24 |       75.00% |       61.40% |       0.8555 |       1.3462 |
0.0010 |
|       4 |         100 |       00:00:26 |       76.56% |              |       0.6795 |              |
0.0010 |
|       4 |         124 |       00:00:32 |       71.88% |       65.07% |       0.6877 |       1.2031 |
0.0010 |
|       5 |         150 |       00:00:37 |       82.81% |              |       0.6685 |              |
0.0010 |
|       5 |         155 |       00:00:40 |       82.81% |       70.37% |       0.7142 |       1.2249 |
0.0010 |
|       6 |         186 |       00:00:49 |       82.81% |       70.45% |       0.5283 |       1.1095 |
0.0010 |
|       7 |         200 |       00:00:52 |       85.94% |              |       0.4336 |              |
0.0010 |
|       7 |         217 |       00:00:57 |       90.62% |       70.11% |       0.4456 |       1.1346 |
0.0010 |
```

Registration No: 230118234

```
|       8 |         248 |       00:01:05 |       90.62% |       73.87% |       0.2630 |       1.0402 |
0.0010 |
|       9 |         250 |       00:01:05 |       96.88% |              |       0.1190 |              |
0.0010 |
|       9 |         279 |       00:01:12 |       93.75% |       73.87% |       0.1678 |       1.0169 |
0.0010 |
|      10 |         300 |       00:01:16 |       92.19% |              |       0.2252 |              |
0.0010 |
|      10 |         310 |       00:01:19 |       96.88% |       75.58% |       0.1260 |       1.1036 |
0.0010 |
|      11 |         341 |       00:01:27 |       95.31% |       74.47% |       0.1374 |       1.1156 |
0.0010 |
|      12 |         350 |       00:01:28 |       96.88% |              |       0.1139 |              |
0.0010 |
|      12 |         372 |       00:01:34 |       98.44% |       75.41% |       0.0580 |       1.1398 |
0.0010 |
|      13 |         400 |       00:01:39 |       92.19% |              |       0.1239 |              |
0.0010 |
|      13 |         403 |       00:01:41 |       95.31% |       71.65% |       0.1042 |       1.4058 |
0.0010 |
|      14 |         434 |       00:01:50 |       90.62% |       74.81% |       0.1983 |       1.0754 |
0.0010 |
|========================================================================================================
============|
Training finished: Met validation criterion.
```



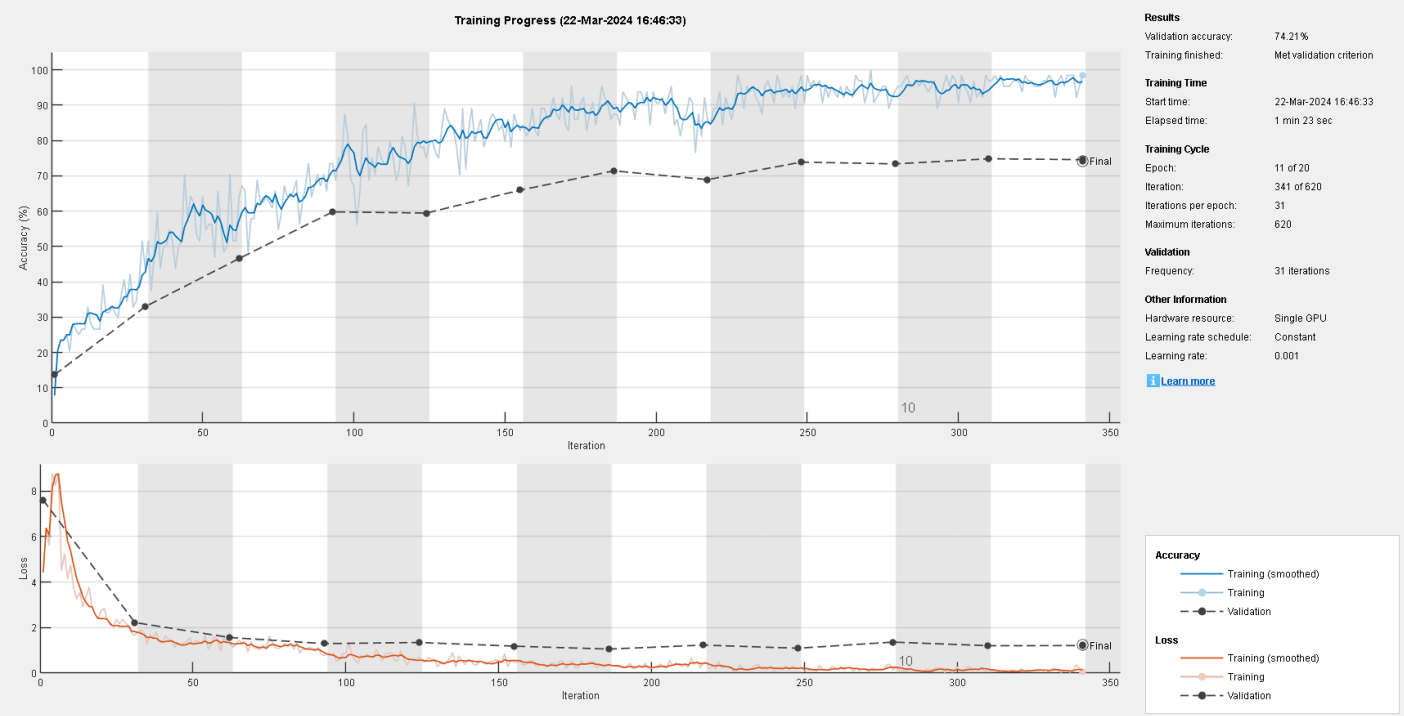

```
Layers: 5, Initial Filters: 24, Validation Accuracy: 75.15%
Training on single GPU.
Initializing input data normalization.
|========================================================================================================
============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
```

Registration No: 230118234

1	1	00:00:04	7.81%	13.66%	4.4139	7.6053	0.0010
1	31	00:00:11	43.75%	32.96%	1.6683	2.2205	0.0010
2	50	00:00:15	64.06%		1.2842		0.0010
2	62	00:00:19	64.06%	46.54%	1.3282	1.5638	0.0010
3	93	00:00:26	73.44%	59.78%	0.7749	1.2947	0.0010
4	100	00:00:27	67.19%		0.8706		0.0010
4	124	00:00:33	75.00%	59.44%	0.7123	1.3435	0.0010
5	150	00:00:37	79.69%		0.5536		0.0010
5	155	00:00:40	82.81%	65.93%	0.5121	1.1825	0.0010
6	186	00:00:47	87.50%	71.39%	0.4038	1.0548	0.0010
7	200	00:00:49	90.62%		0.2192		0.0010
7	217	00:00:53	85.94%	68.92%	0.4452	1.2277	0.0010
8	248	00:01:00	98.44%	73.87%	0.1883	1.0915	0.0010
9	250	00:01:01	98.44%		0.0887		0.0010
9	279	00:01:07	93.75%	73.36%	0.1458	1.3617	0.0010
10	300	00:01:11	90.62%		0.2656		0.0010
10	310	00:01:14	96.88%	74.81%	0.0902	1.2029	0.0010
11	341	00:01:21	98.44%	74.55%	0.0415	1.2061	0.0010

Registration No: 230118234

```
Training finished: Met validation criterion.
```



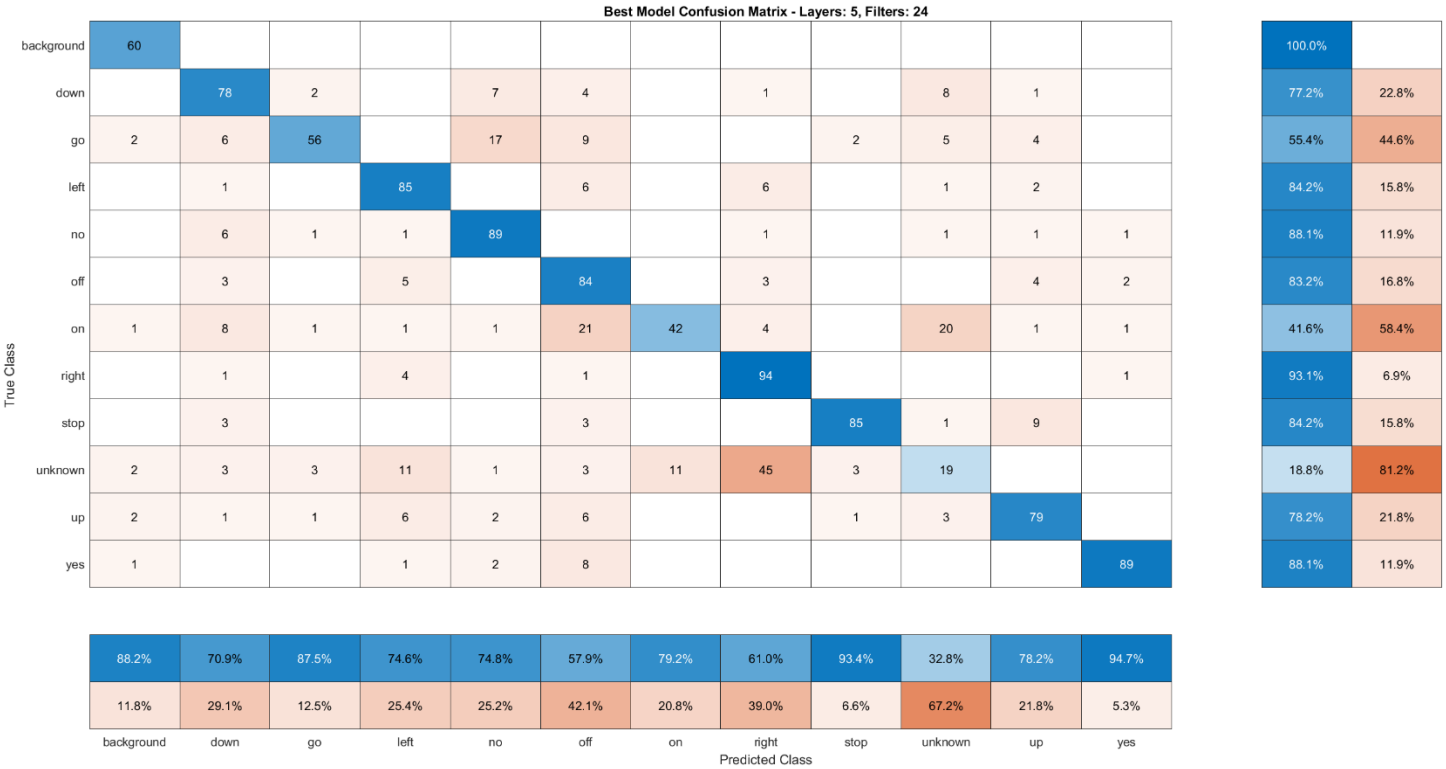

```
Layers: 5, Initial Filters: 32, Validation Accuracy: 74.21%
```

```
% Display and Plot Best Model
fprintf('\nBest Model: %d Layers, %d Initial Filters, Validation Accuracy: %.2f%%\n',
bestNumLayers, bestNumFilters, bestAccuracy * 100);
```

```
Best Model: 5 Layers, 24 Initial Filters, Validation Accuracy: 75.15%
```

```
% Optional: Plot Confusion Matrix for Best Model
YValidationBest = classify(bestModel, adsValidation);
TValidationBest = adsValidation.Labels;
figure;
cm = confusionchart(TValidationBest, YValidationBest, ...
    'Title', ['Best Model Confusion Matrix - Layers: ', num2str(bestNumLayers), ', Filters:
', num2str(bestNumFilters)], ...
    'ColumnSummary', 'column-normalized', 'RowSummary', 'row-normalized');
sortClasses(cm, categories(adsTrain.Labels));
```

Registration No: 230118234



Registration No: 230118234

Task 3: Model Averaging Scheme

Objective: Implement a model averaging scheme with three models to improve the ASR system's generalization on unseen data.

Description: In this task, the focus shifts to improving model generalization through an ensemble method. Three distinct models are trained on randomly split subsets of the training data, encouraging each to learn from a slightly different perspective of the data. These models are designed with a consistent architecture suitable for speech recognition tasks.

Bootstrapping is employed to create several subsets of the original training data and each subset is then used to train a distinctive model. The ensemble's prediction for each input is determined by majority voting - each model casts a "vote" for its predicted class, and the class receiving the majority of votes is selected as the final prediction. This voting mechanism capitalizes on the collective intelligence of multiple models, potentially leading to more accurate and robust predictions.

To assess the effectiveness of this approach, the ensemble's accuracy on the validation dataset is calculated, which is around 75% (similar to the baseline model however with one less layer and more efficient), and compared to the performance of a single model is slightly higher. This comparison aims to highlight the ensemble method's capability to outperform individual models by leveraging their combined strengths, ultimately leading to a more reliable model.

```
fprintf('Starting Model Averaging Approach...\n');
```

```
Starting Model Averaging Approach...
```

```
% Define Network Architecture
numClasses = numel(categories(adsTrain.Labels));
layers = [
    imageInputLayer(inputSize)
    convolution2dLayer(3, 16, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')
    convolution2dLayer(3, 32, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')
    convolution2dLayer(3, 64, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')
    convolution2dLayer(3, 64, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer([ceil(inputSize(1)/8), 1])
    dropoutLayer(0.2)
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];

% Specify Training Options
```

Registration No: 230118234

```
options = trainingOptions('adam', ...
    'InitialLearnRate', 0.001, ...
    'MaxEpochs', 20, ...
    'MiniBatchSize', 64, ...
    'Shuffle', 'every-epoch', ...
    'Plots', 'training-progress', ...
    'Verbose', true, ...
    'ValidationData', adsValidation, ...
    'ValidationFrequency', floor(numel(adsTrain.Files)/128));

% Train and Evaluate Models with Model Averaging
numModels = 3;
models = cell(1, numModels);

% Extract file paths and labels for bootstrapping
allFiles = adsTrain.Files;
allLabels = adsTrain.Labels;

for i = 1:numModels
    % Bootstrap sampling: Generate indices with replacement
    idx = randi([1 numel(allFiles)], numel(allFiles), 1);

    % Create bootstrapped file paths and labels
    bootFiles = allFiles(idx);
    bootLabels = allLabels(idx);

    % Create a new imageDatastore for the bootstrapped dataset
    adsTrainSubset = imageDatastore(bootFiles, 'Labels', bootLabels);
    adsTrainSubset.ReadFcn = adsTrain.ReadFcn;

    % Train model on the subset
    models{i} = trainNetwork(adsTrainSubset, layers, options);
end
```

```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        4.69% |       11.19% |       4.0386 |       3.0361 |
0.0010 |
|       1 |          15 |       00:00:09 |       37.50% |       23.83% |       2.0875 |       2.2581 |
0.0010 |
|       1 |          30 |       00:00:13 |       35.94% |       33.56% |       2.1982 |       1.9969 |
0.0010 |
|       2 |          45 |       00:00:16 |       51.56% |       42.87% |       1.3222 |       1.7865 |
0.0010 |
```

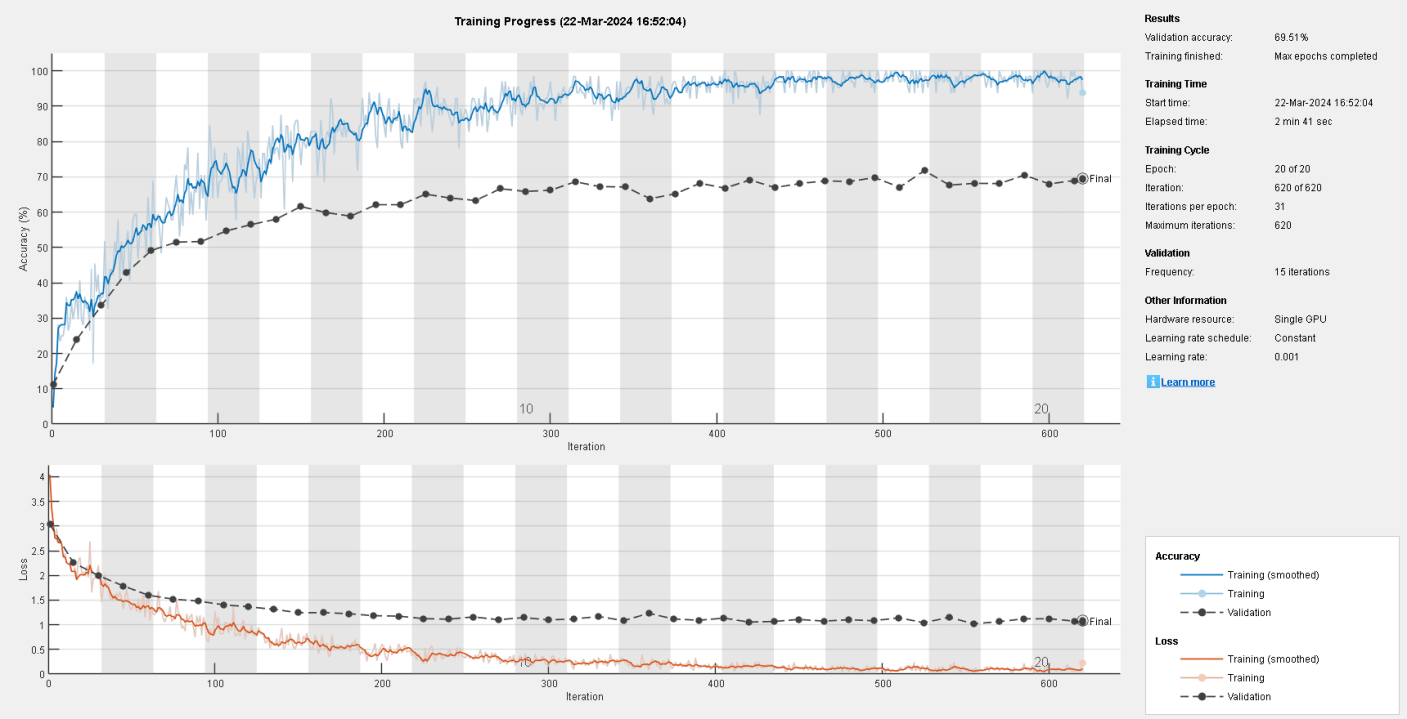
2	50	00:00:17	59.38%		1.2980		0.0010
2	60	00:00:20	50.00%	49.10%	1.5148	1.5979	0.0010
3	75	00:00:24	70.31%	51.49%	1.1207	1.5183	0.0010
3	90	00:00:28	70.31%	51.67%	1.0341	1.4787	0.0010
4	100	00:00:29	64.06%		1.2584		0.0010
4	105	00:00:31	76.56%	54.65%	0.8252	1.4047	0.0010
4	120	00:00:35	76.56%	56.45%	0.8703	1.3631	0.0010
5	135	00:00:39	78.12%	57.98%	0.6095	1.3128	0.0010
5	150	00:00:42	73.44%	61.57%	0.8124	1.2429	0.0010
6	165	00:00:46	81.25%	59.86%	0.6022	1.2453	0.0010
6	180	00:00:50	82.81%	58.84%	0.5950	1.2241	0.0010
7	195	00:00:54	81.25%	62.08%	0.4962	1.1825	0.0010
7	200	00:00:54	84.38%		0.6009		0.0010
7	210	00:00:57	79.69%	62.08%	0.6079	1.1744	0.0010
8	225	00:01:01	96.88%	65.07%	0.2198	1.1233	0.0010
8	240	00:01:05	85.94%	63.96%	0.4691	1.1121	0.0010
9	250	00:01:06	87.50%		0.3230		0.0010
9	255	00:01:09	90.62%	63.28%	0.2920	1.1586	0.0010
9	270	00:01:13	92.19%	66.61%	0.3499	1.1014	0.0010
10	285	00:01:16	89.06%	65.84%	0.2492	1.1450	0.0010
10	300	00:01:20	96.88%	66.18%	0.1639	1.0975	0.0010
11	315	00:01:24	98.44%	68.57%	0.1011	1.1160	0.0010
11	330	00:01:27	90.62%	67.21%	0.3294	1.1669	0.0010
12	345	00:01:31	93.75%	67.12%	0.2948	1.0824	0.0010
12	350	00:01:32	96.88%		0.1401		0.0010
12	360	00:01:35	98.44%	63.79%	0.0899	1.2311	0.0010
13	375	00:01:38	96.88%	65.16%	0.1706	1.1167	0.0010
13	390	00:01:42	95.31%	68.15%	0.1242	1.0902	0.0010
13	400	00:01:43	95.31%		0.1543		0.0010

Registration No: 230118234

```
|      14 |         405 |       00:01:46 |      100.00% |       66.78% |       0.1197 |       1.1342 |
0.0010 |
|      14 |         420 |       00:01:49 |       96.88% |       69.09% |       0.1486 |       1.0542 |
0.0010 |
|      15 |         435 |       00:01:53 |      100.00% |       66.95% |       0.0713 |       1.0657 |
0.0010 |
|      15 |         450 |       00:01:57 |       98.44% |       68.06% |       0.1141 |       1.1072 |
0.0010 |
|      15 |         465 |       00:02:00 |       98.44% |       68.83% |       0.0964 |       1.0731 |
0.0010 |
|      16 |         480 |       00:02:04 |      100.00% |       68.66% |       0.0512 |       1.1012 |
0.0010 |
|      16 |         495 |       00:02:07 |       93.75% |       69.77% |       0.1608 |       1.0803 |
0.0010 |
|      17 |         500 |       00:02:08 |       96.88% |              |       0.0886 |              |
0.0010 |
|      17 |         510 |       00:02:11 |       96.88% |       67.04% |       0.1065 |       1.1416 |
0.0010 |
|      17 |         525 |       00:02:15 |       98.44% |       71.73% |       0.1095 |       1.0395 |
0.0010 |
|      18 |         540 |       00:02:19 |       98.44% |       67.63% |       0.0820 |       1.1504 |
0.0010 |
|      18 |         550 |       00:02:20 |       98.44% |              |       0.0768 |              |
0.0010 |
|      18 |         555 |       00:02:22 |       98.44% |       68.15% |       0.0676 |       1.0186 |
0.0010 |
|      19 |         570 |       00:02:26 |      100.00% |       68.06% |       0.0673 |       1.0628 |
0.0010 |
|      19 |         585 |       00:02:29 |       95.31% |       70.45% |       0.2112 |       1.1201 |
0.0010 |
|      20 |         600 |       00:02:33 |       96.88% |       67.98% |       0.0925 |       1.1212 |
0.0010 |
|      20 |         615 |       00:02:37 |      100.00% |       68.92% |       0.0551 |       1.0735 |
0.0010 |
|      20 |         620 |       00:02:39 |       93.75% |       69.26% |       0.2273 |       1.0479 |
0.0010 |
|=============================================================================================================
=============|
```

Registration No: 230118234

```
Training finished: Max epochs completed.
```

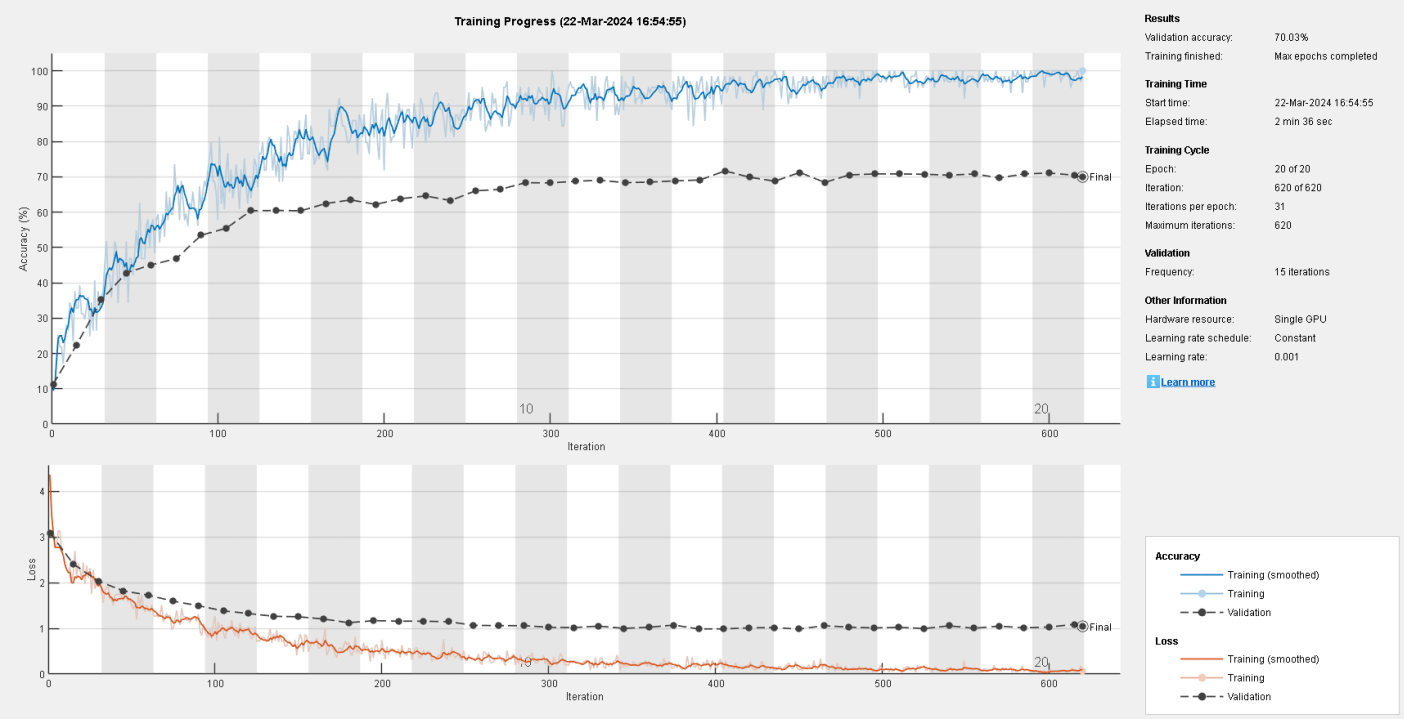



```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        9.38% |       11.27% |       4.3562 |       3.0890 |
0.0010 |
|       1 |          15 |       00:00:07 |       32.81% |       22.37% |       2.2191 |       2.4072 |
0.0010 |
|       1 |          30 |       00:00:11 |       35.94% |       35.18% |       1.9400 |       2.0333 |
0.0010 |
|       2 |          45 |       00:00:14 |       51.56% |       42.78% |       1.6116 |       1.8179 |
0.0010 |
|       2 |          50 |       00:00:15 |       48.44% |              |       1.4897 |              |
0.0010 |
|       2 |          60 |       00:00:18 |       56.25% |       45.09% |       1.3878 |       1.7245 |
0.0010 |
|       3 |          75 |       00:00:22 |       65.62% |       46.80% |       1.0031 |       1.5923 |
0.0010 |
|       3 |          90 |       00:00:25 |       60.94% |       53.46% |       1.3010 |       1.4944 |
0.0010 |
|       4 |         100 |       00:00:27 |       57.81% |              |       1.2596 |              |
0.0010 |
|       4 |         105 |       00:00:29 |       71.88% |       55.42% |       0.9026 |       1.3820 |
0.0010 |
|       4 |         120 |       00:00:32 |       64.06% |       60.38% |       1.0370 |       1.3243 |
0.0010 |
|       5 |         135 |       00:00:36 |       70.31% |       60.46% |       0.8765 |       1.2627 |
0.0010 |
```

5	150	00:00:40	75.00%	60.38%	0.7310	1.2528	0.0010
6	165	00:00:44	73.44%	62.43%	0.7656	1.2034	0.0010
6	180	00:00:47	81.25%	63.45%	0.6926	1.1234	0.0010
7	195	00:00:51	90.62%	62.17%	0.3554	1.1666	0.0010
7	200	00:00:52	71.88%		0.7329		0.0010
7	210	00:00:54	87.50%	63.79%	0.5474	1.1573	0.0010
8	225	00:00:58	79.69%	64.56%	0.5140	1.1555	0.0010
8	240	00:01:02	82.81%	63.28%	0.4767	1.1447	0.0010
9	250	00:01:03	89.06%		0.3069		0.0010
9	255	00:01:05	92.19%	66.01%	0.3225	1.0664	0.0010
9	270	00:01:09	85.94%	66.44%	0.4072	1.0578	0.0010
10	285	00:01:13	93.75%	68.32%	0.2223	1.0597	0.0010
10	300	00:01:16	92.19%	68.23%	0.1936	1.0228	0.0010
11	315	00:01:20	93.75%	68.74%	0.2868	1.0163	0.0010
11	330	00:01:24	92.19%	69.00%	0.2289	1.0480	0.0010
12	345	00:01:27	96.88%	68.32%	0.1563	0.9992	0.0010
12	350	00:01:28	98.44%		0.2125		0.0010
12	360	00:01:31	93.75%	68.49%	0.2717	1.0241	0.0010
13	375	00:01:34	93.75%	68.83%	0.1417	1.0708	0.0010
13	390	00:01:39	96.88%	69.09%	0.1209	0.9862	0.0010
13	400	00:01:40	96.88%		0.1480		0.0010
14	405	00:01:42	98.44%	71.65%	0.1178	0.9869	0.0010
14	420	00:01:46	93.75%	69.85%	0.1952	1.0099	0.0010
15	435	00:01:50	100.00%	68.74%	0.0952	1.0172	0.0010
15	450	00:01:53	96.88%	71.22%	0.0952	0.9825	0.0010
15	465	00:01:57	95.31%	68.40%	0.2166	1.0580	0.0010
16	480	00:02:00	98.44%	70.54%	0.0902	1.0255	0.0010
16	495	00:02:04	98.44%	70.79%	0.0748	1.0097	0.0010
17	500	00:02:05	98.44%		0.0950		0.0010

Registration No: 230118234

```
|      17 |         510 |       00:02:07 |       98.44% |       70.79% |       0.0744 |       1.0214 |
0.0010 |
|      17 |         525 |       00:02:11 |       96.88% |       70.71% |       0.1877 |       0.9950 |
0.0010 |
|      18 |         540 |       00:02:15 |       96.88% |       70.45% |       0.1160 |       1.0540 |
0.0010 |
|      18 |         550 |       00:02:16 |       96.88% |              |       0.1040 |              |
0.0010 |
|      18 |         555 |       00:02:18 |       96.88% |       70.79% |       0.1138 |       1.0016 |
0.0010 |
|      19 |         570 |       00:02:22 |       96.88% |       69.68% |       0.0726 |       1.0477 |
0.0010 |
|      19 |         585 |       00:02:25 |      100.00% |       70.79% |       0.0596 |       1.0111 |
0.0010 |
|      20 |         600 |       00:02:29 |       98.44% |       71.05% |       0.0786 |       1.0275 |
0.0010 |
|      20 |         615 |       00:02:33 |       98.44% |       70.45% |       0.0542 |       1.0875 |
0.0010 |
|      20 |         620 |       00:02:35 |      100.00% |       69.85% |       0.0663 |       1.0384 |
0.0010 |
|========================================================================================================
=============|
Training finished: Max epochs completed.
```




```
Training on single GPU.
Initializing input data normalization.
|========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|========================================================================================================
=============|
|       1 |           1 |       00:00:03 |        7.81% |       10.93% |       4.6470 |       3.7600 |
0.0010 |
```

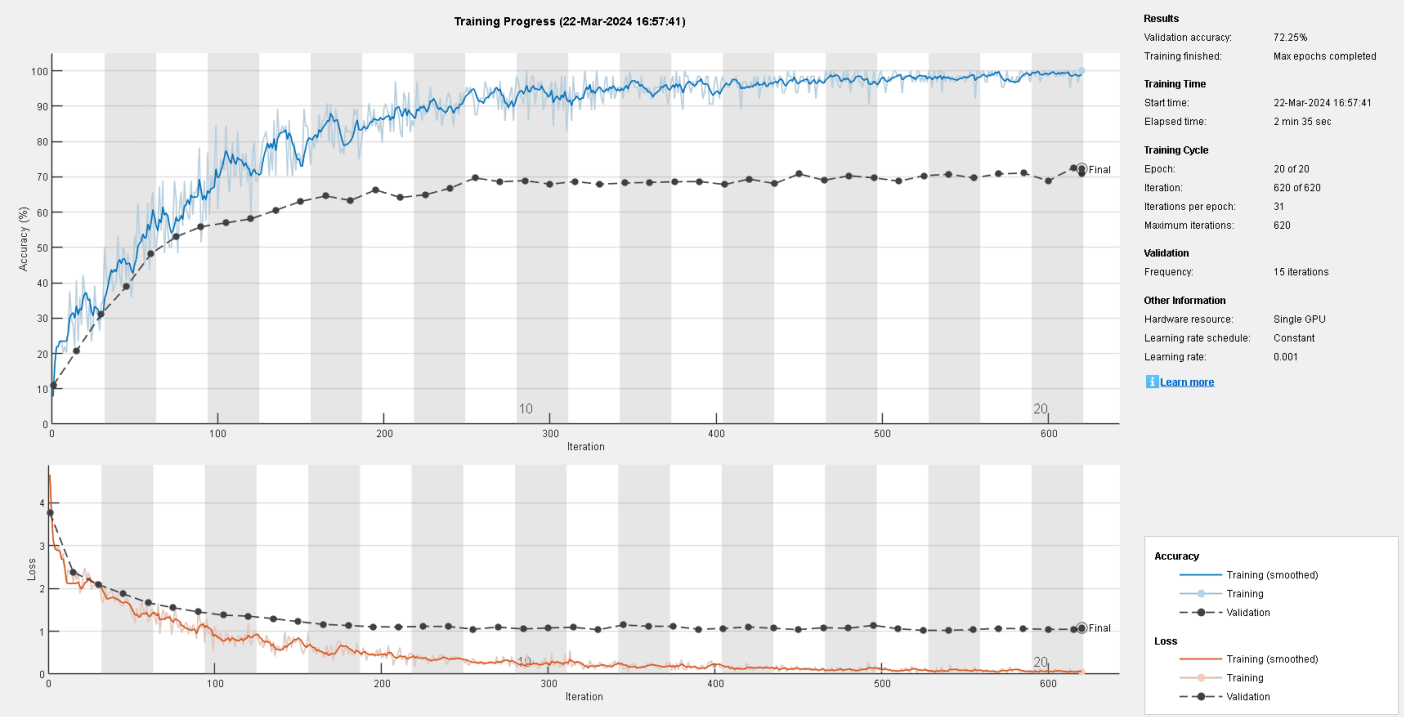
1	15	00:00:06	40.62%	20.67%	2.1312	2.3770	0.0010
1	30	00:00:10	34.38%	31.17%	2.0963	2.0884	0.0010
2	45	00:00:14	40.62%	38.86%	1.5940	1.8705	0.0010
2	50	00:00:14	53.12%		1.4068		0.0010
2	60	00:00:17	62.50%	48.25%	1.2122	1.6613	0.0010
3	75	00:00:21	64.06%	53.03%	1.1905	1.5468	0.0010
3	90	00:00:24	51.56%	55.85%	1.4992	1.4498	0.0010
4	100	00:00:26	60.94%		1.1079		0.0010
4	105	00:00:28	84.38%	56.96%	0.6894	1.3733	0.0010
4	120	00:00:31	59.38%	58.07%	1.1910	1.3510	0.0010
5	135	00:00:35	70.31%	60.55%	0.7378	1.2919	0.0010
5	150	00:00:39	75.00%	63.02%	0.8125	1.2235	0.0010
6	165	00:00:42	87.50%	64.56%	0.4166	1.1590	0.0010
6	180	00:00:46	82.81%	63.28%	0.4712	1.1335	0.0010
7	195	00:00:50	84.38%	66.27%	0.4853	1.1022	0.0010
7	200	00:00:51	89.06%		0.3816		0.0010
7	210	00:00:53	82.81%	64.13%	0.5020	1.0974	0.0010
8	225	00:00:57	87.50%	64.90%	0.3491	1.1127	0.0010
8	240	00:01:01	92.19%	66.78%	0.2786	1.1090	0.0010
9	250	00:01:02	95.31%		0.2537		0.0010
9	255	00:01:05	87.50%	69.68%	0.3608	1.0461	0.0010
9	270	00:01:08	96.88%	68.57%	0.2354	1.0953	0.0010
10	285	00:01:12	93.75%	68.92%	0.2157	1.0509	0.0010
10	300	00:01:15	95.31%	67.89%	0.1834	1.0762	0.0010
11	315	00:01:19	92.19%	68.66%	0.2500	1.0908	0.0010
11	330	00:01:23	95.31%	67.89%	0.1940	1.0342	0.0010
12	345	00:01:26	96.88%	68.23%	0.1164	1.1446	0.0010
12	350	00:01:27	96.88%		0.2113		0.0010
12	360	00:01:30	95.31%	68.40%	0.2420	1.1156	0.0010

Registration No: 230118234

```
|      13 |         375 |       00:01:34 |       92.19% |       68.57% |       0.2237 |       1.1119 |
0.0010 |
|      13 |         390 |       00:01:37 |       98.44% |       68.66% |       0.1037 |       1.0472 |
0.0010 |
|      13 |         400 |       00:01:39 |       98.44% |              |       0.1307 |              |
0.0010 |
|      14 |         405 |       00:01:41 |       95.31% |       67.81% |       0.1588 |       1.0628 |
0.0010 |
|      14 |         420 |       00:01:44 |       95.31% |       69.34% |       0.1888 |       1.0987 |
0.0010 |
|      15 |         435 |       00:01:48 |       98.44% |       68.06% |       0.0983 |       1.0673 |
0.0010 |
|      15 |         450 |       00:01:52 |       96.88% |       70.79% |       0.1173 |       1.0372 |
0.0010 |
|      15 |         465 |       00:01:55 |       96.88% |       69.00% |       0.1356 |       1.0807 |
0.0010 |
|      16 |         480 |       00:01:59 |       96.88% |       70.28% |       0.1186 |       1.0674 |
0.0010 |
|      16 |         495 |       00:02:02 |      100.00% |       69.68% |       0.0543 |       1.1368 |
0.0010 |
|      17 |         500 |       00:02:03 |       98.44% |              |       0.0642 |              |
0.0010 |
|      17 |         510 |       00:02:06 |       98.44% |       68.74% |       0.0821 |       1.0548 |
0.0010 |
|      17 |         525 |       00:02:10 |       96.88% |       70.28% |       0.1023 |       1.0189 |
0.0010 |
|      18 |         540 |       00:02:13 |       98.44% |       70.71% |       0.0594 |       1.0198 |
0.0010 |
|      18 |         550 |       00:02:15 |       96.88% |              |       0.1351 |              |
0.0010 |
|      18 |         555 |       00:02:17 |       98.44% |       69.68% |       0.1063 |       1.0385 |
0.0010 |
|      19 |         570 |       00:02:21 |      100.00% |       70.79% |       0.0398 |       1.0660 |
0.0010 |
|      19 |         585 |       00:02:24 |       98.44% |       71.05% |       0.0835 |       1.0577 |
0.0010 |
|      20 |         600 |       00:02:28 |       98.44% |       68.83% |       0.0719 |       1.0421 |
0.0010 |
|      20 |         615 |       00:02:31 |      100.00% |       72.59% |       0.0433 |       1.0447 |
0.0010 |
|      20 |         620 |       00:02:34 |      100.00% |       70.96% |       0.0385 |       1.0544 |
0.0010 |
|=============================================================================================================
=============|
```

Registration No: 230118234

```
Training finished: Max epochs completed.
```



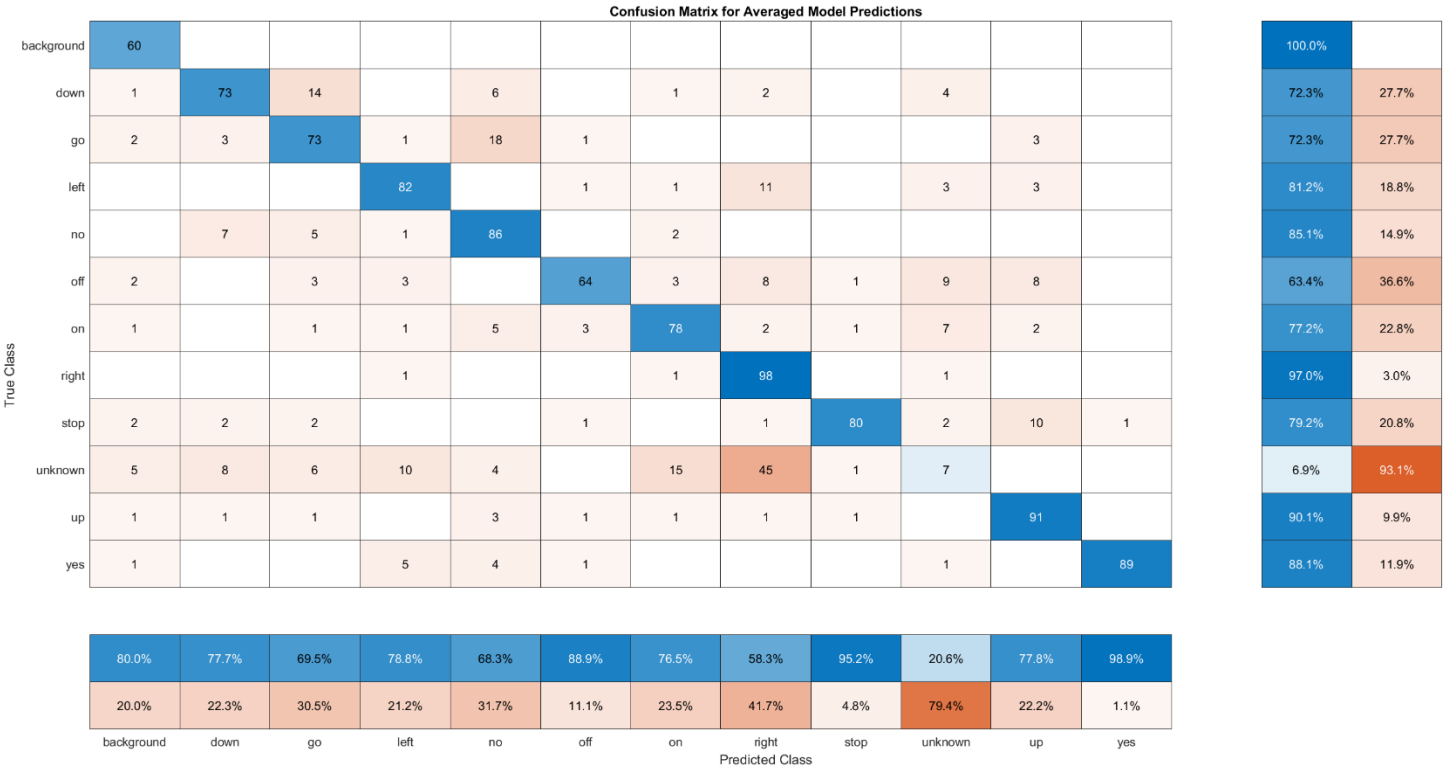

```
% Model Averaging for Predictions
predictions = zeros(numel(adsValidation.Files), numClasses, numModels);

for i = 1:numModels
    % Get predictions from each model
    [YPred, scores] = classify(models{i}, adsValidation);
    predictions(:, :, i) = scores; % Storing the raw scores for averaging
end

% Average the predictions from all models
meanPredictions = mean(predictions, 3);

% Convert averaged scores to categorical predictions
[~, maxScoreIndices] = max(meanPredictions, [], 2);
averagedYPred = categorical(maxScoreIndices, 1:numClasses,
categories(adsValidation.Labels));

% Evaluate the averaged predictions
TValidation = adsValidation.Labels;
accuracy = sum(averagedYPred == TValidation) / numel(TValidation);
fprintf('Ensemble Model Accuracy: %.2f%%\n', accuracy * 100);
```

```
Ensemble Model Accuracy: 74.89%
```

```
% Plot confusion matrix for averaged predictions
figure;
cm = confusionchart(TValidation, averagedYPred, ...
    'Title', 'Confusion Matrix for Averaged Model Predictions', ...
    'ColumnSummary', 'column-normalized', 'RowSummary', 'row-normalized');
```

Registration No: 230118234

```
sortClasses(cm, categories(adsTrain.Labels));
```



Registration No: 230118234

Task 4: Bayesian Optimisation

Objective: Refine by optimising the number of convolutional layers and filters throguh Bayesian Optimisation.

Description: Bayesian Optimization is employed to efficiently fine-tune the hyperparameters concerning the number of convolutional layers (between 4 and 5) and filters per layer (between 16 and 32). This approach uses a probabilistic model to predict the performance of different configurations and selects the next set of parameters to test based on maximizing the expected improvement in validation accuracy.

The optimization runs for five iterations, evaluating the accuracy of the ASR model under different architectural settings. The process aims to find the optimal configuration that leads to the highest accuracy on the validation dataset with minimal evaluations.

The results provide the best-found number of layers and filters, which is 5 layers and 32 filters per layer, the same as found by the grid search.

objectiveFunction is at the end of this file (Objective Function).

```
fprintf('Starting Bayesian Optimization Approach...\n');
```

```
Starting Bayesian Optimization Approach...
```

```
% Set Up and Run Bayesian Optimization
objectiveFunc = @(x) objectiveFunction(x.numLayers, x.numFilters, adsTrain, adsValidation,
inputSize);

numLayers = optimizableVariable('numLayers',[4, 5],'Type','integer');
numFilters = optimizableVariable('numFilters',[16, 32],'Type','integer');

results = bayesopt(objectiveFunc, [numLayers, numFilters], ...
    'IsObjectiveDeterministic', false, ...
    'MaxObjectiveEvaluations', 5, ...
    'AcquisitionFunctionName', 'expected-improvement-plus');
```

```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:03 |       20.31% |       11.96% |       3.0308 |       4.3961 |
0.0010 |
|       1 |          30 |       00:00:09 |       32.81% |       27.24% |       2.1946 |       2.2314 |
0.0010 |
|       2 |          50 |       00:00:11 |       54.69% |              |       1.3123 |              |
0.0010 |
|       2 |          60 |       00:00:14 |       62.50% |       38.51% |       1.1026 |       1.8997 |
0.0010 |
|       3 |          90 |       00:00:20 |       73.44% |       44.06% |       0.9007 |       1.7323 |
0.0010 |
```

```
|       4 |         100 |       00:00:21 |       68.75% |              |       0.9037 |              |
0.0010 |
|       4 |         120 |       00:00:25 |       82.81% |       48.33% |       0.5500 |       1.7045 |
0.0010 |
|       5 |         150 |       00:00:32 |       90.62% |       50.13% |       0.4916 |       1.5881 |
0.0010 |
|       6 |         180 |       00:00:36 |       93.75% |       53.89% |       0.3155 |       1.5215 |
0.0010 |
|       7 |         200 |       00:00:39 |       96.88% |              |       0.2021 |              |
0.0010 |
|       7 |         210 |       00:00:41 |      100.00% |       54.91% |       0.1096 |       1.5242 |
0.0010 |
|       8 |         240 |       00:00:46 |      100.00% |       58.33% |       0.1211 |       1.4654 |
0.0010 |
|       9 |         250 |       00:00:47 |       98.44% |              |       0.0879 |              |
0.0010 |
|       9 |         270 |       00:00:52 |       98.44% |       55.25% |       0.1601 |       1.5203 |
0.0010 |
|      10 |         300 |       00:00:57 |      100.00% |       57.64% |       0.0460 |       1.4666 |
0.0010 |
|      11 |         330 |       00:01:02 |      100.00% |       56.62% |       0.0502 |       1.5077 |
0.0010 |
|      12 |         350 |       00:01:04 |      100.00% |              |       0.0195 |              |
0.0010 |
|      12 |         360 |       00:01:07 |      100.00% |       57.81% |       0.0228 |       1.4825 |
0.0010 |
|      13 |         390 |       00:01:11 |      100.00% |       59.27% |       0.0164 |       1.4973 |
0.0010 |
|      13 |         400 |       00:01:13 |      100.00% |              |       0.0185 |              |
0.0010 |
|      14 |         420 |       00:01:16 |      100.00% |       58.58% |       0.0092 |       1.4920 |
0.0010 |
|      15 |         450 |       00:01:21 |      100.00% |       60.29% |       0.0099 |       1.4544 |
0.0010 |
|      16 |         480 |       00:01:26 |      100.00% |       59.95% |       0.0071 |       1.4910 |
0.0010 |
|      17 |         500 |       00:01:28 |      100.00% |              |       0.0072 |              |
0.0010 |
|      17 |         510 |       00:01:31 |      100.00% |       59.35% |       0.0068 |       1.4744 |
0.0010 |
|      18 |         540 |       00:01:36 |      100.00% |       60.46% |       0.0062 |       1.4924 |
0.0010 |
|      18 |         550 |       00:01:37 |      100.00% |              |       0.0062 |              |
0.0010 |
|      19 |         570 |       00:01:41 |      100.00% |       60.46% |       0.0084 |       1.4750 |
0.0010 |
|      20 |         600 |       00:01:46 |      100.00% |       60.72% |       0.0061 |       1.4982 |
0.0010 |
|      20 |         620 |       00:01:50 |      100.00% |       59.86% |       0.0065 |       1.5060 |
0.0010 |
|=============================================================================================================
=============|
Training finished: Max epochs completed.
|=====================================================================================================|
| Iter | Eval   | Objective   | Objective   | BestSoFar   | BestSoFar   |    numLayers |   numFilters |
|      | result |             | runtime     | (observed)  | (estim.)    |              |              |
|=====================================================================================================|
|    1 | Best   |    -0.60632 |         124 |    -0.60632 |    -0.60632 |            4 |           20 |
```

Registration No: 230118234

```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:01 |       9.38% |      13.41% |       3.8388 |       4.2492 |
0.0010 |
|       1 |          30 |       00:00:06 |      43.75% |      29.97% |       1.7636 |       2.0858 |
0.0010 |
|       2 |          50 |       00:00:08 |      59.38% |             |       1.4206 |              |
0.0010 |
|       2 |          60 |       00:00:11 |      57.81% |      43.89% |       1.1309 |       1.6308 |
0.0010 |
|       3 |          90 |       00:00:15 |      70.31% |      55.25% |       0.9150 |       1.4772 |
0.0010 |
|       4 |         100 |       00:00:17 |      85.94% |             |       0.5766 |              |
0.0010 |
|       4 |         120 |       00:00:20 |      76.56% |      57.05% |       0.5782 |       1.3651 |
0.0010 |
|       5 |         150 |       00:00:25 |      89.06% |      62.17% |       0.3774 |       1.2309 |
0.0010 |
|       6 |         180 |       00:00:30 |      89.06% |      62.77% |       0.2675 |       1.2056 |
0.0010 |
|       7 |         200 |       00:00:32 |      98.44% |             |       0.1803 |              |
0.0010 |
|       7 |         210 |       00:00:35 |      96.88% |      62.43% |       0.1424 |       1.3365 |
0.0010 |
|       8 |         240 |       00:00:41 |     100.00% |      62.85% |       0.0519 |       1.3009 |
0.0010 |
|       9 |         250 |       00:00:42 |      98.44% |             |       0.0514 |              |
0.0010 |
|       9 |         270 |       00:00:46 |     100.00% |      65.16% |       0.0279 |       1.2255 |
0.0010 |
|      10 |         300 |       00:00:50 |     100.00% |      64.47% |       0.0294 |       1.2663 |
0.0010 |
|      11 |         330 |       00:00:55 |     100.00% |      64.30% |       0.0092 |       1.2663 |
0.0010 |
|      12 |         350 |       00:00:57 |     100.00% |             |       0.0073 |              |
0.0010 |
|      12 |         360 |       00:01:00 |      96.88% |      65.76% |       0.1869 |       1.2609 |
0.0010 |
|      13 |         390 |       00:01:05 |     100.00% |      66.70% |       0.0095 |       1.2338 |
0.0010 |
|      13 |         400 |       00:01:06 |     100.00% |             |       0.0052 |              |
0.0010 |
|      14 |         420 |       00:01:10 |     100.00% |      66.44% |       0.0065 |       1.2341 |
0.0010 |
|      15 |         450 |       00:01:15 |     100.00% |      66.52% |       0.0063 |       1.2619 |
0.0010 |
|      16 |         480 |       00:01:20 |     100.00% |      65.58% |       0.0029 |       1.2749 |
0.0010 |
|      17 |         500 |       00:01:22 |     100.00% |             |       0.0043 |              |
0.0010 |
```

```
|      17 |         510 |       00:01:25 |      100.00% |       66.95% |       0.0036 |       1.2704 |
0.0010 |
|      18 |         540 |       00:01:29 |      100.00% |       65.93% |       0.0019 |       1.2888 |
0.0010 |
|      18 |         550 |       00:01:31 |      100.00% |              |       0.0095 |              |
0.0010 |
|      19 |         570 |       00:01:34 |      100.00% |       66.78% |       0.0047 |       1.2513 |
0.0010 |
|      20 |         600 |       00:01:39 |      100.00% |       65.24% |       0.0020 |       1.3105 |
0.0010 |
|      20 |         620 |       00:01:43 |      100.00% |       67.04% |       0.0024 |       1.2869 |
0.0010 |
|=========================================================================================================
=============|
Training finished: Max epochs completed.
|    2 | Best   |    -0.66695 |      112.84 |    -0.66695 |      -0.664 |           5 |          17 |
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:01 |        3.12% |       11.19% |       3.7349 |       7.1202 |
0.0010 |
|       1 |          30 |       00:00:06 |       35.94% |       25.19% |       2.0353 |       2.4321 |
0.0010 |
|       2 |          50 |       00:00:09 |       39.06% |              |       1.8054 |              |
0.0010 |
|       2 |          60 |       00:00:11 |       54.69% |       38.77% |       1.3682 |       1.9399 |
0.0010 |
|       3 |          90 |       00:00:16 |       53.12% |       48.08% |       1.6644 |       1.7102 |
0.0010 |
|       4 |         100 |       00:00:18 |       81.25% |              |       0.6434 |              |
0.0010 |
|       4 |         120 |       00:00:22 |       60.94% |       46.88% |       1.0089 |       1.7791 |
0.0010 |
|       5 |         150 |       00:00:27 |       71.88% |       50.38% |       0.7987 |       1.7533 |
0.0010 |
|       6 |         180 |       00:00:32 |       89.06% |       54.91% |       0.3469 |       1.4745 |
0.0010 |
|       7 |         200 |       00:00:34 |       98.44% |              |       0.1586 |              |
0.0010 |
|       7 |         210 |       00:00:37 |       89.06% |       54.74% |       0.3505 |       1.6128 |
0.0010 |
|       8 |         240 |       00:00:42 |      100.00% |       59.78% |       0.1089 |       1.4181 |
0.0010 |
|       9 |         250 |       00:00:43 |      100.00% |              |       0.0600 |              |
0.0010 |
|       9 |         270 |       00:00:47 |       98.44% |       60.80% |       0.0665 |       1.4068 |
0.0010 |
|      10 |         300 |       00:00:52 |       96.88% |       61.91% |       0.1015 |       1.3923 |
0.0010 |
|      11 |         330 |       00:00:57 |       98.44% |       61.57% |       0.0502 |       1.3927 |
0.0010 |
```

Registration No: 230118234

```
|      12 |         350 |       00:01:00 |       96.88% |              |       0.2092 |              |
0.0010 |
|      12 |         360 |       00:01:02 |      100.00% |       61.32% |       0.0219 |       1.4368 |
0.0010 |
|      13 |         390 |       00:01:08 |      100.00% |       63.45% |       0.0194 |       1.3942 |
0.0010 |
|      13 |         400 |       00:01:09 |      100.00% |              |       0.0155 |              |
0.0010 |
|      14 |         420 |       00:01:13 |      100.00% |       63.11% |       0.0105 |       1.3810 |
0.0010 |
|      15 |         450 |       00:01:18 |      100.00% |       63.11% |       0.0069 |       1.3909 |
0.0010 |
|      16 |         480 |       00:01:23 |      100.00% |       63.45% |       0.0052 |       1.4225 |
0.0010 |
|      17 |         500 |       00:01:25 |      100.00% |              |       0.0072 |              |
0.0010 |
|      17 |         510 |       00:01:28 |      100.00% |       64.05% |       0.0066 |       1.3987 |
0.0010 |
|      18 |         540 |       00:01:33 |      100.00% |       63.88% |       0.0058 |       1.3730 |
0.0010 |
|      18 |         550 |       00:01:34 |      100.00% |              |       0.0032 |              |
0.0010 |
|      19 |         570 |       00:01:38 |      100.00% |       64.05% |       0.0051 |       1.3991 |
0.0010 |
|      20 |         600 |       00:01:43 |      100.00% |       63.96% |       0.0052 |       1.4068 |
0.0010 |
|      20 |         620 |       00:01:47 |      100.00% |       64.65% |       0.0052 |       1.3928 |
0.0010 |
|=========================================================================================================
=============|
Training finished: Max epochs completed.
|    3 | Accept |     -0.6456 |      116.91 |    -0.66695 |    -0.66429 |           4 |          32 |
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:01 |        6.25% |       12.30% |       3.5055 |       5.5072 |
0.0010 |
|       1 |          30 |       00:00:06 |       42.19% |       28.01% |       1.8452 |       2.1969 |
0.0010 |
|       2 |          50 |       00:00:08 |       68.75% |              |       0.9791 |              |
0.0010 |
|       2 |          60 |       00:00:11 |       70.31% |       46.29% |       1.0086 |       1.5946 |
0.0010 |
|       3 |          90 |       00:00:16 |       64.06% |       54.91% |       0.9709 |       1.4368 |
0.0010 |
|       4 |         100 |       00:00:17 |       92.19% |              |       0.4086 |              |
0.0010 |
|       4 |         120 |       00:00:21 |       84.38% |       60.38% |       0.4738 |       1.3569 |
0.0010 |
|       5 |         150 |       00:00:26 |       95.31% |       60.63% |       0.2243 |       1.3791 |
0.0010 |
```

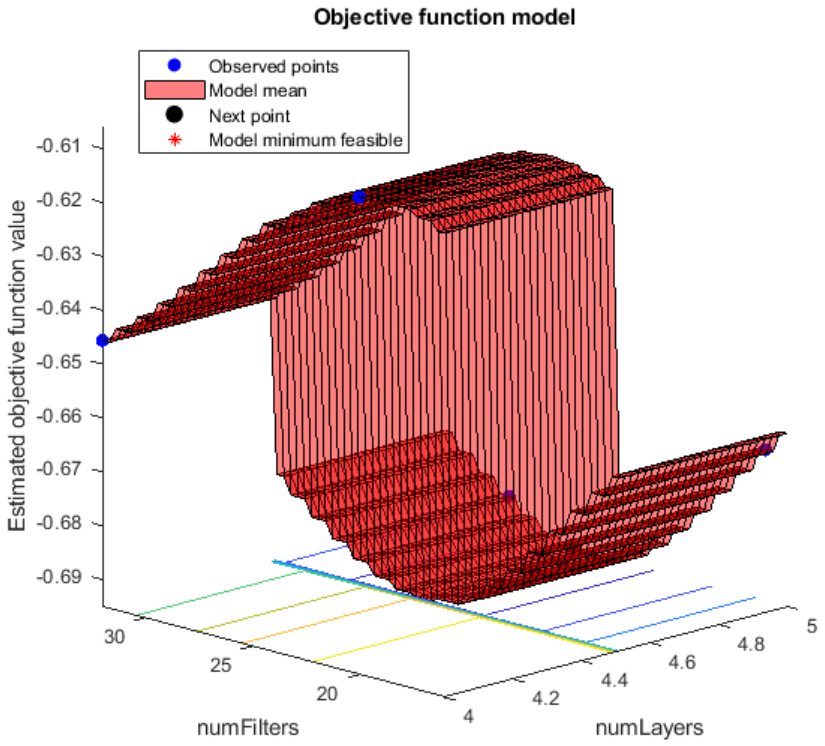
```
|       6 |         180 |       00:00:31 |       92.19% |       62.25% |       0.1965 |       1.4222 |
0.0010 |
|       7 |         200 |       00:00:33 |       98.44% |              |       0.0807 |              |
0.0010 |
|       7 |         210 |       00:00:36 |      100.00% |       67.12% |       0.0692 |       1.1888 |
0.0010 |
|       8 |         240 |       00:00:41 |      100.00% |       66.27% |       0.0366 |       1.3227 |
0.0010 |
|       9 |         250 |       00:00:42 |      100.00% |              |       0.0130 |              |
0.0010 |
|       9 |         270 |       00:00:46 |      100.00% |       67.55% |       0.0265 |       1.3479 |
0.0010 |
|      10 |         300 |       00:00:51 |      100.00% |       67.55% |       0.0072 |       1.2801 |
0.0010 |
|      11 |         330 |       00:00:56 |      100.00% |       68.92% |       0.0092 |       1.2580 |
0.0010 |
|      12 |         350 |       00:00:58 |      100.00% |              |       0.0048 |              |
0.0010 |
|      12 |         360 |       00:01:01 |      100.00% |       69.00% |       0.0051 |       1.2344 |
0.0010 |
|      13 |         390 |       00:01:06 |      100.00% |       69.43% |       0.0020 |       1.2240 |
0.0010 |
|      13 |         400 |       00:01:07 |      100.00% |              |       0.0041 |              |
0.0010 |
|      14 |         420 |       00:01:11 |      100.00% |       69.09% |       0.0022 |       1.2484 |
0.0010 |
|      15 |         450 |       00:01:16 |      100.00% |       69.85% |       0.0041 |       1.2402 |
0.0010 |
|      16 |         480 |       00:01:21 |      100.00% |       68.83% |       0.0032 |       1.2676 |
0.0010 |
|      17 |         500 |       00:01:23 |      100.00% |              |       0.0013 |              |
0.0010 |
|      17 |         510 |       00:01:26 |      100.00% |       69.60% |       0.0031 |       1.2599 |
0.0010 |
|      18 |         540 |       00:01:31 |      100.00% |       68.92% |       0.0010 |       1.2733 |
0.0010 |
|      18 |         550 |       00:01:33 |      100.00% |              |       0.0022 |              |
0.0010 |
|      19 |         570 |       00:01:36 |      100.00% |       69.77% |       0.0021 |       1.2582 |
0.0010 |
|      20 |         600 |       00:01:42 |      100.00% |       69.85% |       0.0009 |       1.2884 |
0.0010 |
|      20 |         620 |       00:01:45 |      100.00% |       69.26% |       0.0032 |       1.2569 |
0.0010 |
|=========================================================================================================
=============|
Training finished: Max epochs completed.
|    4 | Best   |    -0.69428 |      115.77 |    -0.69428 |    -0.69427 |           5 |          24 |
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
```

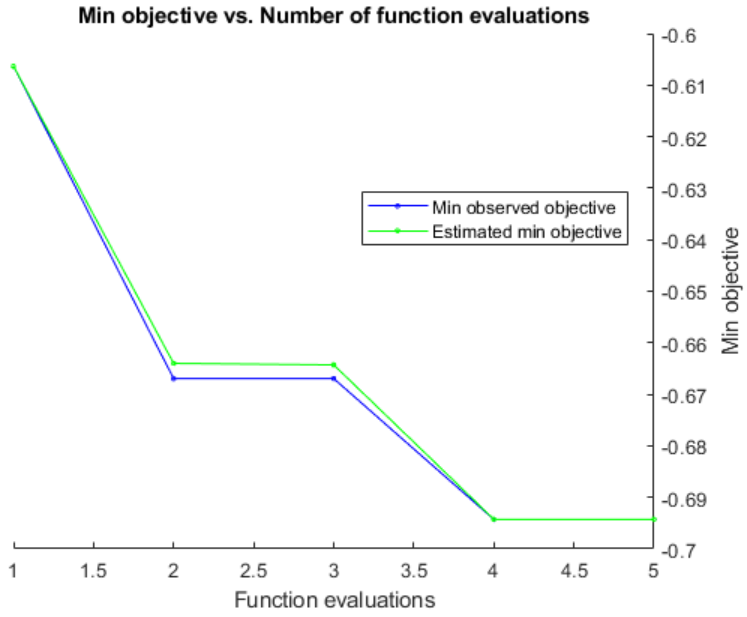
Registration No: 230118234

1	1	00:00:02	10.94%	12.13%	3.1314	8.4928	0.0010
1	30	00:00:06	39.06%	27.33%	1.7853	2.2762	0.0010
2	50	00:00:09	76.56%		0.9004		0.0010
2	60	00:00:12	54.69%	41.16%	1.5628	1.8446	0.0010
3	90	00:00:17	65.62%	54.40%	0.9967	1.4181	0.0010
4	100	00:00:18	70.31%		0.7754		0.0010
4	120	00:00:22	87.50%	57.22%	0.4413	1.4494	0.0010
5	150	00:00:28	82.81%	61.23%	0.5424	1.3339	0.0010
6	180	00:00:33	95.31%	63.36%	0.2074	1.2127	0.0010
7	200	00:00:35	100.00%		0.0718		0.0010
7	210	00:00:38	95.31%	67.29%	0.1939	1.2558	0.0010
8	240	00:00:43	96.88%	67.98%	0.1127	1.1539	0.0010
9	250	00:00:45	100.00%		0.0510		0.0010
9	270	00:00:49	98.44%	67.63%	0.0486	1.2994	0.0010
10	300	00:00:54	100.00%	67.29%	0.0192	1.1959	0.0010
11	330	00:00:59	100.00%	69.09%	0.0072	1.2431	0.0010
12	350	00:01:02	100.00%		0.0061		0.0010
12	360	00:01:04	100.00%	70.28%	0.0052	1.2308	0.0010
13	390	00:01:10	98.44%	68.74%	0.1409	1.2504	0.0010
13	400	00:01:11	100.00%		0.0027		0.0010
14	420	00:01:15	100.00%	69.09%	0.0043	1.2316	0.0010
15	450	00:01:20	100.00%	68.66%	0.0022	1.2819	0.0010
16	480	00:01:25	100.00%	68.57%	0.0016	1.2895	0.0010
17	500	00:01:28	100.00%		0.0013		0.0010
17	510	00:01:30	100.00%	69.34%	0.0020	1.2865	0.0010
18	540	00:01:36	100.00%	68.66%	0.0013	1.3025	0.0010
18	550	00:01:37	100.00%		0.0025		0.0010
19	570	00:01:41	100.00%	68.57%	0.0008	1.2818	0.0010
20	600	00:01:46	100.00%	69.17%	0.0013	1.2898	0.0010

Registration No: 230118234

```
|      20 |         620 |       00:01:50 |      100.00% |       69.17% |       0.0013 |       1.3139 |
0.0010 |
|=============================================================================================================
=============|
Training finished: Max epochs completed.
|    5 | Accept |     -0.6883 |      120.77 |    -0.69428 |    -0.69427 |            5 |           29 |
```






```
Optimization completed.
MaxObjectiveEvaluations of 5 reached.
Total function evaluations: 5
Total elapsed time: 598.1969 seconds
Total objective function evaluation time: 590.2915
```

Registration No: 230118234

```
Best observed feasible point:
    numLayers    numFilters
    _________    __________

        5            24

Observed objective function value = -0.69428
Estimated objective function value = -0.69427
Function evaluation time = 115.7659

Best estimated feasible point (according to models):
    numLayers    numFilters
    _________    __________

        5            24

Estimated objective function value = -0.69427
Estimated function evaluation time = 117.9938
```

```
% Display Best Parameters
bestParams = bestPoint(results);
fprintf('Best Number of Layers: %d\n', bestParams.numLayers);
```

```
Best Number of Layers: 5
```

```
fprintf('Best Number of Filters: %d\n', bestParams.numFilters);
```

```
Best Number of Filters: 24
```

Registration No: 230118234

Task 5: Open Ended Extension of the Project

5.1. Spectrogram Augmentation by Masking Blocks of Frequency Channels and Time Steps

Objective: Implement and evalueat the impact of spectrogram augmentation to improve model robustness and performance.

Description: This extension uses spectrogram augmentation to enhance training. The created applySpecAugment function masks random frequency channels and time steps in training images, simulating audio input variations and potentially improving model generalisation. Implementing spectrogram augmentation is much easier in Python due to the existing specAugment function.

The network architecture emphasizes convolutional, batch normalization, relu, and max pooling layers, ending with a classification layer. Training includes a mini-batch size of 64, an initial learning rate of 0.001, 30 epochs to accommodate augmentation with early stopping (if validation loss doesn't improve for 5 epochs).

Spectrogram augmentation is integrated into the data preprocessing pipeline (adsTrain.ReadFcn), dynamically altering each image during training for diversity. This technique is expected to enhance the model's robustness against speech input variations.

The effectiveness of this pectrogram augmentation is assessed by training the model with the augmented dataset, evaluating its performance on the validation set, and comparing the results to non-augmented training. The model preicts with a validation accuracy of upto 79% plus, as opposed to the 75% previously, suggesting that the spectrogram augmentation has had a positive impact on the model's ability to generalise from the training data to unseen data.

applySpecAugment function is at the end of this file (SpecAugment Function).

```
% Apply spectrogram augmentation (using specAugment function created)
adsTrain.ReadFcn = @(x) applySpecAugment(imresize(imread(x), inputSize(1:2)));

% Define Network Architecture
numClasses = numel(categories(adsTrain.Labels)); % Determine the number of classes
dynamically
timePoolSize = ceil(inputSize(1)/8);
numF = 16; % Number of filters for the convolutional layers
dropoutProb = 0.2; % Dropout probability

layers = [
    imageInputLayer(inputSize)

    convolution2dLayer(3, numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

    convolution2dLayer(3, 2*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')
```

```
    convolution2dLayer(3, 4*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

    convolution2dLayer(3, 4*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer

    convolution2dLayer(3, 4*numF, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer([timePoolSize, 1])
    dropoutLayer(dropoutProb)

    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];

% Specify Training Options
minibatchsize = 64;
options = trainingOptions('adam', ...
    'InitialLearnRate', 0.001, ...
    'MaxEpochs', 30, ...
    'MiniBatchSize', minibatchsize, ...
    'Shuffle', 'every-epoch', ...
    'Plots', 'training-progress', ...
    'Verbose', true, ...
    'ValidationData', adsValidation, ...
    'ValidationFrequency', floor(numel(adsTrain.Files)/minibatchsize))
```

```
options =
  TrainingOptionsADAM with properties:

            GradientDecayFactor: 0.9000
        SquaredGradientDecayFactor: 0.9990
                           Epsilon: 1.0000e-08
                  InitialLearnRate: 1.0000e-03
                 LearnRateSchedule: 'none'
                LearnRateDropFactor: 0.1000
                LearnRateDropPeriod: 10
                   L2Regularization: 1.0000e-04
           GradientThresholdMethod: 'l2norm'
                 GradientThreshold: Inf
                         MaxEpochs: 30
                     MiniBatchSize: 64
                           Verbose: 1
                  VerboseFrequency: 50
                    ValidationData: [1×1 matlab.io.datastore.ImageDatastore]
                ValidationFrequency: 31
                 ValidationPatience: Inf
                           Shuffle: 'every-epoch'
                    CheckpointPath: ''
```

Registration No: 230118234

```
             ExecutionEnvironment: 'auto'
                      WorkerLoad: []
                       OutputFcn: []
                           Plots: 'training-progress'
                  SequenceLength: 'longest'
             SequencePaddingValue: 0
         SequencePaddingDirection: 'right'
             DispatchInBackground: 0
          ResetInputNormalization: 1
     BatchNormalizationStatistics: 'population'
                   OutputNetwork: 'last-iteration'
```

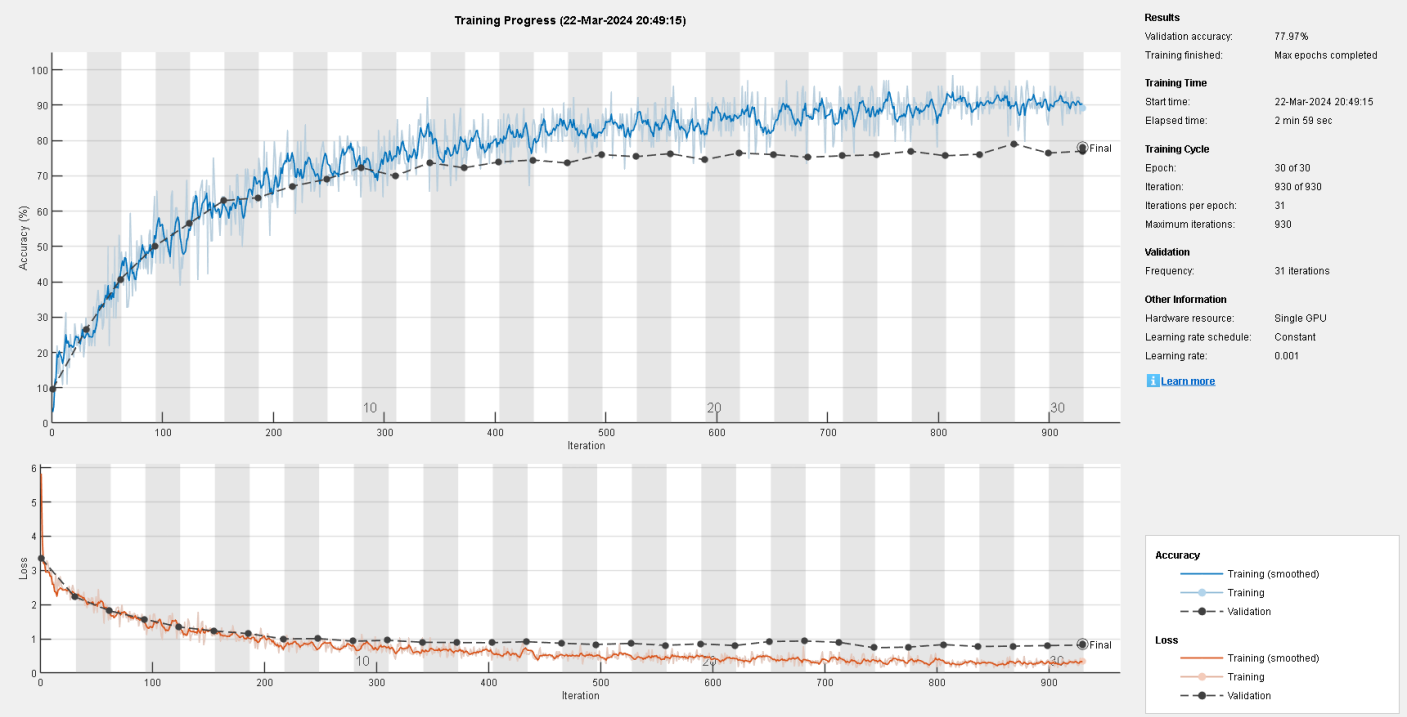
```
% Train Network
trainedNet = trainNetwork(adsTrain, layers, options);
```

```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:04 |        3.12% |        9.65% |       5.8274 |       3.3561 |
0.0010 |
|       1 |          31 |       00:00:11 |       29.69% |       26.39% |       2.2402 |       2.2275 |
0.0010 |
|       2 |          50 |       00:00:13 |       29.69% |              |       2.2321 |              |
0.0010 |
|       2 |          62 |       00:00:16 |       43.75% |       40.65% |       1.7412 |       1.8210 |
0.0010 |
|       3 |          93 |       00:00:22 |       64.06% |       50.04% |       1.2517 |       1.5749 |
0.0010 |
|       4 |         100 |       00:00:23 |       56.25% |              |       1.2448 |              |
0.0010 |
|       4 |         124 |       00:00:28 |       56.25% |       56.45% |       1.4704 |       1.3470 |
0.0010 |
|       5 |         150 |       00:00:32 |       60.94% |              |       1.2329 |              |
0.0010 |
|       5 |         155 |       00:00:34 |       64.06% |       62.94% |       1.2708 |       1.2311 |
0.0010 |
|       6 |         186 |       00:00:40 |       65.62% |       63.79% |       0.9974 |       1.1533 |
0.0010 |
|       7 |         200 |       00:00:42 |       70.31% |              |       1.0312 |              |
0.0010 |
|       7 |         217 |       00:00:46 |       76.56% |       67.04% |       0.6660 |       0.9983 |
0.0010 |
|       8 |         248 |       00:00:51 |       68.75% |       69.00% |       0.8031 |       1.0078 |
0.0010 |
|       9 |         250 |       00:00:52 |       70.31% |              |       0.9266 |              |
0.0010 |
|       9 |         279 |       00:00:57 |       64.06% |       72.33% |       0.9146 |       0.9311 |
0.0010 |
```

10	300	00:01:00	81.25%		0.5795		0.0010
10	310	00:01:03	71.88%	69.94%	0.8679	0.9558	0.0010
11	341	00:01:09	65.62%	73.70%	1.0241	0.9006	0.0010
12	350	00:01:10	78.12%		0.5502		0.0010
12	372	00:01:15	75.00%	72.25%	0.7057	0.8914	0.0010
13	400	00:01:18	81.25%		0.4851		0.0010
13	403	00:01:20	85.94%	73.87%	0.3802	0.8869	0.0010
14	434	00:01:26	84.38%	74.47%	0.5482	0.9163	0.0010
15	450	00:01:28	82.81%		0.4902		0.0010
15	465	00:01:31	81.25%	73.61%	0.5374	0.8718	0.0010
16	496	00:01:37	81.25%	76.00%	0.5376	0.8356	0.0010
17	500	00:01:38	89.06%		0.3485		0.0010
17	527	00:01:43	87.50%	75.41%	0.4253	0.8666	0.0010
18	550	00:01:46	82.81%		0.5232		0.0010
18	558	00:01:49	81.25%	76.26%	0.6625	0.8110	0.0010
19	589	00:01:54	78.12%	74.55%	0.5669	0.8531	0.0010
20	600	00:01:56	89.06%		0.4229		0.0010
20	620	00:02:00	87.50%	76.43%	0.3462	0.8049	0.0010
21	650	00:02:04	89.06%		0.3678		0.0010
21	651	00:02:06	85.94%	76.00%	0.3364	0.9217	0.0010
22	682	00:02:12	85.94%	75.23%	0.4026	0.9362	0.0010
23	700	00:02:14	89.06%		0.3152		0.0010
23	713	00:02:17	79.69%	75.66%	0.6150	0.8980	0.0010
24	744	00:02:23	89.06%	75.92%	0.3512	0.7405	0.0010
25	750	00:02:24	93.75%		0.1715		0.0010
25	775	00:02:29	79.69%	76.94%	0.6277	0.7589	0.0010
26	800	00:02:32	95.31%		0.1792		0.0010
26	806	00:02:35	90.62%	75.66%	0.3134	0.8322	0.0010
27	837	00:02:41	90.62%	76.09%	0.3470	0.7763	0.0010

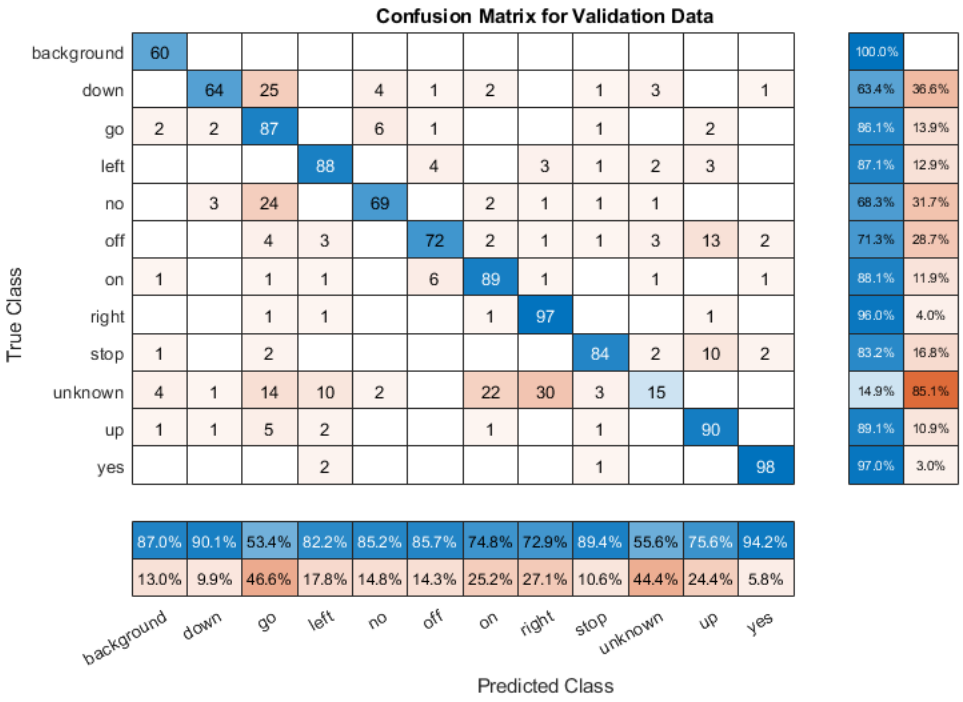
Registration No: 230118234

```
|      28 |         850 |       00:02:43 |       89.06% |              |       0.3242 |              |
0.0010 |
|      28 |         868 |       00:02:46 |       89.06% |       79.08% |       0.3355 |       0.7868 |
0.0010 |
|      29 |         899 |       00:02:52 |       84.38% |       76.43% |       0.3845 |       0.8033 |
0.0010 |
|      30 |         900 |       00:02:52 |       89.06% |              |       0.3250 |              |
0.0010 |
|      30 |         930 |       00:02:58 |       89.06% |       76.94% |       0.3377 |       0.8179 |
0.0010 |
|=========================================================================================================
=============|
Training finished: Max epochs completed.
```




```
% Plot Confusion Matrix for Validation Set
YValidation = classify(trainedNet, adsValidation);
TValidation = adsValidation.Labels;
figure('Units', 'normalized', 'Position', [0.2 0.2 0.5 0.5]);
cm = confusionchart(TValidation, YValidation, ...
    'Title', 'Confusion Matrix for Validation Data', ...
    'ColumnSummary', 'column-normalized', 'RowSummary', 'row-normalized');
sortClasses(cm, categories(adsTrain.Labels));
```

Registration No: 230118234



5.2. Guassian Noise and L2 Regularisation with Spectrogram Augmentation

To further enhance the generalisation and robustness, two additional techniques, Guassian noise and L2 regularisation, were implemented alongisde the previously implemented spectrogram augmentation, which can be found here.

Injecting Guassian noise into the training images introduces randomnes and simulates real-world imperfections in the audio data. This encourages the model to learn more generalised features rather than overfitting to the noise-free data. It tries to mirror distorted audio signals.

By incorporating L2 regularization in the training options, the model penalizes large weights, promoting simpler models that are less prone to overfitting. This regularization technique effectively constrains the model's complexity, encouraging the learning of more generalizable patterns rather than memorizing the training data.

5.3. Spectrogram Generation from 1 Second .wav Files

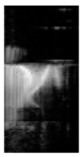
GitHub Link - https://github.com/mafazsyed/ASR_custom_voice

The ASR_custom_voice repository on GitHub contains the custom audio files recorded using high-quality microphones in Diamond's Media Booths, for all commands. These files, along with their conversions and the relevant MATLAB scripts, are included in the repository. The audio files were then cropped down to exactly 1 second .wav files; this cropping could be done using the AudioFileCropping1s.m script or using Adobe Audition, with the main audio part of each file centered within the one-second duration to match the training data (which remains the same as the provided speechImageData).

The generated one-second .wav files were then processed by a slightly modified version of the provided dataPreProcessing.m file (renamed SpectrogramGeneration.m), which converted them into grayscale spectrograms of size [98 50]. These spectrograms are then used as the validation data (with the training data remaining the same - the complete set of generated spectrograms with training data is available at customDataImage.zip) to test the model's accuracy on unseen, custom data, using the CustomAudioASR.m script.

Registration No: 230118234
Example Generated Spectrogram for "Down".

```
 url_newspec =
'https://raw.githubusercontent.com/mafazsyed/ASR_custom_voice/main/Example%20Down%20Spectrogr
am.png';
 img = webread(url_newspec);
 url_training =
'https://raw.githubusercontent.com/mafazsyed/ASR_custom_voice/main/Training%20Down%20Spectrog
ram.png';
 img_training = webread(url_training);
 imshow(img);
```



```
 imshow(img_training);
```

The spectrograms vary in their density due to inaccuracies in the dataPreProcessing.m code provided. However, the overall shape, outline, and relative densities between the different areas remain fairly consistent.

5.4. Custom Voice Testing

The model with spectrogram augmentation (masking out frequency and time channels) was used to test the custom voice validation data to for more generalisation and fit to unseeen and potentially slightly varying data. An accuracy of 70% was observed, which is plausible, however, three commands could not be detected at all, perhaps due to overfitting to the training data or the inaccuracy in the validation data spectrograms produced by the SpectrogramGeneration.m.

```
clear all;

% Define paths to your dataset
dataFolder = 'owndataImage_test';
trainDataFolder = fullfile(dataFolder, 'TrainData');
valDataFolder = fullfile(dataFolder, 'ValData');

% Create imageDatastores for training and validation datasets
```

```
 adsTrain = imageDatastore(trainDataFolder, 'IncludeSubfolders', true, 'LabelSource',
'foldernames');
 adsValidation = imageDatastore(valDataFolder, 'IncludeSubfolders', true, 'LabelSource',
'foldernames');

 % Define input size and resize the images in the datastores
 inputSize = [98 50 1];
 adsTrain.ReadFcn = @(x) applySpecAugment(imresize(imread(x), inputSize(1:2)));
 adsValidation.ReadFcn = @(x)imresize(imread(x), inputSize(1:2));

 % Define Network Architecture
 numClasses = numel(categories(adsTrain.Labels)); % Use adsTrain to determine the number of
classes dynamically
 timePoolSize = ceil(inputSize(1)/8);
 numF = 16; % Number of filters for the convolutional layers
 dropoutProb = 0.2; % Dropout probability

 layers = [
     imageInputLayer(inputSize)

     convolution2dLayer(3, numF, 'Padding', 'same')
     batchNormalizationLayer
     reluLayer
     maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

     convolution2dLayer(3, 2*numF, 'Padding', 'same')
     batchNormalizationLayer
     reluLayer
     maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

     convolution2dLayer(3, 4*numF, 'Padding', 'same')
     batchNormalizationLayer
     reluLayer
     maxPooling2dLayer(3, 'Stride', 2, 'Padding', 'same')

     convolution2dLayer(3, 4*numF, 'Padding', 'same')
     batchNormalizationLayer
     reluLayer

     convolution2dLayer(3, 4*numF, 'Padding', 'same')
     batchNormalizationLayer
     reluLayer
     maxPooling2dLayer([timePoolSize, 1])
     dropoutLayer(dropoutProb)

     fullyConnectedLayer(numClasses)
     softmaxLayer
     classificationLayer];

 % Specify Training Options
```

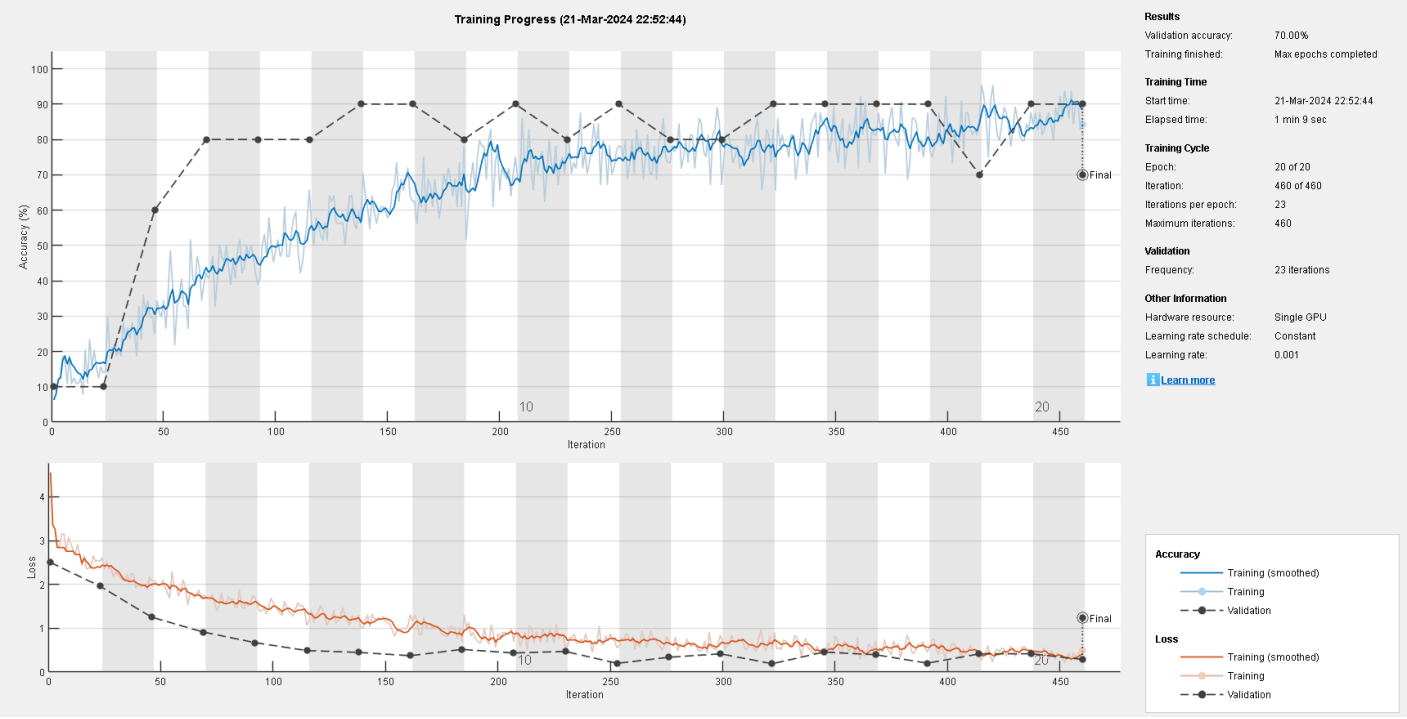
Registration No: 230118234

```
minibatchsize = 64;
options = trainingOptions('adam', ...
    'InitialLearnRate', 0.001, ...
    'MaxEpochs', 20, ...
    'MiniBatchSize', minibatchsize, ...
    'Shuffle', 'every-epoch', ...
    'Plots', 'training-progress', ...
    'Verbose', true, ...
    'ValidationData', adsValidation, ...
    'ValidationFrequency', floor(numel(adsTrain.Files)/minibatchsize));

% Train Network
trainedNet = trainNetwork(adsTrain, layers, options);
```

```
Training on single GPU.
Initializing input data normalization.
|=========================================================================================================
=============|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |
Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |
Rate       |
|=========================================================================================================
=============|
|       1 |           1 |       00:00:03 |        6.25% |       10.00% |       4.5546 |       2.5006 |
0.0010 |
|       1 |          23 |       00:00:07 |       14.06% |       10.00% |       2.5338 |       1.9592 |
0.0010 |
|       2 |          46 |       00:00:10 |       25.00% |       60.00% |       2.0886 |       1.2524 |
0.0010 |
|       3 |          50 |       00:00:10 |       34.38% |              |       1.9997 |              |
0.0010 |
|       3 |          69 |       00:00:13 |       42.19% |       80.00% |       1.7501 |       0.9126 |
0.0010 |
|       4 |          92 |       00:00:16 |       39.06% |       80.00% |       1.6940 |       0.6711 |
0.0010 |
|       5 |         100 |       00:00:17 |       48.44% |              |       1.5829 |              |
0.0010 |
|       5 |         115 |       00:00:19 |       65.62% |       80.00% |       1.0684 |       0.4905 |
0.0010 |
|       6 |         138 |       00:00:22 |       56.25% |       90.00% |       1.2975 |       0.4488 |
0.0010 |
|       7 |         150 |       00:00:24 |       57.81% |              |       1.1710 |              |
0.0010 |
|       7 |         161 |       00:00:26 |       62.50% |       90.00% |       1.3113 |       0.3731 |
0.0010 |
|       8 |         184 |       00:00:29 |       78.12% |       80.00% |       0.6962 |       0.5172 |
0.0010 |
|       9 |         200 |       00:00:31 |       65.62% |              |       0.8754 |              |
0.0010 |
|       9 |         207 |       00:00:32 |       75.00% |       90.00% |       0.7089 |       0.4356 |
0.0010 |
|      10 |         230 |       00:00:36 |       81.25% |       80.00% |       0.5561 |       0.4706 |
0.0010 |
|      11 |         250 |       00:00:39 |       70.31% |              |       0.8684 |              |
0.0010 |
```

Registration No: 230118234

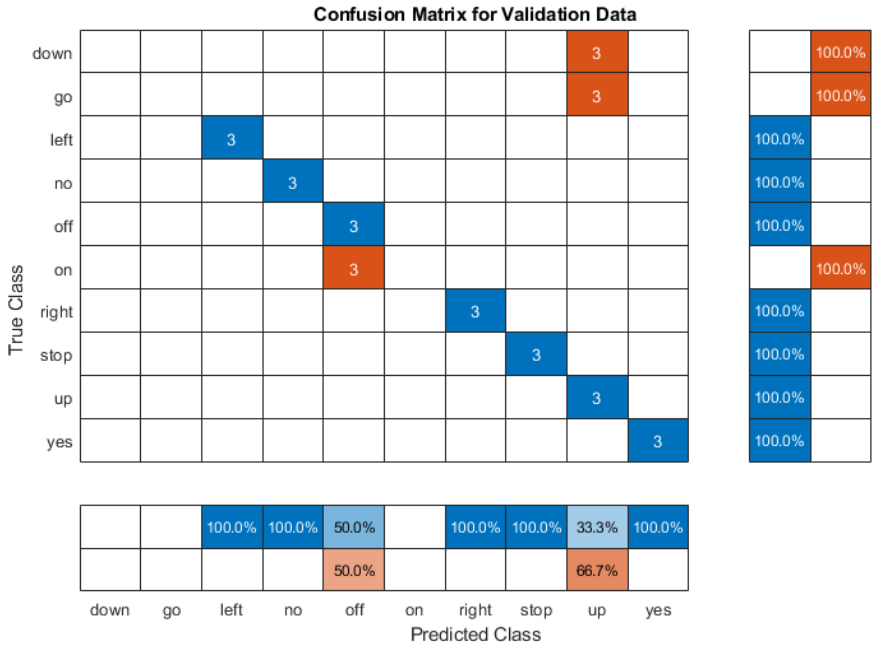
```
|      11 |         253 |       00:00:39 |       78.12% |       90.00% |       0.5489 |       0.1995 |
0.0010 |
|      12 |         276 |       00:00:43 |       73.44% |       80.00% |       0.7005 |       0.3373 |
0.0010 |
|      13 |         299 |       00:00:46 |       67.19% |       80.00% |       0.8367 |       0.4168 |
0.0010 |
|      14 |         300 |       00:00:46 |       75.00% |              |       0.6797 |              |
0.0010 |
|      14 |         322 |       00:00:49 |       82.81% |       90.00% |       0.5390 |       0.1899 |
0.0010 |
|      15 |         345 |       00:00:53 |       84.38% |       90.00% |       0.5329 |       0.4562 |
0.0010 |
|      16 |         350 |       00:00:53 |       78.12% |              |       0.8054 |              |
0.0010 |
|      16 |         368 |       00:00:56 |       81.25% |       90.00% |       0.7063 |       0.3955 |
0.0010 |
|      17 |         391 |       00:00:59 |       87.50% |       90.00% |       0.4553 |       0.2036 |
0.0010 |
|      18 |         400 |       00:01:01 |       81.25% |              |       0.5317 |              |
0.0010 |
|      18 |         414 |       00:01:03 |       85.94% |       70.00% |       0.3861 |       0.4131 |
0.0010 |
|      19 |         437 |       00:01:06 |       87.50% |       90.00% |       0.5065 |       0.4185 |
0.0010 |
|      20 |         450 |       00:01:08 |       92.19% |              |       0.3878 |              |
0.0010 |
|      20 |         460 |       00:01:09 |       84.38% |       90.00% |       0.5009 |       0.2891 |
0.0010 |
|========================================================================================================
=============|
Training finished: Max epochs completed.
```




```
% Plot Confusion Matrix for Validation Set
YValidation = classify(trainedNet, adsValidation);
```

Registration No: 230118234

```
TValidation = adsValidation.Labels;
figure('Units', 'normalized', 'Position', [0.2 0.2 0.5 0.5]);
cm = confusionchart(TValidation, YValidation, ...
    'Title', 'Confusion Matrix for Validation Data', ...
    'ColumnSummary', 'column-normalized', 'RowSummary', 'row-normalized');
sortClasses(cm, categories(adsTrain.Labels));
```



Functions Created & Used

Objective Function

```
function objective = objectiveFunction(numLayers, numFilters, adsTrain, adsValidation,
inputSize)

    numClasses = numel(categories(adsTrain.Labels));

    layers = [
        imageInputLayer(inputSize, 'Name', 'input')
    ];

    for i = 1:numLayers
        layerName = ['conv' num2str(i)];
        reluName = ['relu' num2str(i)];
        poolName = ['pool' num2str(i)];

        layers = [
            layers
            convolution2dLayer(3, numFilters * 2^(i-1), 'Padding', 'same', 'Name',
layerName)
            batchNormalizationLayer('Name', ['bn' num2str(i)])
            reluLayer('Name', reluName)
            maxPooling2dLayer(2, 'Stride', 2, 'Padding', 'same', 'Name', poolName)
        ];
    end
```

```
    layers = [
        layers
        fullyConnectedLayer(numClasses, 'Name', 'fc')
        softmaxLayer('Name', 'softmax')
        classificationLayer('Name', 'output')
    ];

    options = trainingOptions('adam', ...
        'InitialLearnRate', 0.001, ...
        'MaxEpochs', 20, ...
        'MiniBatchSize', 64, ...
        'Shuffle', 'every-epoch', ...
        'ValidationData', adsValidation, ...
        'ValidationFrequency', 30, ...
        'Verbose', true, ...
        'Plots', 'none');

    trainedNet = trainNetwork(adsTrain, layers, options);

    YValidation = classify(trainedNet, adsValidation);
    TValidation = adsValidation.Labels;
    accuracy = sum(YValidation == TValidation) / numel(TValidation);

    % Objective is to maximize accuracy, so we minimize the negative accuracy
    objective = -accuracy;
end
```

SpecAugment Function

```
function imgOut = applySpecAugment(imgIn)
    [height, width, ~] = size(imgIn);

    % Define size of the masks
    freqMaskWidth = randi([1, floor(width * 0.15)], 1); % up to 15% of the width
    timeMaskHeight = randi([1, floor(height * 0.15)], 1); % up to 15% of the height

    % Define starting points of the masks
    freqMaskStart = randi([1, width - freqMaskWidth + 1], 1);
    timeMaskStart = randi([1, height - timeMaskHeight + 1], 1);

    % Create copy of the input image to apply masks
    imgOut = imgIn;

    % Apply frequency masking
    imgOut(:, freqMaskStart:(freqMaskStart+freqMaskWidth-1), :) = 0;

    % Apply time masking
    imgOut(timeMaskStart:(timeMaskStart+timeMaskHeight-1), :, :) = 0;
end
```