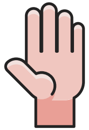
FINGERSPELLING – A SIGN LANGUAGE RECOGNTION


Tan Jing Jie
1804560
(Leader)


Ng Jan Hui
1802347


Sia Ken Yen
1804562

Tan Jing Jie
1804560

Outline

Introduction
Related Work
System Design
Experiment & Evaluation
Conclusion

Simulation Video

Tan Jing Jie
1804560

01
INTRODUCTION

Tan Jing Jie
1804560

Project Background

Project Objective

1) To design a convolutional neural network (CNN) architecture with competitive performance.

2) To investigate the relationship between reduction of color channels and model predictive performance

3) To investigate the effects of different regularization methods on model performance

4) To investigate the performance of famous CNNs when being reimplemented

Tan Jing Jie
1804560

Scope

- VGGNet16
- ResNet18
- EfficientNetB0

Tan Jing Jie
1804560

Scope

- VGGNet16
- ResNet18
- EfficientNetB0

ShallowNet

Grayscale + CustomNet

Tan Jing Jie
1804560

02
Related Work

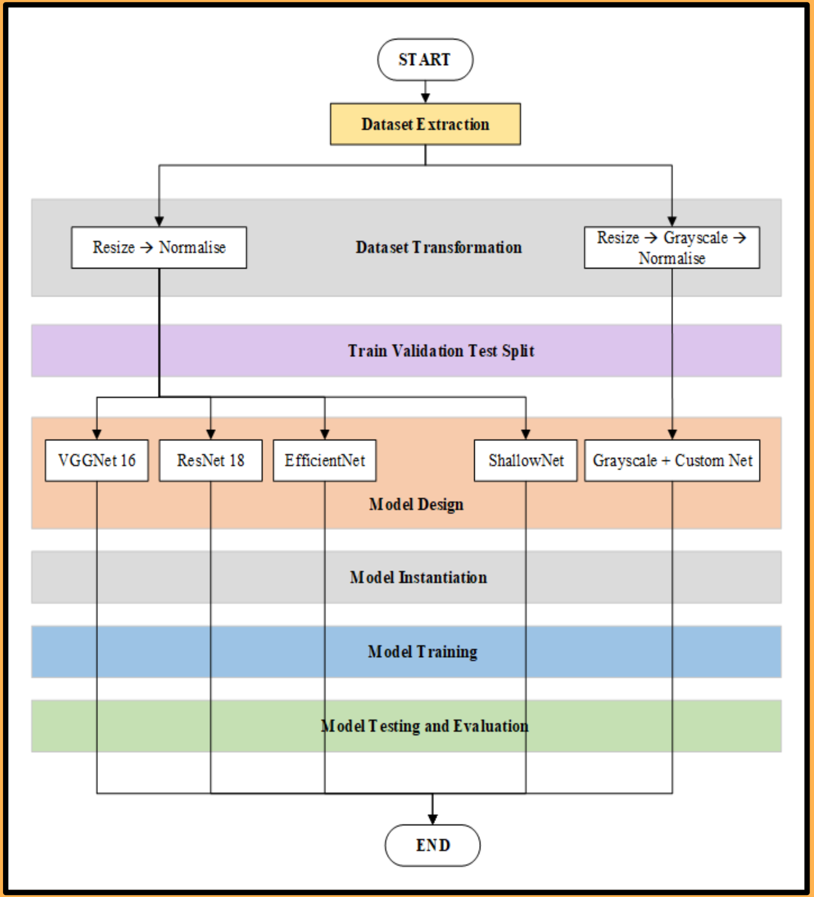
Sia Ken Yen

Authors	Paper Title	Remarks
Masood, S.et. al., 2018 [3]	American Sign Language Character Recognition Using Convolution Neural Network	- Image resize to 224 * 224 - Learning rate and weight decay - 95.54% accuracy
H.-w. Zhang [4]	Fingerspelling Identification for American Sign Language Based on Resnet-18	- Presence of global average pooling - Important for real-time use cases
N. Gupta [5]	Gesture Detection using Tensor flow Lite EfficientNet Model for Communication and E-learning Module for Mute and Deaf	
V. Bheda [6]	Using Deep Convolutional Networks for Gesture Recognition in American Sign Language	- Dropout layers - Image substraction - 82.50% accuracy
P. Das [7]	Static Hand Gesture Recognition for American Sign Language using Deep Convolutional Neural Netork	- Batch Normalization - 98.05% accuracy
A. Kasapbaşi [8]	DeepASLR: A CNN based human computer interface for American Sign Language recognition for hearing-impaired individuals	- Batch Normalization - Binarized input image - 99.38% accuracy

03
System Design

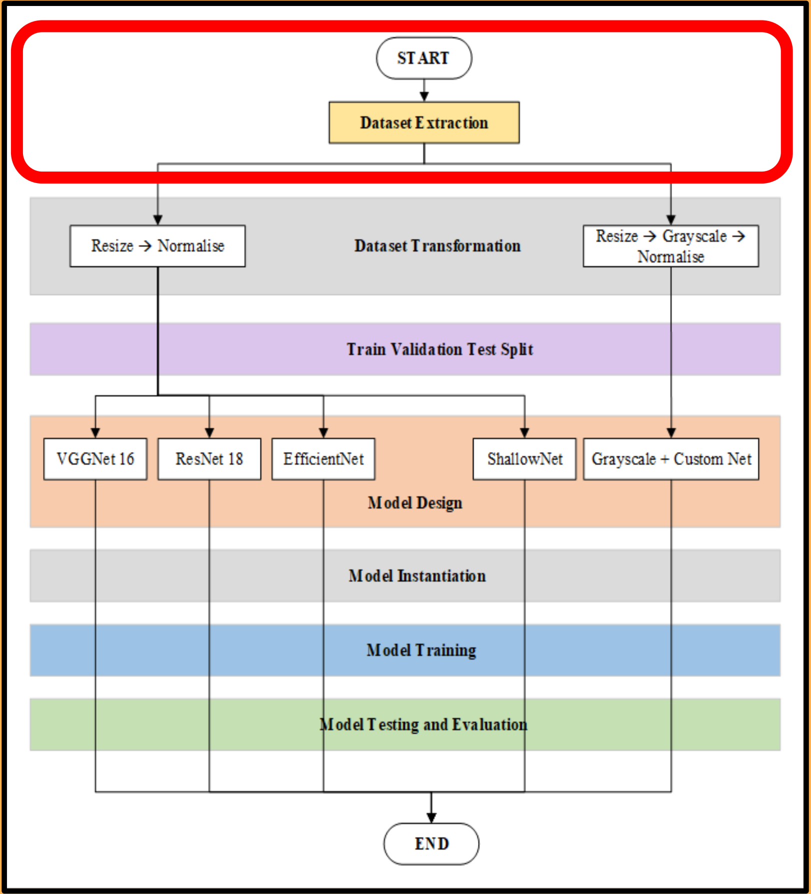
Tan Jing Jie
1804560

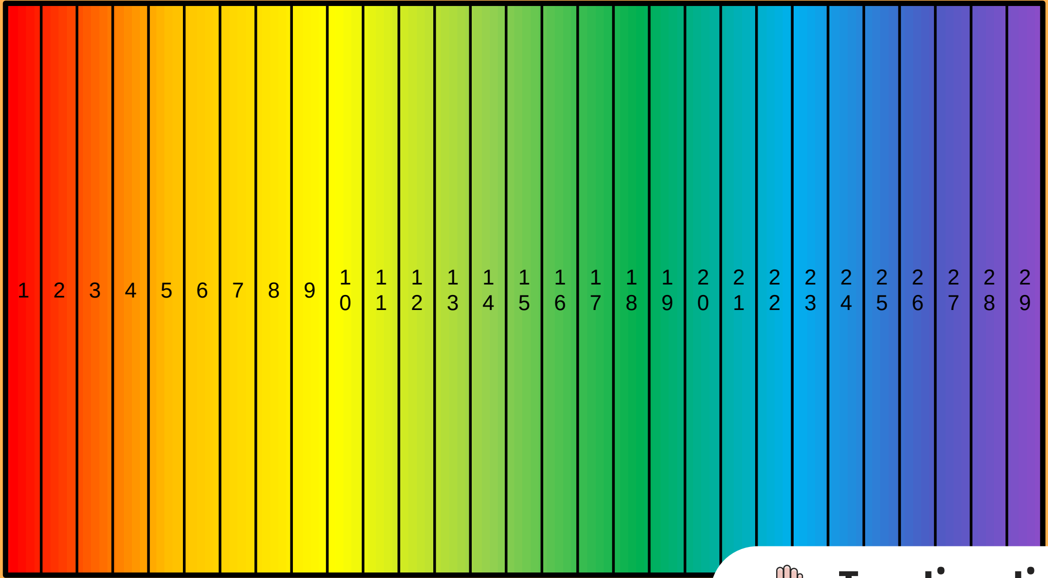
System Block Diagram



Tan Jing Jie
1804560

System Block Diagram

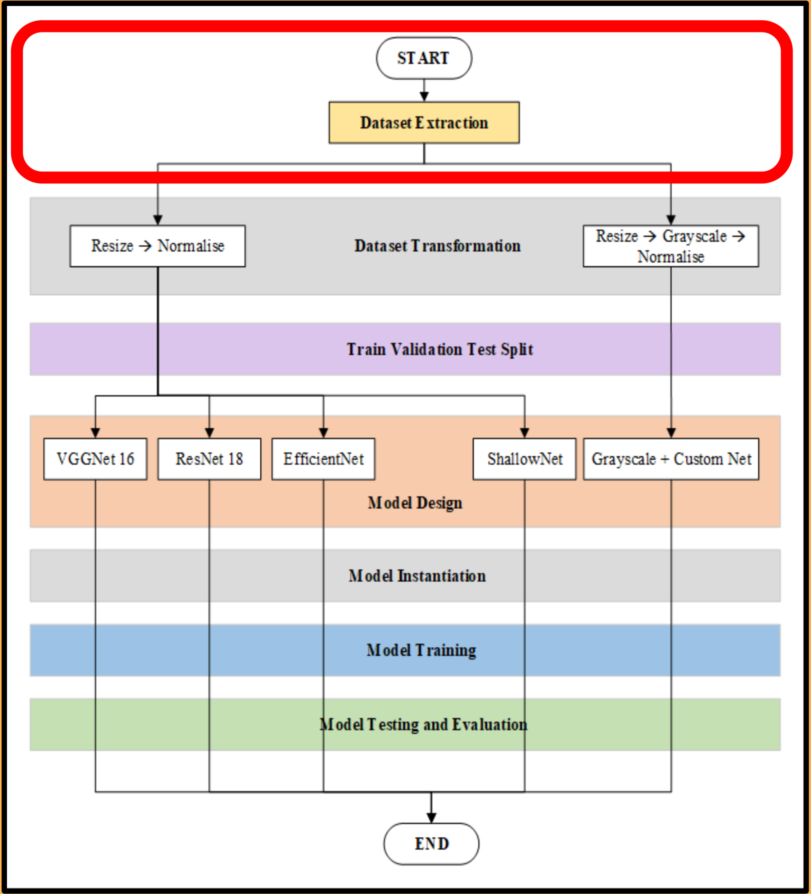


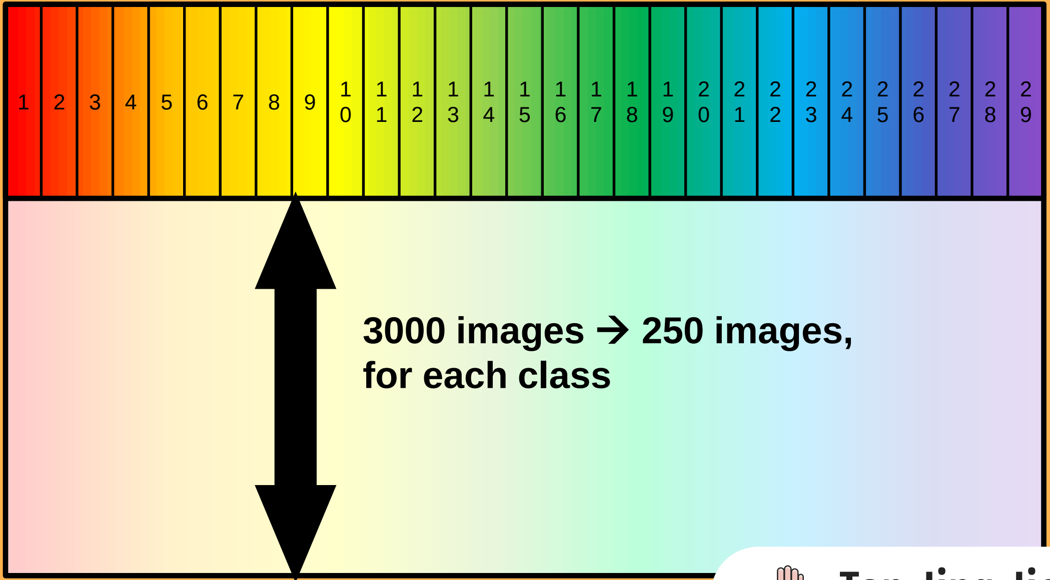


Each class having 3000 image
26 class: A – Z
3 class: del, nothing, space

Tan Jing Jie
1804560

System Block Diagram

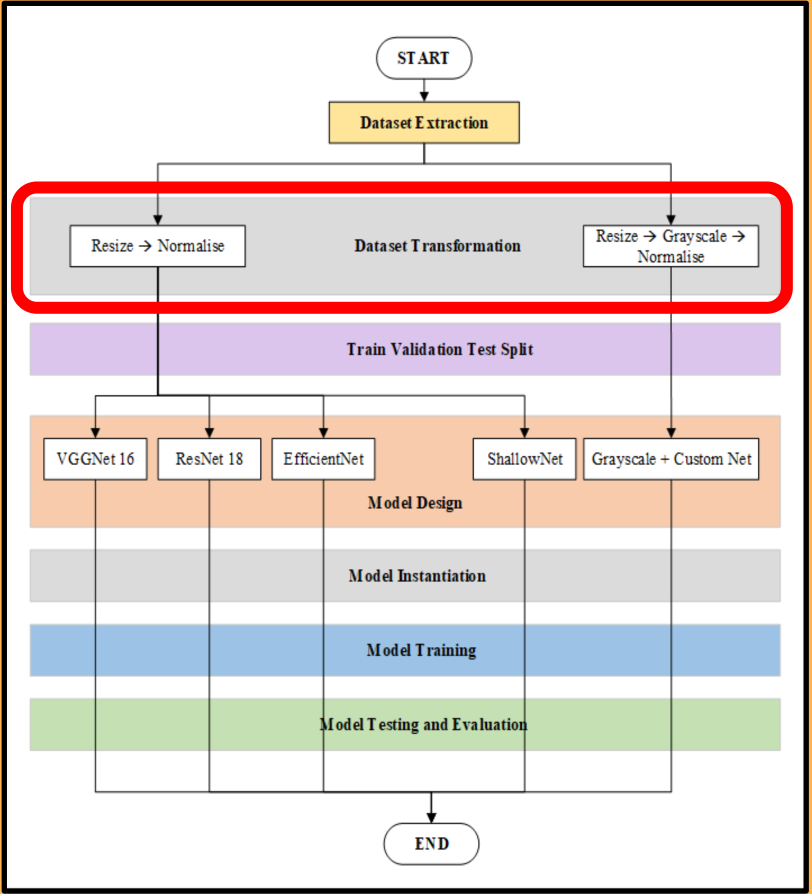




Each class having 3000 image
26 class: A – Z
3 class: del, nothing, space

Tan Jing Jie
1804560

System Block Diagram



2) To investigate the relationship between reduction of color channels and model predictive performance

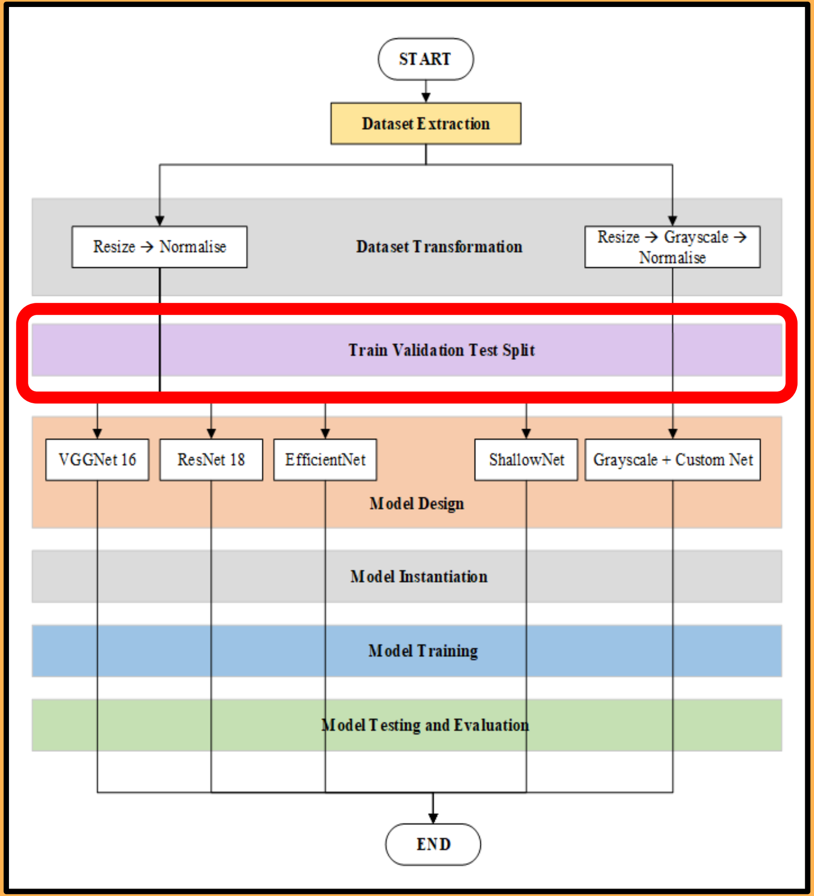
Resize to 224*224 dimension

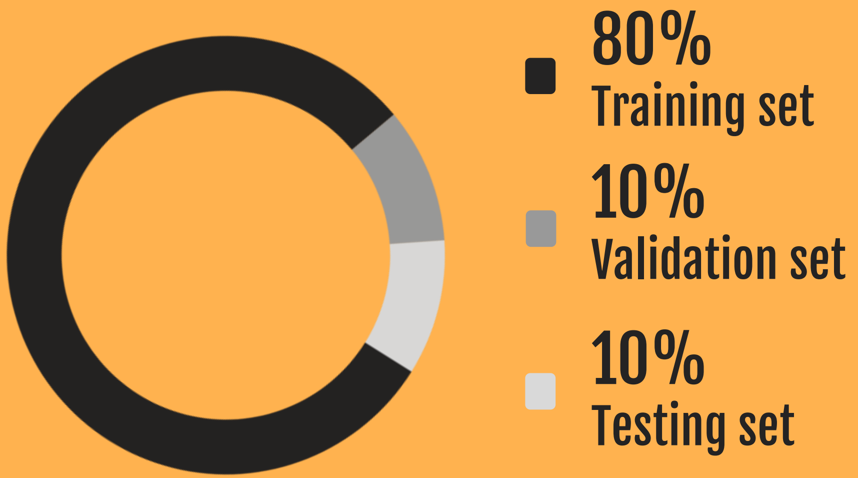
Adding grayscale transformation

Normalise

Tan Jing Jie
1804560

System Block Diagram

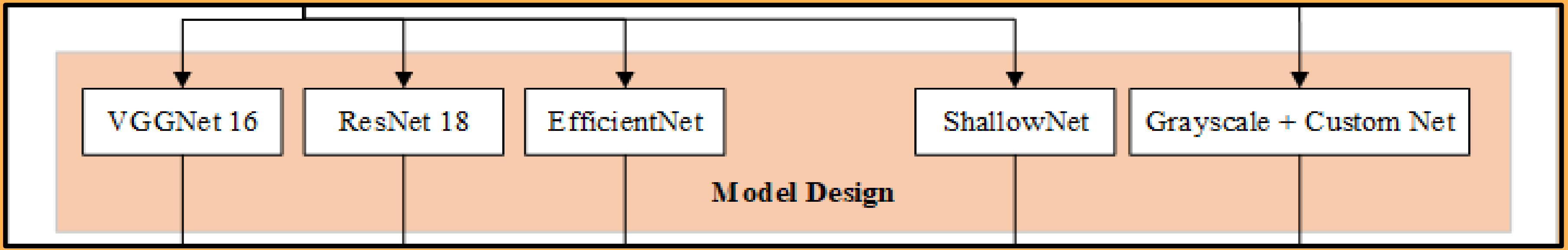




American Sign Language (ASL) Dataset

Tan Jing Jie
1804560

System Block Diagram

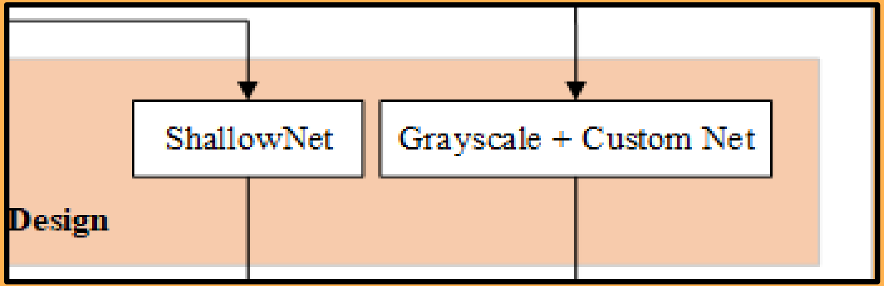


4) To investigate the performance of famous CNNs when being reimplemented

1) To design a convolutional neural network (CNN) architecture with competitive performance.

Tan Jing Jie
1804560

System Block Diagram

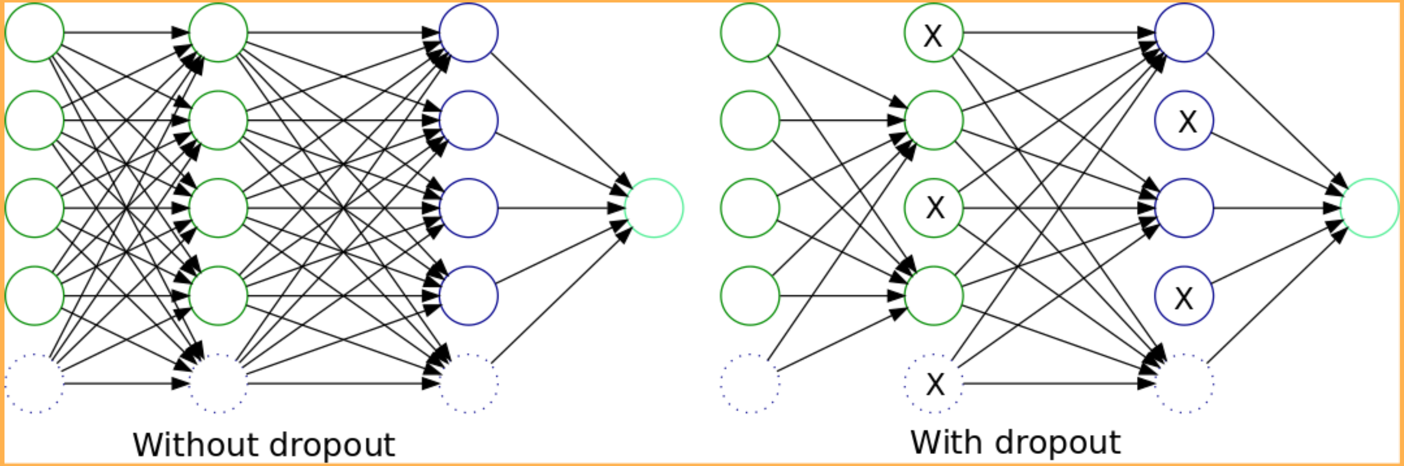


1) To design a convolutional neural network (CNN) architecture with competitive performance.

3) To investigate the effects of different regularization methods on model performance

Ng Jan Hui

System Block Diagram



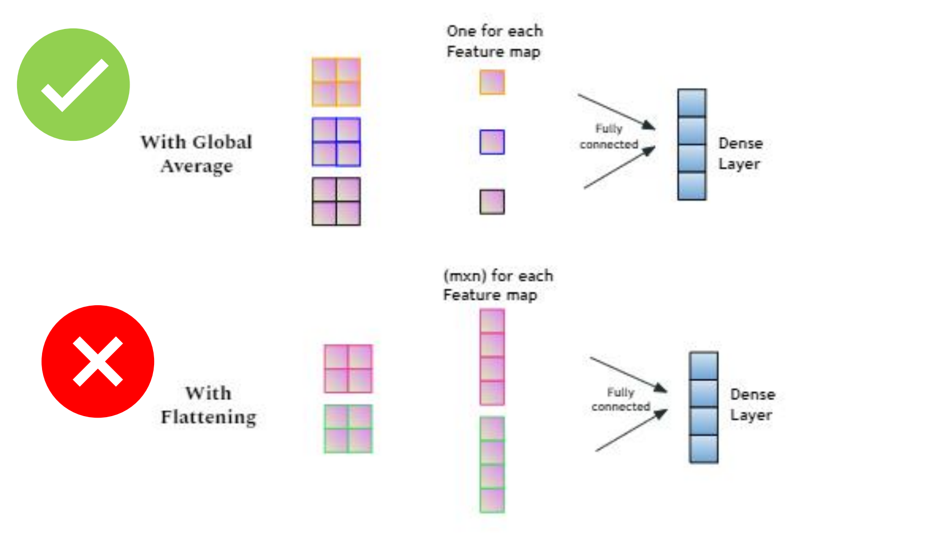
Ng Jan Hui

3) To investigate the effects of different regularization methods on model performance

System Block Diagram

Other techniques used:

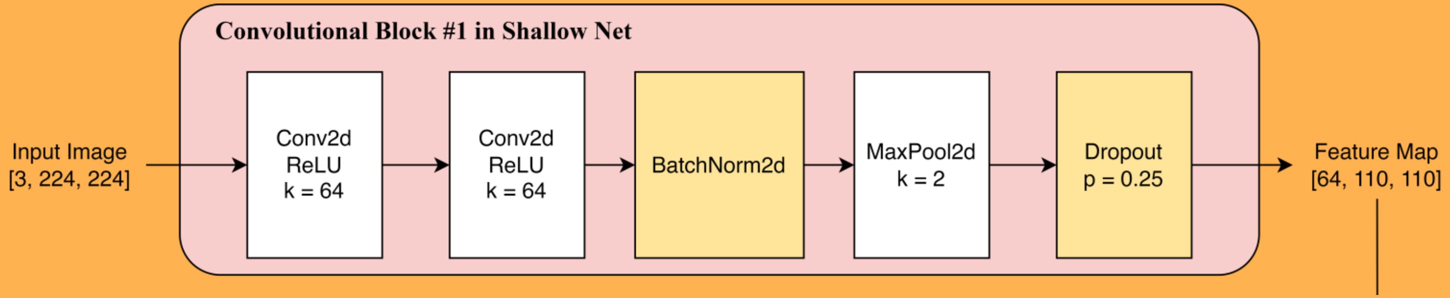
1. Batch Normalization Layers
2. Using Global Average Pooling instead of flatten layer



3) To investigate the effects of different regularization methods on model performance

Ng Jan Hui

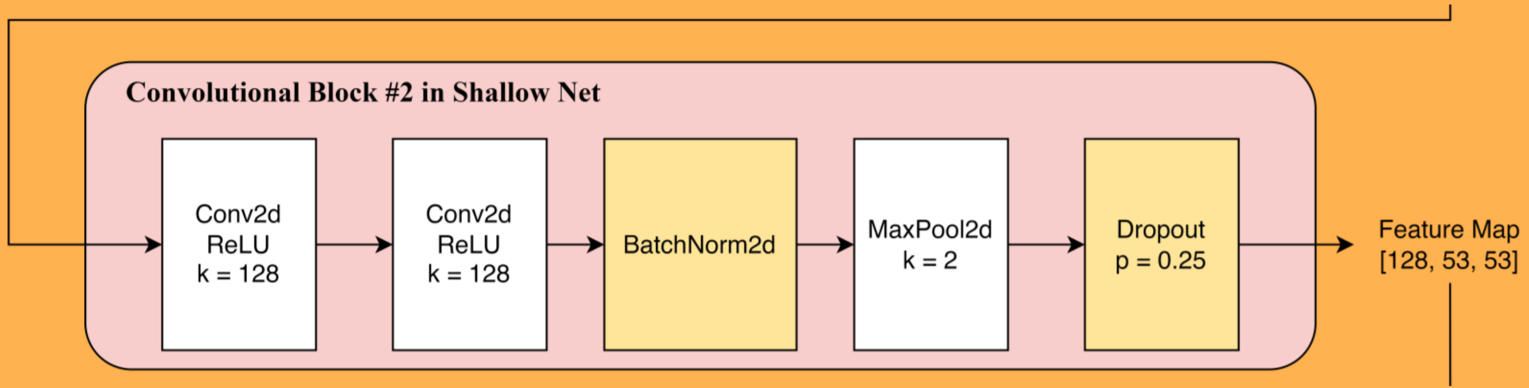
Model Architecture of ShallowNet



Ng Jan Hui

3) To investigate the effects of different regularization methods on model performance

Model Architecture of ShallowNet

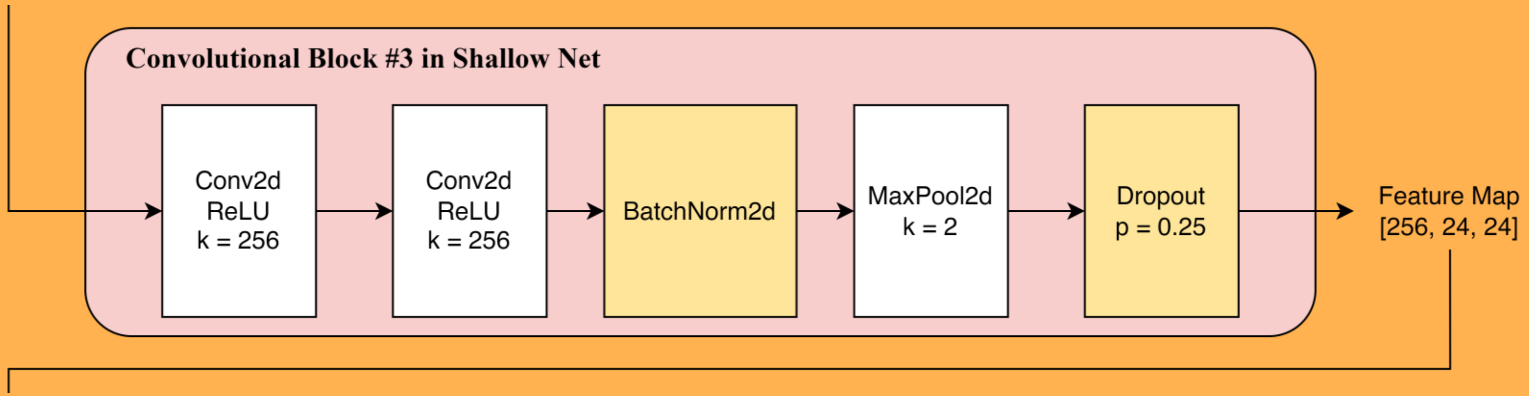


Notable Design Aspects:

- Aggressive down scaling of image dimensions
- Regularization used early

Ng Jan Hui

Model Architecture of ShallowNet

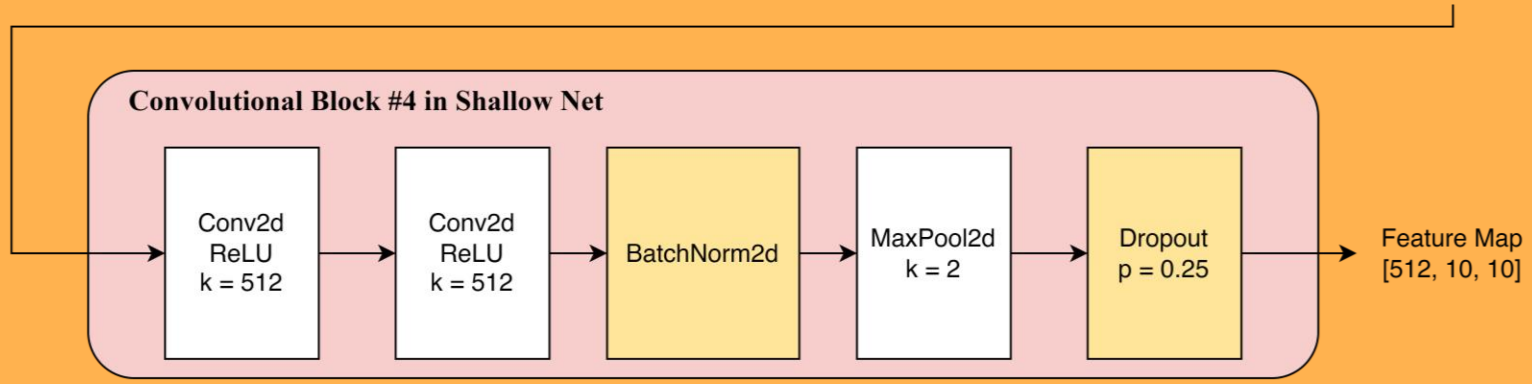


Ng Jan Hui

Notable Design Aspects:

- Convolutional Filters increase going deeper

Model Architecture of ShallowNet

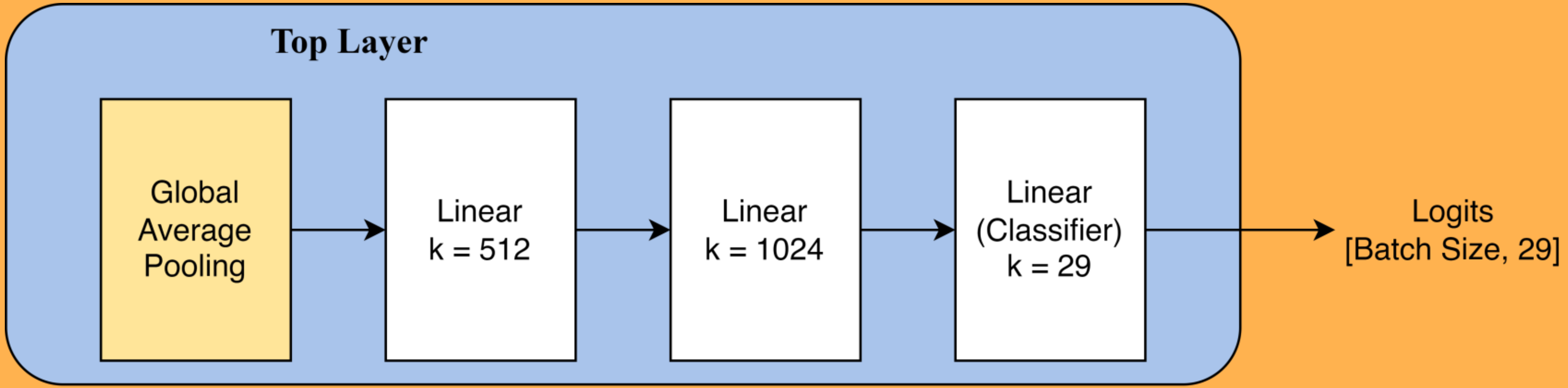


Ng Jan Hui

Notable Design Aspects:

- 4 homogenous blocks
- Spatial information dispersed across channels

Model Architecture of ShallowNet

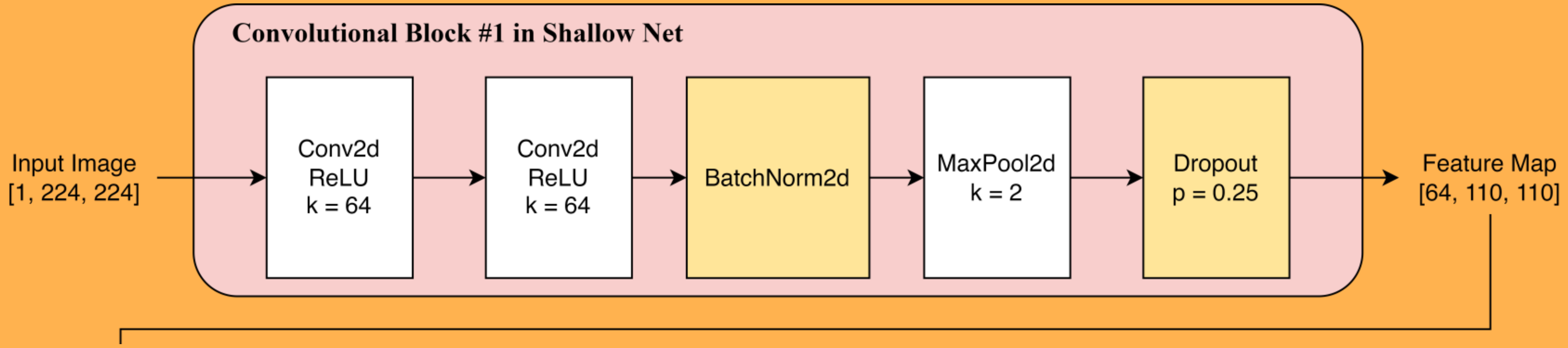


Notable Design Aspects:

- Global Average Pooling
- 1024 features learned
- Shallow, hence the name ShallowNet

Ng Jan Hui

Model Architecture of Grayscale + CustomNet

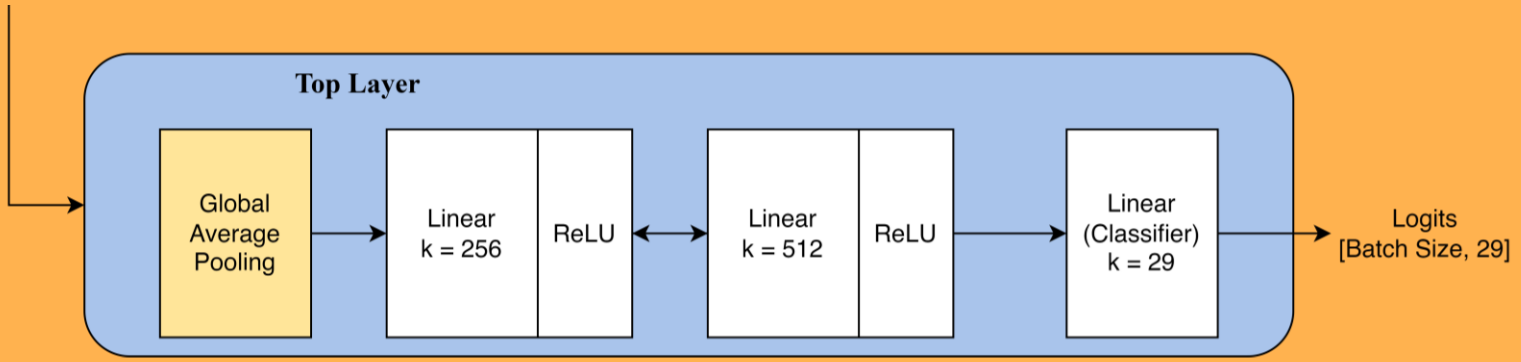


Notable Design Aspects:

- Utilizes similar design of homogenous blocks
- Input image is grayscale -> 1 channel only
- Hence, the name Grayscale

Ng Jan Hui

Model Architecture of Grayscale + CustomNet

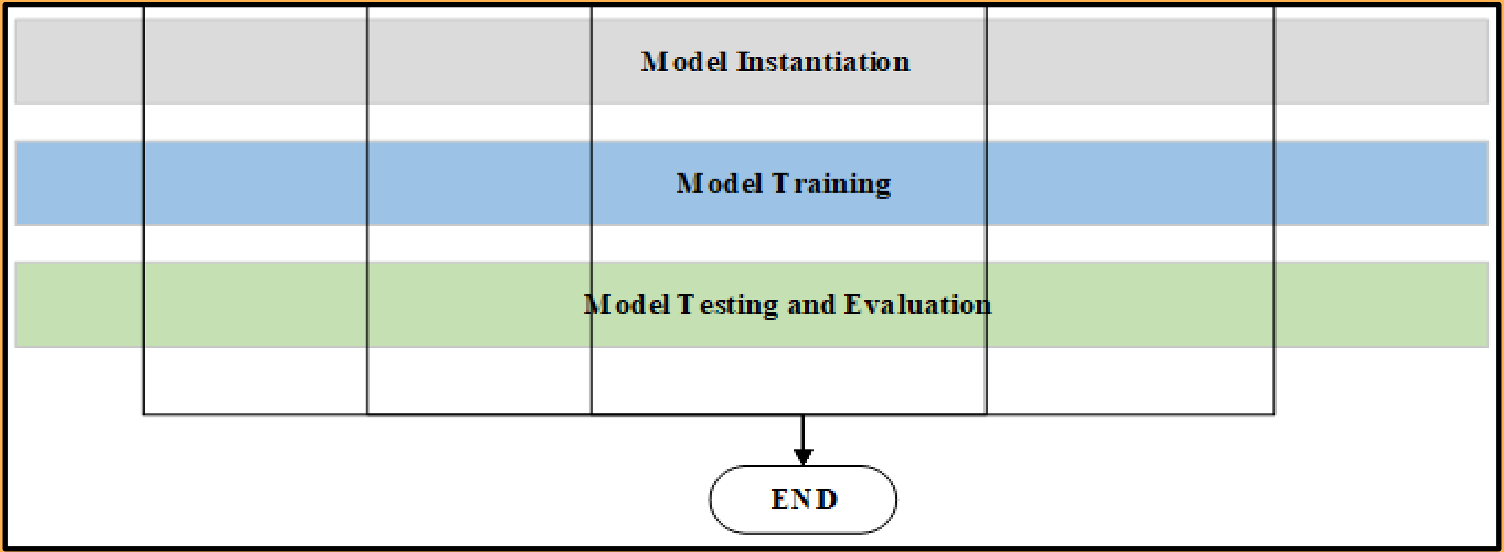


Notable Design Aspects:

- Less features learned
- Inline with objective 2

Ng Jan Hui

System Block Diagram



Training multiple models and tabulate their performance

Ng Jan Hui

04
Experiment & Evaluation

Tan Jing Jie
1804560

Fine tune direction

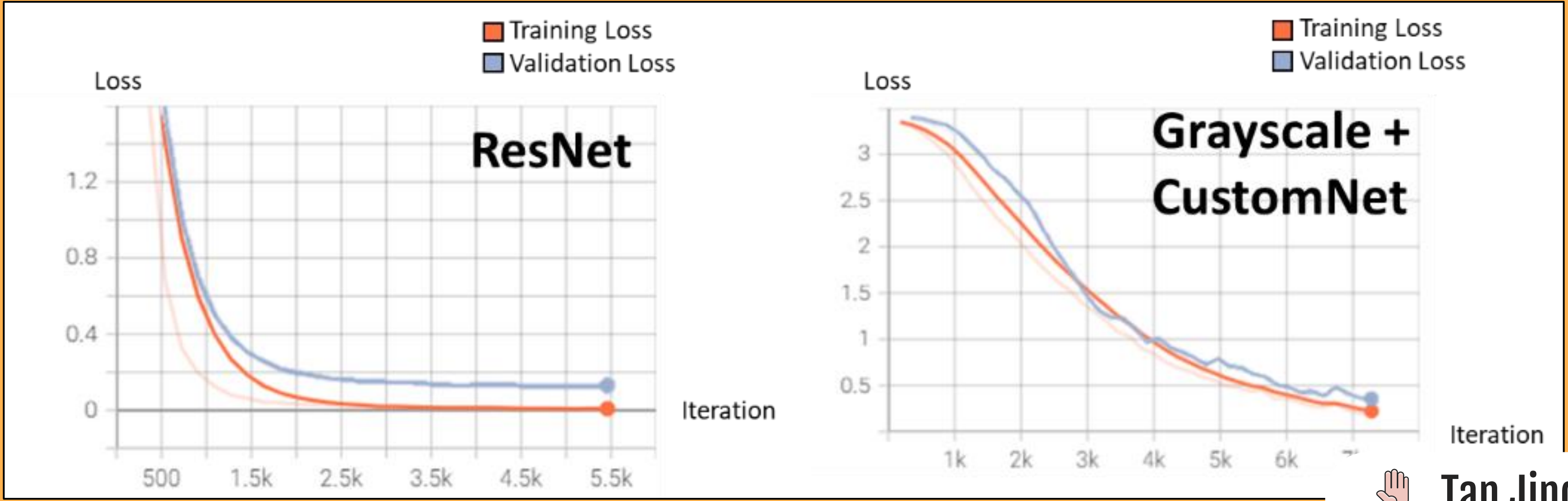
Condition	Fine tune action
The training loss and validation loss is decreasing, but training has ended.	Increase number of epochs
The training loss and validation loss are jittery	Decrease learning rate
The training loss is not decreasing	Decrease filter size

Fine tune of this project include:

Epoch, learning rate, filter size, number of layers, etcetera

Tan Jing Jie
1804560

Training Loss and Validation Loss Visualisation



Tan Jing Jie
1804560

Study the trainability of model:
No overfitting issue occur

Comparison of model for
Layer vs Loss & Layer vs Inference Time

Architecture type	Transfer Learning			Proposed		
Model Net	**VGGNet**	**ResNet**	**EfficientNet**	**ShallowNet**	**Grayscale + CustomNet**	
Number of trainable layers	1	2	2	10	8	
Training Loss (epoch =30)	0.6140	0.0071	0.4685	0.0593	Epoch=30 0.4391	Epoch=65 0.0271
Validation Loss (epoch =30)	0.4350	0.0332	0.4638	0.1924	Epoch=30 0.4823	Epoch=65 0.0506
Inference time (s)	0.0104	0.0091	0.0489	0.0808	**0.0063**	

The higher the number of trainable layers, the higher the difficulty for a model to converse

The shallow the layer in CNN model, the faster the inference time.

Tan Jing Jie
1804560

Model Evaluation and Performance Metrics

Model	Method	Precision	Recall	F1 Score	Accuracy	Inference time
VGG16	Transfer Learning	0.86	0.87	0.85	85.64%	0.01042 seconds
EfficientNetB0	Transfer Learning	0.94	0.93	0.93	93.09%	0.04892 seconds
ResNet18	Transfer Learning	0.99	0.99	0.99	99.17%	0.009099 seconds
ShallowNet	Self-Designed	0.99	0.99	0.99	99.17%	0.0808 seconds
Grayscale + CustomNet	Self-Designed	0.98	0.98	0.98	98.07%	0.0063 seconds

Key Takeaway:

- Our self designed models were able to beat the pretrained models
  - Regularization played an important role in small datasets

Ng Jan Hui

Model Benchmarking

Authors	Model	Accuracy	Inference time
Garcia and Viesca, 2016 [9]	GoogLeNet transfer learning	70%	-
Kania, K.et. al., 2018 [10]	Wide Residual Networks transfer learning	93.30%	-
Masood, S.et. al., 2018 [3]	VGG16 transfer learning	94.68%	0.0104 seconds
Our work	**ShallowNet**	**99.17%**	**0.0808 seconds**
	Grayscale + CustomNet	**98.07%**	**0.0063 seconds**

1) To design a convolutional neural network (CNN) architecture with competitive performance.

2) To investigate the relationship between reduction of color channels and model predictive performance

Ng Jan Hui

Experiments

Hypothesis 1 : “**Reimplementation of prominent CNN models affects the training process due to the improper initialization of weights**.”

Testing : Experimenting the recreated VGG model by training on CIFAR10 dataset

Result : - Training loss decreasing
- Hypothesis rejected

Hypothesis 2 : “**Training on images with lesser than 3 channels, would cause the model to learn non-representative features and perform badly**.”

Testing : implementing Grayscale with CustomNet

Result : - Predictive performance of Grayscale + CustomNet able to outperform the pretrained models
- Hypothesis rejected

Experiments

Hypothesis 3 : “**Models that are built from scratch are prone to overfitting.**”

Testing : training the models built from scratch such as CustomNet and ShallowNet and observing its performance

Result : - Training times at par with the pretrained models and with better predictive performance
- Hypothesis rejected

Hypothesis 4 : “**Models that are built from scratch may outperform pretrained models on simple datasets**.”

Testing : Implementation of Grayscale with CustomNet and ShallowNet

Result : - Predictive performance of CustomNet and ShalloNet able to beat pretrained networks with less inference time
- Hypothesis accepted

05
Conclusion

Sia Ken Yen

Conclusion

- Objectives of the project were archieved

- Developed ShallowNet and CustomNet that outperform established CNN architectures

- Pre-trained and deeper models are able to train with a shorter time, but custom-built models that are usually less complex and shallow, tailored to perform better

- The future work for this research would be focusing on reducing training time

Demostration

Thank you