

# Deep Learning Project

Team ID: CS\_H15

1-Youssef Mohamady Abdel Fatah Mohamady

ID: 20191700789

2-Nada Tarek Hassan Emam

ID: 20191700682

3- Menna Tulla Yehia Hamdy

ID: 20191700663

4-Monira Osama Ahmed Diab

ID: 20191700667

5-Esraa Mohamed Saeed

ID: 20191700102



# Agenda

Preprocessing

Model 1

Model 2

Model 3







# Introduction

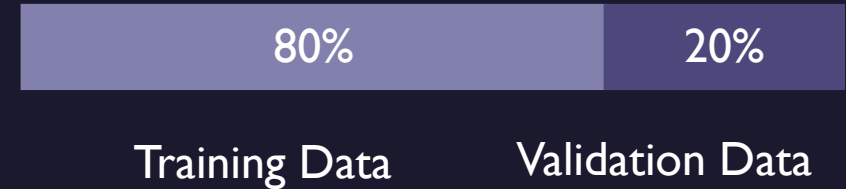
- In today's world of internet, a massive amount of data is getting generated every day and content-based classification of images is becoming an essential aspect for efficient retrieval of images and have attracted application in several fields and one of such field is sports. Building a model that is able to classify different sports activities into different categories could be useful for automated sports analysis tasks.

# Preprocessing

Preparing the data for Training

# Preprocessing

- 1- Load Dataset for Training by c2v
- 2- Create six list for six sports
- 3-Split each list 80% for training and 20 % for validation
- 4- Concatenate five list for one list
- 5- 1300 image for training and 300 validation



# Model 1

AlexNet Architecture

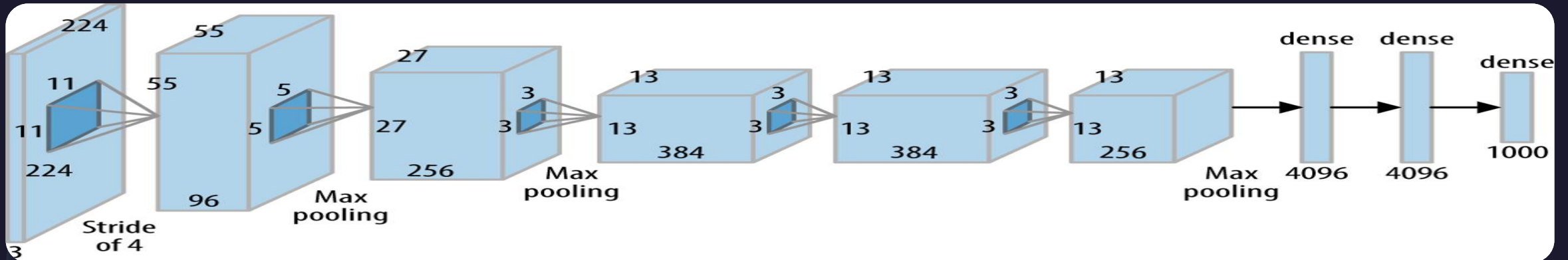


# AlexNet

1-Model Preprocessing resize image (224,224,3)

2- Normalize image from 0 to 1

3- Architecture



# AlexNet

## 4- After my modifications on AlexNet

- For input layer is  $227 \times 227$  instead of  $224 \times 224$
- For first convolution is  $7 \times 7$  filter size instead of  $11 \times 11$
- For last layer (Output) is 6 instead of 1000

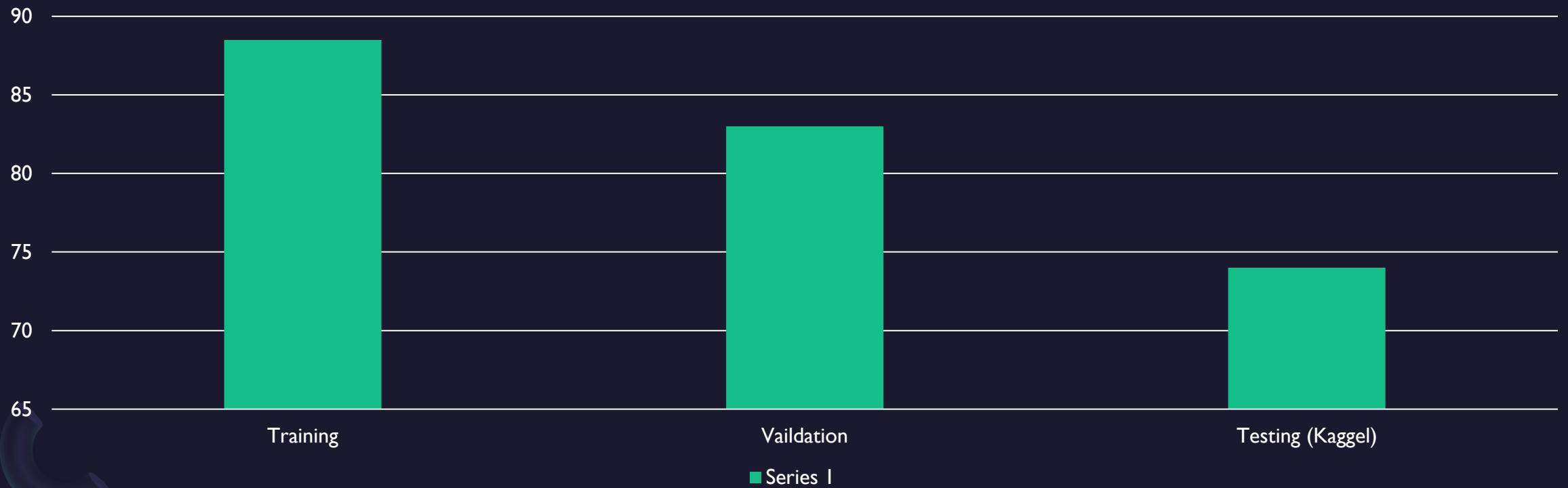
## 5- Use optimizer SDG (LR=0.0001, momentum=0.9)





# AlexNet

Model Accuracy



# AlexNet

## Conclusion :

- When use Architecture AlexNet I got bad accuracy but when some modification this accuracy become better
- These modifications are a similar to the ZFNet model
- Use Early Stopping technique to avoid overfitting



# Model 2

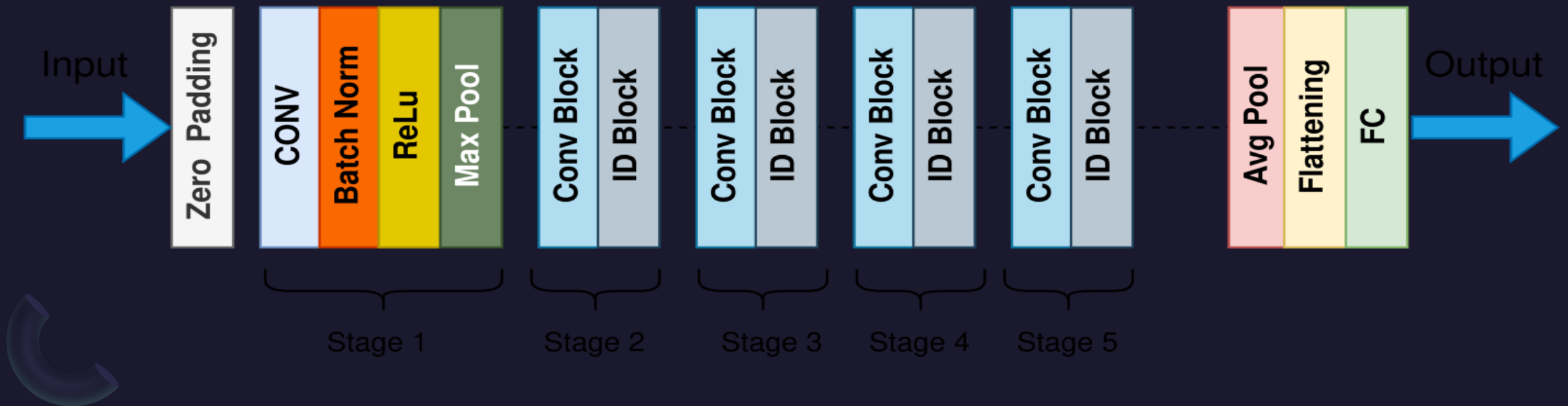
ResNet50 Architecture

# ResNet50

1-Model Preprocessing resize image (224,224,3)

## 2- Architecture

ResNet50 Model Architecture





# ResNet50

4- After my modifications on ResNet50

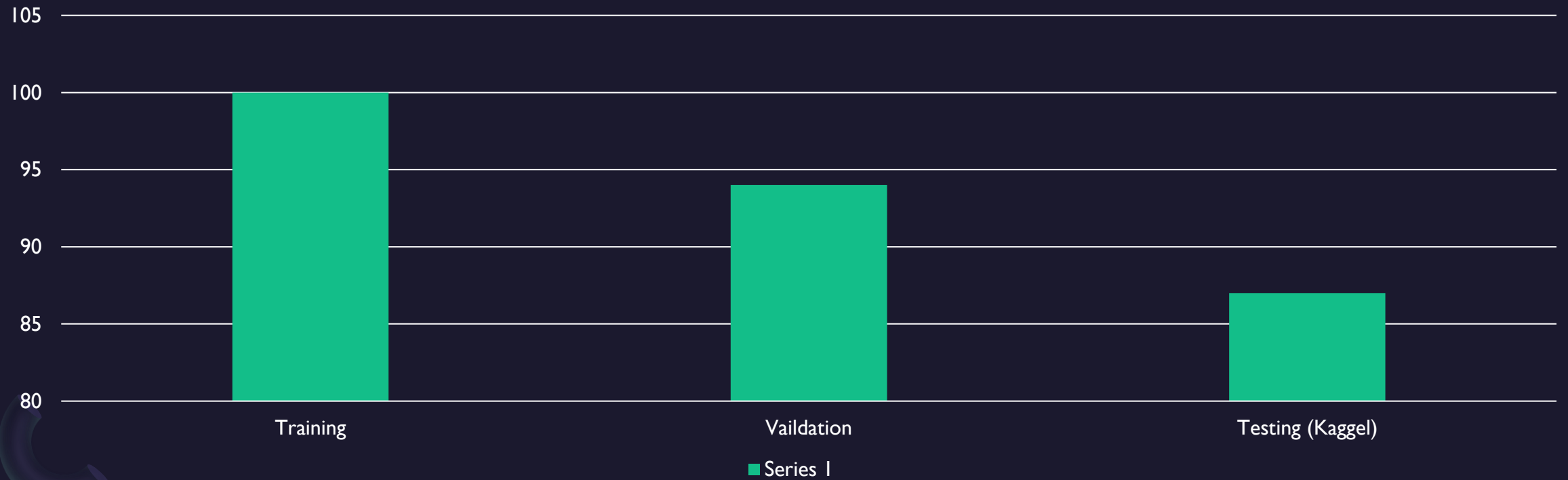
- For last layer (Output) is 6 instead of 1000

5- Use optimizer Adam



# ResNet50

Model Accuracy



# ResNet50

## Conclusion :

- When use Architecture ResNet50 I got best accuracy
- ResNet50 architecture is one of the best model for classification
- Use Early Stopping technique to avoid overfitting



# Model 3

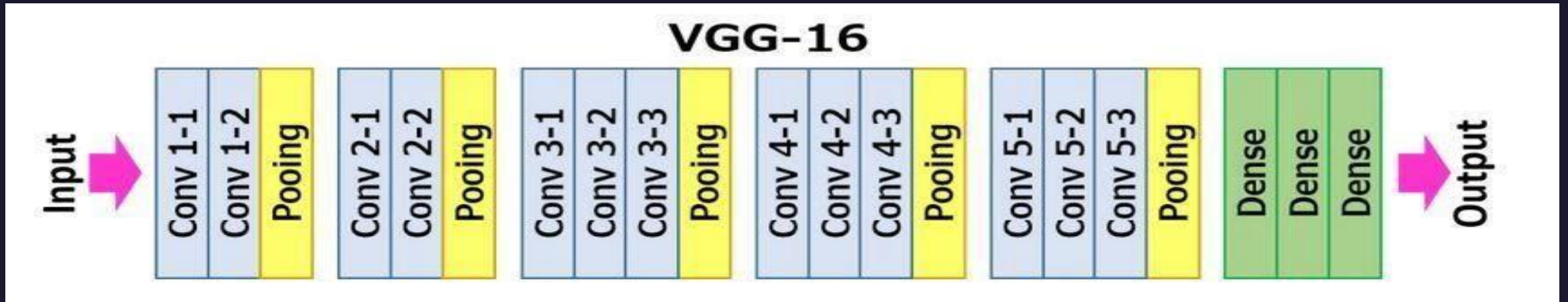
VGGI6 Architecture



# VGG16

1-Model Preprocessing resize image (224,224,3)

2- Architecture



# VGG16

4- After my modifications on VGG16

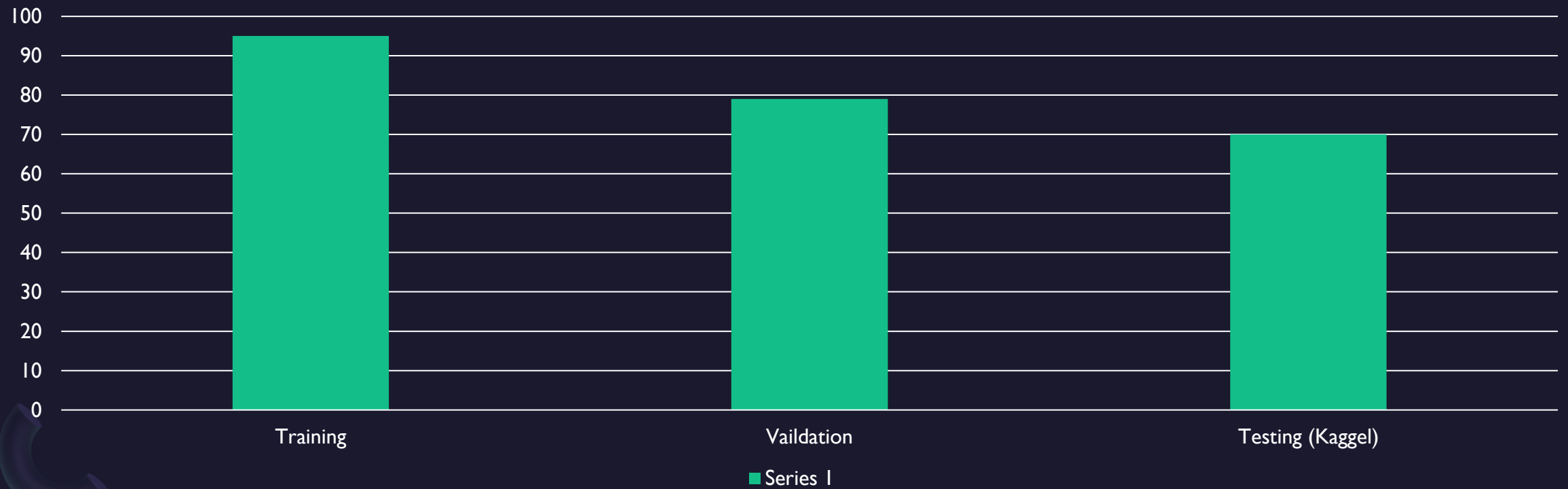
- For last layer (Output) is 6 instead of 1000
- Reduce number of filter in each layer to half

5- Use optimizer RMSprop(LR=0.0001)



# VGG16

Model Accuracy



# VGG16

## Conclusion :

- When use Architecture VGG16 I got bad accuracy but when some modification this accuracy become better
- Because vgg16 is deeper model and high computations when make this less computations by less filters I got better accuracy
- Use Early Stopping technique to avoid overfitting







# Summary

- Finally when use alexnet accuracy 74% but resnet50 accuracy 87% on Kaggle website
- In test script will use resnet50 model for predicate

# Thank You

