VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590018

TECHNICAL SEMINAR REPORT
ON
"A REVIEW OF CONVOLUTIONAL NEURAL NETWORKS"

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Lokesh R

USN: 1KS18CS040

Under the guidance of

Mr. Kumar K

Assistant Professor

Department of Computer Science & Engineering

K.S.I.T. Bengaluru-560109

Department of Computer Science & Engineering

K.S. INSTITUTE OF TECHNOLOGY

#14, Raghuvanahalli, Kanakapura Road, Bengaluru- 560109

2021-2022

K.S. INSTITUTE OF TECHNOLOGY

BENGALURU-560109

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that the Technical Seminar entitled **"A Review of Convolutional Neural Networks"** presented by **Lokesh R, USN: 1KS18CS040 of VIII semester** in partial fulfillment of the award of Bachelor of Engineering in CSE in Visvesvaraya Technological University, Belagavi during the academic year **2021-2022**. The Seminar Report has been approved as it satisfies the academic requirements in respect of Seminar work prescribed for the Bachelor of Engineering degree.

----------	----------	----------
Signature of the Guide	Signature of the H.O.D.	Signature of the Principal
Mr. Kumar K	Dr. Rekha.B.Venkatapur	Dr. Dilip Kumar K
Assistant Professor,	Professor, Head of the Dept.,	Principal/Director,
K.S.I.T., Bengaluru	K.S.I.T., Bengaluru	K.S.I.T., Bengaluru

Name of the student:	University Seat Number:
Lokesh R	1KS18CS040

ACKNOWLEDGEMENT

The successful presentation of the seminar would be incomplete without the mention of the people who made it possible and whose constant guidance crowned my effort with success.

I take this opportunity to express my sincere gratitude to our **Management** K S Institute of Technology, Bengaluru for providing the environment to present the seminar.

I would express my gratitude to **Dr. K. V. A. Balaji,** C.E.O. K.S. Institute of Technology, Bengaluru, for facilitating me to present the seminar.

I would like to extend my gratitude to **Dr. Dilip Kumar K.,** Principal/Director, K.S. Institute of Technology, Bengaluru, for providing opportunity to present the seminar.

I thank **Dr. Rekha B. Venkatapur**, Professor and Head, Department of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru, for her encouragement.

I whole heartedly thank Technical Seminar Coordinators, **Mr. Krishna Gudi and Mr. Kumar K**, Department of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru, for their support and guidance.

I would also like to thank my seminar guide **Mr. Kumar K** Assistant Professor, Department of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru, for her constant support.

Finally, I would like to thank all the teaching and non-teaching staff of the college for their co-operation. Moreover, I thank all my **family** and **friends** for their invaluable support and cooperation.

Lokesh R

1KS18CS040

ABSTRACT

Prior Convolutional Neural Networks became prominent, computer classification challenges required gathering characteristics from data that was either inefficient or inaccurate. Convolutional Neural Networks have recently endeavoured to deliver a higher level of performance and accuracy in all of the sectors where they have been used, the most famous of which would be Object Recognition, Digit, and Image Recognition. It uses a well-defined set of procedures to follow, including Backpropagation, Convolutional Layers, Feature Formation, and Pooling. In addition, this post will look at how to leverage several frameworks and tools that use the CNN model.Because they can handle a huge number of datasets, artificial neural networks have shown to be the most effective in deep learning. The Convolutional Neural Network is the most popular (CNN). It has been demonstrated to be helpful for computer vision, pattern recognition, and NLP. Because it uses fewer parameters and focuses more on features particular to a given domain than traditional Neural Nets, CNN is so widely utilised. Many CNN architectures, including LeNet, AlexNet, and GoogleNet, have been proposed. In this essay, we discuss the construction of CNN and every CNN model that has been put forth to yet.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

The neural networks have been designed with the goal of closely mimicking how the human brain operates. Convolutional Neural Networks contribute to this by working with the visual sensory organs of living things and, in the process, recognising various types of objects, be they digital or images or a specific action in any object, using a series of different techniques followed in a particular order, namely Convolutional Operation, ReLu Layer, Pooling, Flattening, and Softmax Cross Entropy. When Hubel and Wiesel released a study on the visual cortices of primates and birds, it marked the beginning of CNN's development.

In the fields of computer vision and image recognition, Deep Convolution Neural Network has now produced state-of-the-art work. Because the hidden layers are not completely connected to the preceding layers and do numerous subsequent calculations between convolution and pooling, CNN is so successful (subsampling layer). In contrast to other neural network models, CNN is simple to train thanks to backpropagation because each layer has very little connectivity. Convolution is accomplished using a linear filter. The mathematical process convolution, which means to roll together (employ together) two or more mathematical operations like sigmoid, leaky ReLU, or Tanh, gave rise to the name of the convolution neural network.

Then, in the 1980s, Kunihiko Fukushima created the convolution process known as neocognitron, which was motivated by the work of Hubel and Wiesel. However, Yann Le Cunn, who created the LeNet-5 convolutional network with seven levels and adaptive weights for different parameters, was largely responsible for elevating CNN to its current state. Different iterations of LeNet-5 make up all the major architectures used today.

Chapter 2

LITERATURE SURVEY

[1] A Review of Convolutional Neural Networks

Arohan Ajit, Koustav Acharya, Abhishek Samanta in their paper have included various phases in Convolutional neural networks, namely convolutional layer what they are, how they function, how they use filters to gather important pixels from images, how they use activation functions, the various types of activation functions, and problems they encounter include the point that they need a lot of data to achieve good accuracy. Pre-trained models were therefore presented by Alex Krizhevsky; these models use less data but yet achieve a high level of accuracy.

[2] Diving deep in Deep Convolutional Neural Network

Divya Arora, Mehak Garg, Megha Gupta, The paper's conclusion states that convolutional neural networks have established themselves as crucial tools in the field of machine learning. Although there are still unresolved problems, such as lowering the Fully Connected layer's parameters and lowering the error margin whenever the structure is subject to intensive data. It has been demonstrated that more layers not only improve performance but also add new parameters with each layer.

Chapter 3

Methodologies

3.1 Convolutional operation

Convolutional layer : CNN uses grayscale and RGB images as input. Each image is represented by a pixel that has a range of 0 to 255. In order to create a feature map or image, convolution is applied to the input data or image using a filter, or kernel in the case of a CNN. Certain details/features from the image are collected via filters. Ex: n-f+1, where n is the image size and f is the filter size.

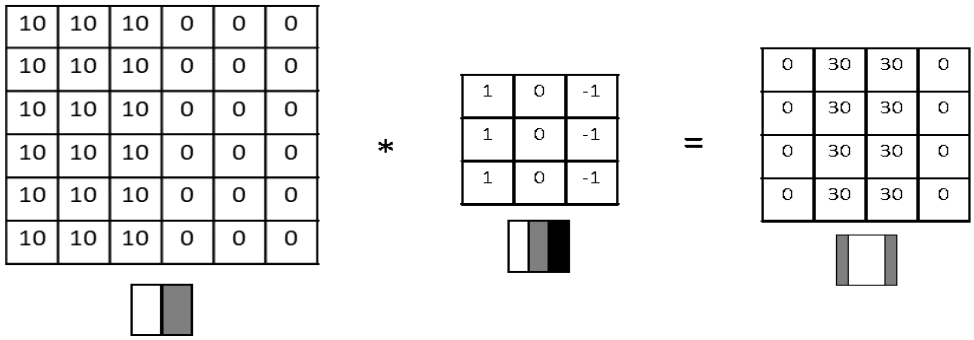

Fig 3.1.1 convolution layer

Filters/kernels : Feature detectors are another name for kernels and filters. An edge detection filter is used for feature detection. To identify the edges of objects within images, edge detection is a technique used in image processing. It operates by spotting variations in brightness, as well as other things like vertical and horizontal edge detection.

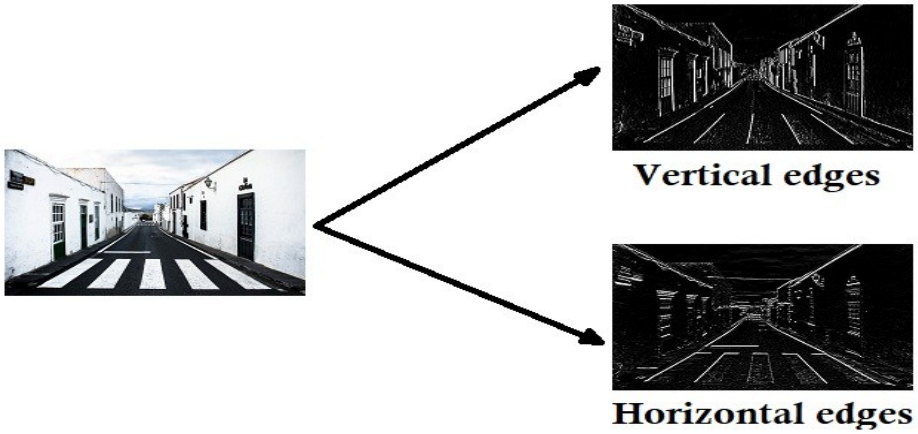

Fig 3.1.2 Filters

Padding : By applying a filter on a regular image, we risk losing some information; therefore, we put another layer on top of the image to prevent this. In order to prevent the issues outlined above, padding is merely the process of adding layers of zeros to our input photos. The padding formula is n+2p-f+1, where p is the number of padding layers.

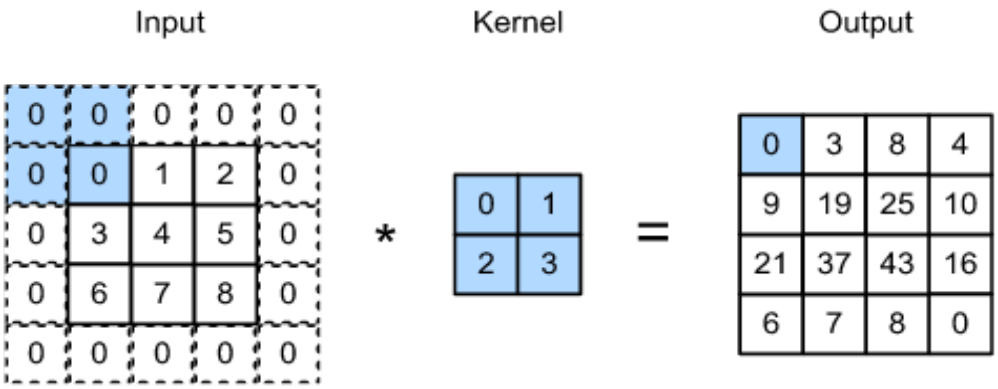

Fig 3.1.3 padding

Stride : Stride is how far the filter moves in every step along one direction. formula for stride is n+2p-f+1/s , s is stride step, n is the size of image, p – padding size.

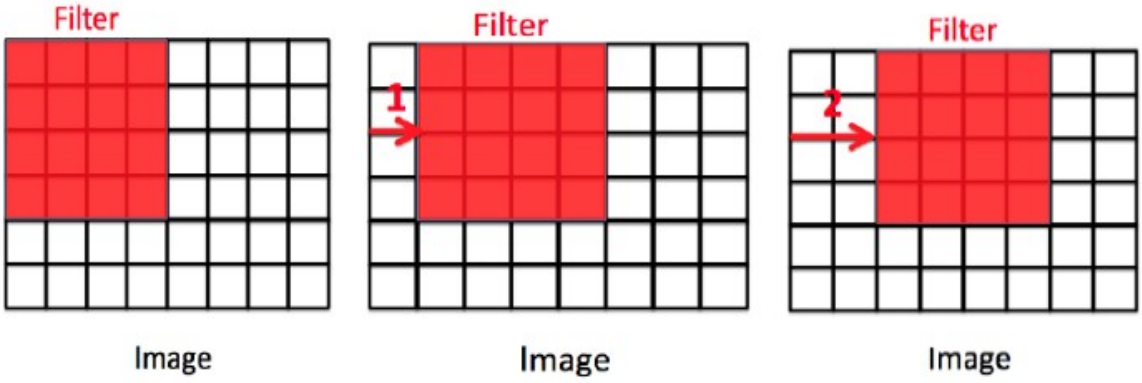

Fig 3.1.4 Stride

Pooling : In order to extract the most crucial features, the pooling step, which comes after the convolution process, creates a pooled feature map. Max, minimum, and average pooling are all options. If the input image is slightly translated to a new position, then it is also perfectly identified during the pooling stage of CNN. This is how the pooling layer solves difficulties like location invariance. There are different pooling types namely Max pooling: The batch's maximum pixel value is chosen. Min pooling: The batch's minimum pixel value is chosen. Average pooling selects the average value of all pixels in the batch.

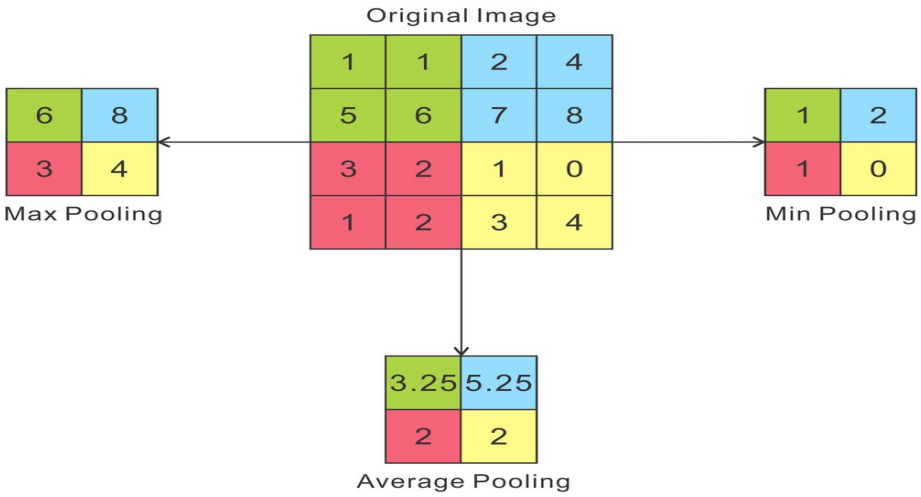

Fig 3.1.5 pooling

Activation Function : The activation function is a straightforward function that returns the output of the node. It is utilised to determine the neural network's output, such as yes or no. The obtained values are mapped between 0 and 1 or -1 and 1, etc.

- **Sigmoid Function :** The logistic regression function known as the sigmoid condenses the range of values between -1 and +1 and delivers the likelihood that the output will be positive. f(x)=1/ 1+e −x

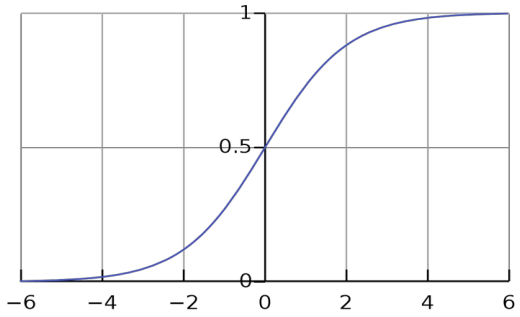

Fig 3.1.6 sigmoid function

- **Tanh :** Tanh compresses the output to between -1 and +1. tanh(x) = 2f(2x) – 1

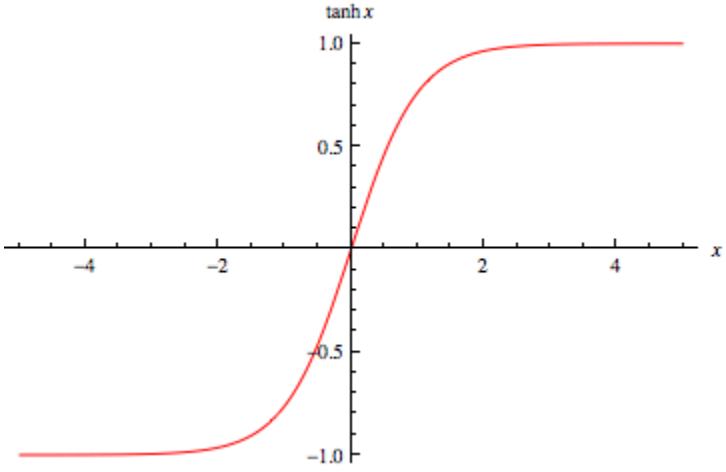

Fig 3.1.7 Tanh function

- **ReLU :** ReLU function is frequently utilised as the fundamental component of neural networks. If the output is negative, it returns 0, and if it is positive, it returns the actual value. rect(x) = max(0, x)

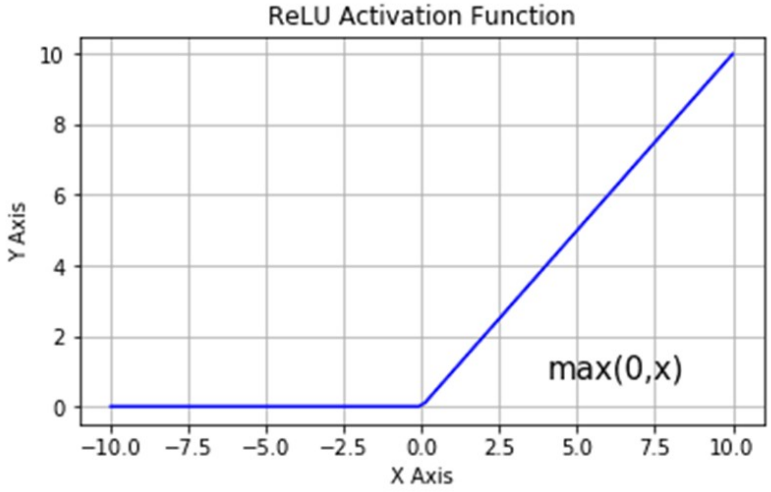

Fig 3.1.8 Relu function

3.2 Losses

Losses in Convolutional neural network : We want to reduce the error for each training example in supervised machine learning algorithms as we learn. Some optimization techniques, such gradient descent, are used for this. The loss function is also where this mistake originates. For a

single training sample, a loss function is used. An error function is another name for it. On the other hand, the average loss throughout the entire training dataset is a cost function. The cost function is to be minimised using the optimization procedures.

Binary Classification Loss Functions : Loss functions for binary classification The name pretty much speaks for itself. Putting an object into one of two classifications is known as binary classification. A rule is used to classify the input feature vector in this classification. For instance, binary classification involves determining whether or not an email is spam based on, let's say, its subject line. There are different loss function are Binary Cross Entropy Loss, Hinge Loss.

Multi-Class Classification Loss Functions : Predictive modelling issues involving more than two classes are known as multi-class classification. A common formulation of the issue is to predict an integer value, with each class being given a distinct integer value between 0 and (num classes – 1). Multi-Class Cross-Entropy Loss, Sparse Multiclass Cross-Entropy Loss ,Kullback Leibler Divergence Loss are the types of loss function.

3.3 Fully Connected Layer

Flatten Layer : The final step in a convolution neural network is flattening, which turns the output of the convolutional portion of the CNN into a 1D feature vector so you may give it as input to the dense layer.

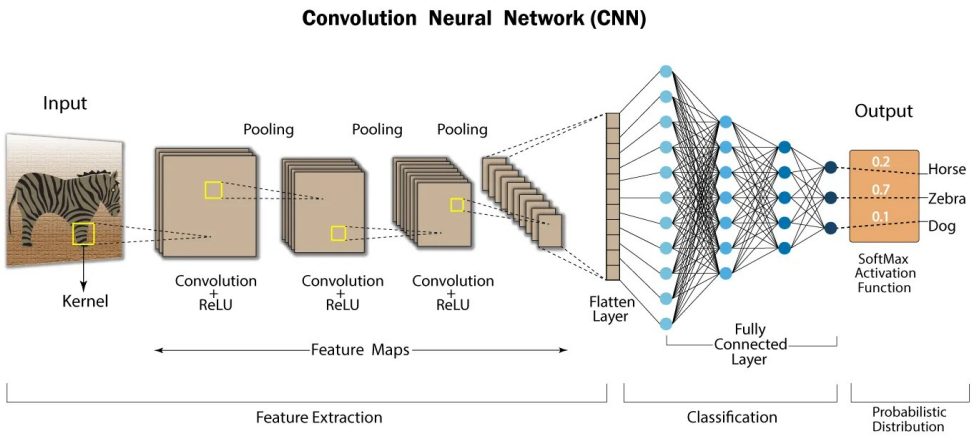

Fig 3.3 Flatter layer

3.4 Transfer Learning

LeNet 5 : Cnn performs less well It can be time-consuming and ineffective to select the ideal filters, layer count, and neurons. But later in 2012, Alex Krizhevsky and colleagues introduced AlexNet, an 8-layer deep neural network, as a replacement for CNN. After Alexnet it introduced new CNN Architectures or Transfer learning/pre-trained model.

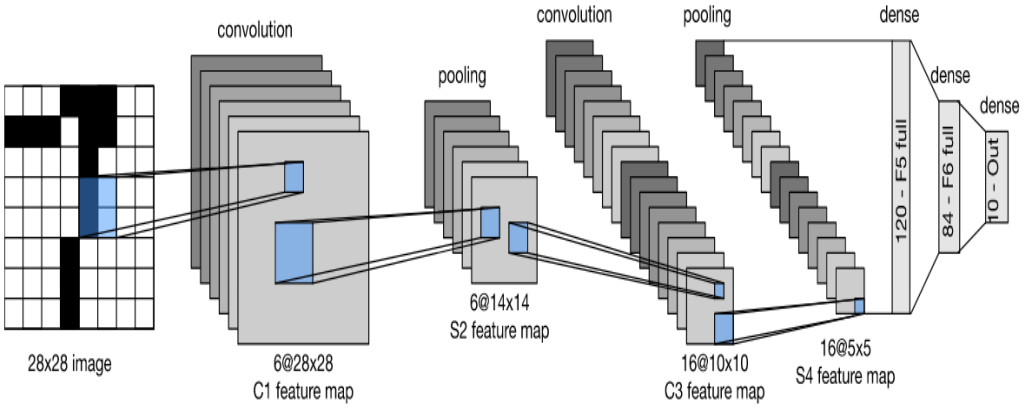

Fig 3.4.1 Lenet 5

Transfer learning/pre-trained model : Transfer learning, which focuses on storing and transferring knowledge learned while addressing one problem to another that is related, is the reuse of a pre-trained model on a new problem. Transfer learning is utilised for The most significant benefits of transfer learning over CNN include shorter training times, higher neural network performance, and a lower data requirement. There are numerous transfer learning models available, including VGG16, VGG19, and many others.

VGG16 : Visual Geometric Group is known as VGG. Because of the design's resemblance to AlexNet, VGG Net has a tonne of features as well. There are 138 million parameters in this network. There are total 16 convolutional layers in VGG 16.

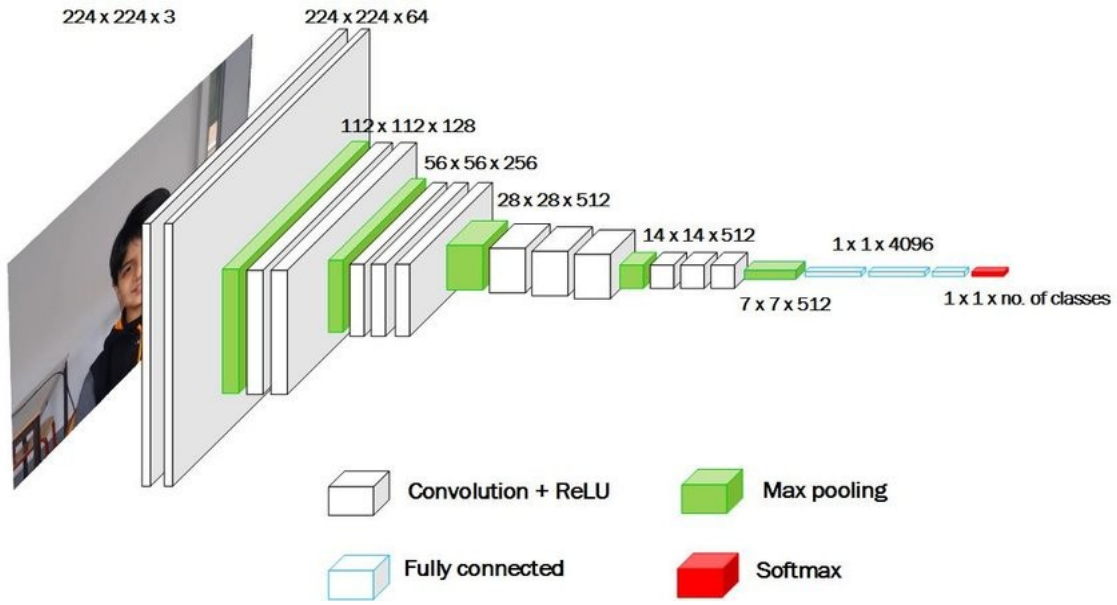

Fig 3.4.2 VGG16 -1

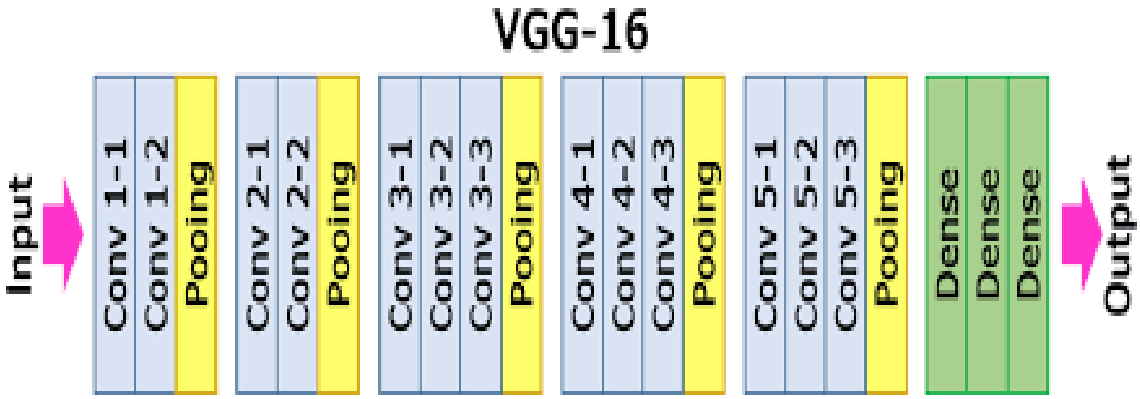

Fig 3.4.3VGG16-2

Chapter 4

Use Cases

Use of CNN

- Medical Image Computing CNN medical image classification system is more accurate than the human eye at detecting anomalies in X-ray and MRI pictures.

- Face recognition can be useful in identifying a person from surveillance footage or a hidden video recording.

- Facebook Messenger filters and Snap chat filters offer new components or effects to the face's auto-generated basic layout.

- CNN has been employed in automated vehicles to help them recognise obstacles and read roadway signs.

- By identifying things based on forms and patterns discovered inside an image, CNN has been applied to object recognition across images.

CONCLUSION

- The application of CNN as a tool for various Pattern Recognition Systems as well as OCR-based machine translation is a groundbreaking and dramatic idea in artificial neural networks (ANNs).

- Convolutional Neural Networks have established themselves as crucial tools for deep learning. Although it has drawbacks like a higher mistake rate and a higher training data requirement.

- It has been demonstrated that adding additional layers enhances performance but also adds more parameters with each layer, extends training time, and necessitates more processing power.

- Transfer learning or pre-trained models, which outperform CNN and come in a variety of models with varying numbers of layers and filters, are introduced. In general, these models use less input to achieve higher accuracy.

REFERENCES

[1] Arohan Ajit, Koustav Acharya, Abhishek Samanta, 'A Review of Convolutional Neural Networks', IEEE 2020.

[2] Divya Arora, Mehak Garg, Megha Gupta, 'Diving deep in Deep Convolutional Neural Network', IEEE 2020.

https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-networkcnn/

https://keras.io/api/applications/