ISSN: 2278-4632
Vol-10 Issue-12 No.03 December 2020

DEEP LEARNING ALGORITHM FOR HUMAN ACTIVITY RECOGNITION BY USING ALEX NET CNN

Pamisetty Chandana[1], N. Naveen Kumar[2], Dr. Y.L. Ajay Kumar[3]

[1]Assistant Professor, [2]Assistant Professor, [3]Associate Professor, ECE Department, Anantha Lakshmi Institute of Technology and Sciences, Ananthapuramu, Andhra Pradesh, India.

ABSTRACT*: The potential of machine learning, and especially deep learning, has become evident as research continues in applications such as the monitoring of the elderly and surveillance for the identification of criminals and items left in public places. While some techniques have been developed for Human Action Recognition (HAR) using wearable sensors, these devices can cause unnecessary mental and physical discomfort to people, especially children and the elderly. Deep learning has automation capabilities.*

Index Terms: ***Human Action Recognition (HAR), binary silhouettes, ALEXNET CNN.***

1. INTRODUCTION

Recognizing human behaviors in a real-world environment finds applications in a number of domains, including smart video monitoring, consumer characteristics, and shopping behavior analysis. However, correct identification of behavior is a very difficult activity due to cluttered contexts, occlusions, and combinations of perspectives, etc. Deep learning models are a class of machines that can learn a hierarchy of features by constructing high-level features from low- level ones, thus automating the process.

2. OVERVIEW OF HAR

The purpose of identification of human activity is to investigate the actions of video sequences or still images. Motivated by this fact, human activity recognition systems strive to correctly classify input data into their underlying activity category. Depending on their complexity, human activities are classified as: I gestures; (ii) atomic actions; (iii) human-to- object or human-to-human interactions; (iv) group actions; (v) behaviors; and (vi) events. Figure
(1) demonstrates the decomposition of the human actions.

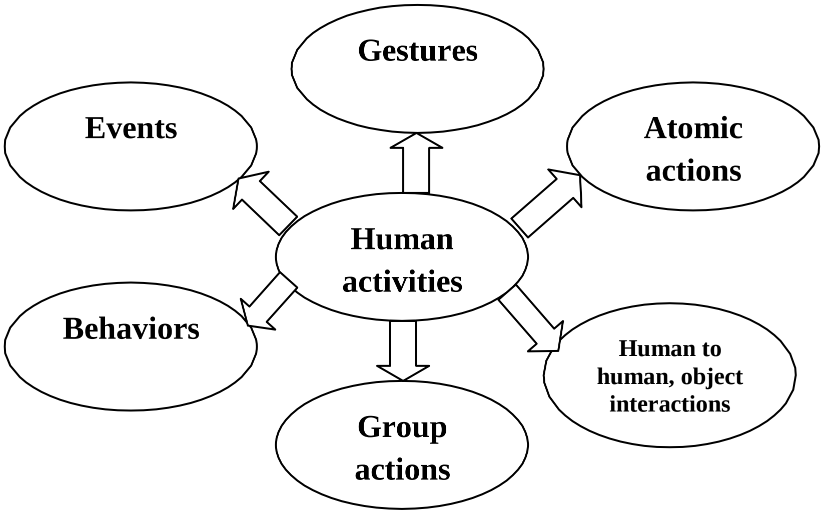


Figure: 1 Human Action Recognitions

3. PROPOSED SYSTEM

Deep Learning

Deep Learning is a sub-field of machine learning that deals with algorithms inspired by brain structure and function. In a word, deep learning accuracy achieves recognition accuracy at higher levels than ever before. This helps consumer electronics fulfill user standards and is important for safety-critical applications such as driverless cars. Recent advancements in deep learning have advanced to the point that deep learning outperforms humans in certain tasks, such as classifying objects in image.

How Deep Learning Works

Step1: The algorithm designer understands the problem and checks whether the deep learning is a good fit or not.

Step2: After understanding the problem he chooses relevant datasets and prepares them for analysis.

Step3: So, there are many deep learning algorithms are there, he chooses the best type of deep learning algorithm that suits to solve the problem.

Step4: Training an algorithm on large amount of labeled data.

Step5: After training he tests the model against the unlabeled data

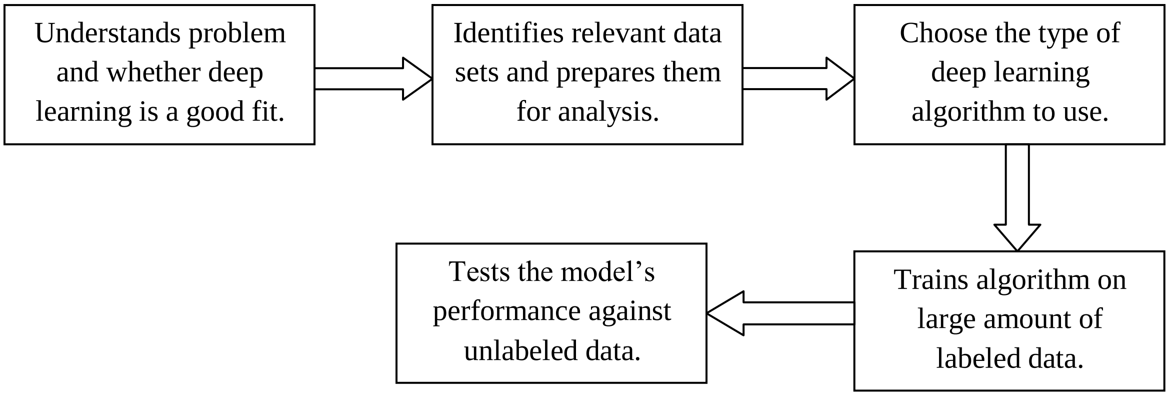

Understands problem and whether deep learning is a good fit.
Identifies relevant data sets and prepares them for analysis.
Choose the type of deep learning algorithm to use.
Trains algorithm on large amount of labeled data.
Tests the model's performance against unlabeled data.

Figure 2: Deep Learning Process

4. DEEP NEURAL NETWORKS

Most of the deep learning approaches use neural network architectures, which is why deep learning models are also referred to as deep neural networks. The word "deep" typically refers to the amount of hidden layers within the neural network. Traditional neural networks comprise only 2-3 hidden layers, whereas deep neural networks may have as many as 150 layers.Deep learning models are trained through the use of large sets of labeled data and neural networkarchitectures that explicitly learn features from data without the need for manual feature extraction.

Nodes are little parts of the system just like neurons in the human brain. Here some nodes are marked and connected but some are not. In general nodes are grouped into layers. The system must process layers of data between the input and output to solve the task. It has to process more layers to get the good result.

A deep neural network is much more complex than a neural network. It can recognize voice commands, recognize sound and graphics, conduct expert evaluations, and perform severalother acts that involve prediction, imaginative thinking, and research. Only the human brain has such possibilities and solves the problem more globally and can draw conclusions or assumptions based on the knowledge given and the desired outcome. It can solve a problem without a large number of marked details.

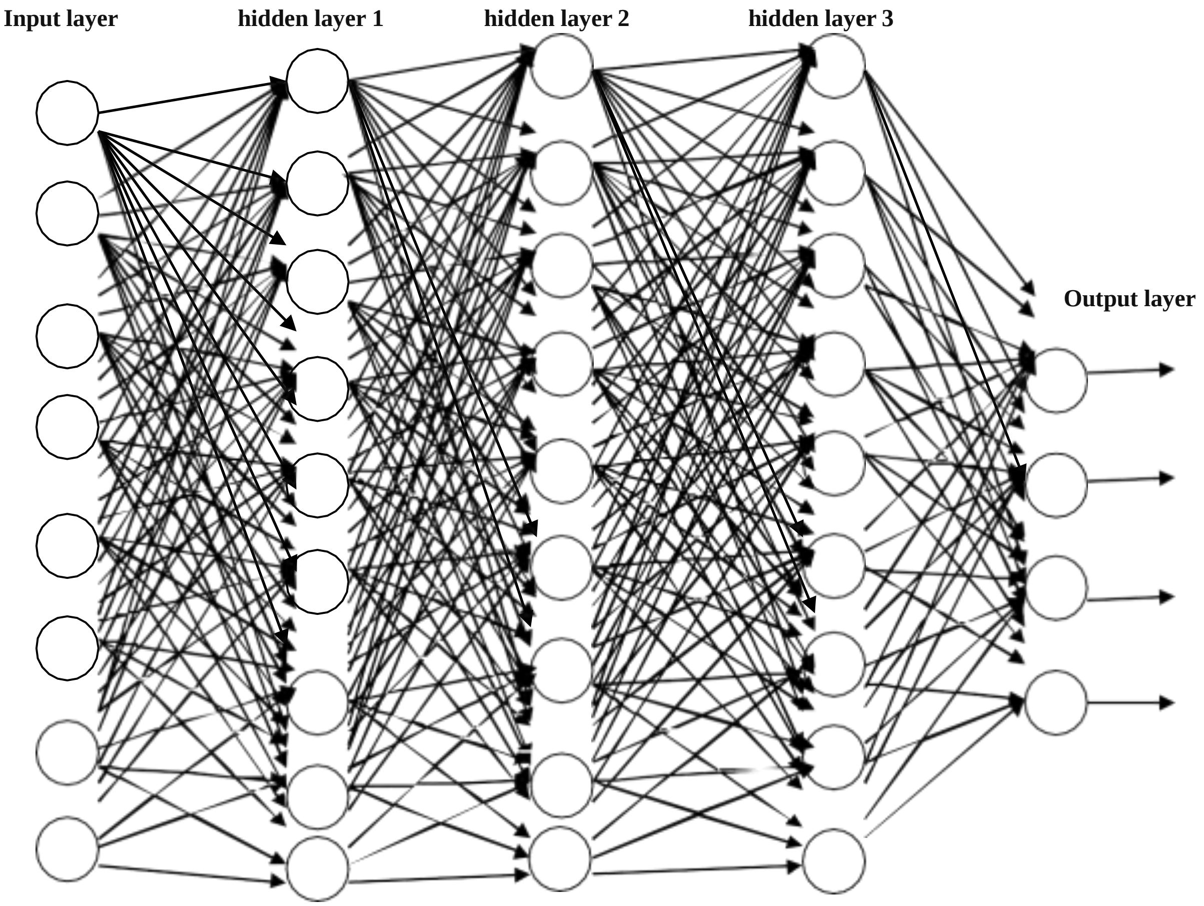


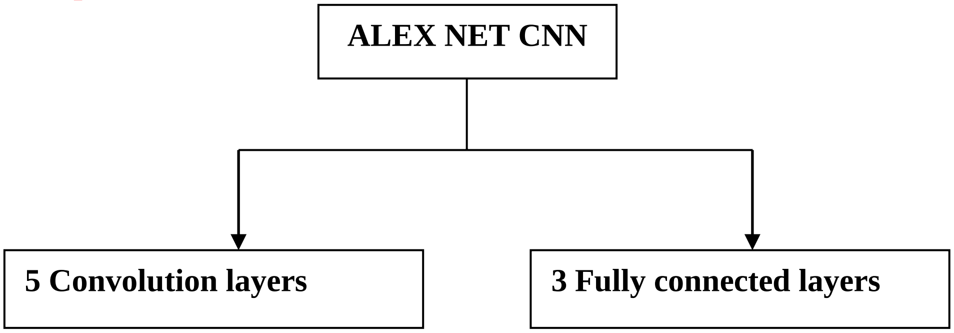
Figure 3: Deep Neural Network

5. ALEX NET CNN ARCHITECTURE

ALEX NET has 8 layers. The first five are convolutionary layers, and the last three are entirely interconnected layers. In between, we also have a few layers called Pooling and Activation. The architecture consists of predefined filters, strings, padding for good object detection. Alex Net is commonly used for object detection tasks. The size of the input picture

227*227*3

→ **Colored image**



ALEX NET CNN only performs the procedure when the input image has dimensions of 227*227*3. If not we need to reshape our input image. Here we send our input to the first convolution layer after we do the pooling, then the output of the pooling will be supplied to the second convolutionary layer and then again we do the pooling operation.

The second pooling output will be given as input to the third convolutionary layer, where we perform three steps of covolutionary operation after three steps of the third pooling operation. The output of the third pooling layer is given to the first fully connected layer. The third fully connected layer will acts as a softmax function that is used to predict the final output.

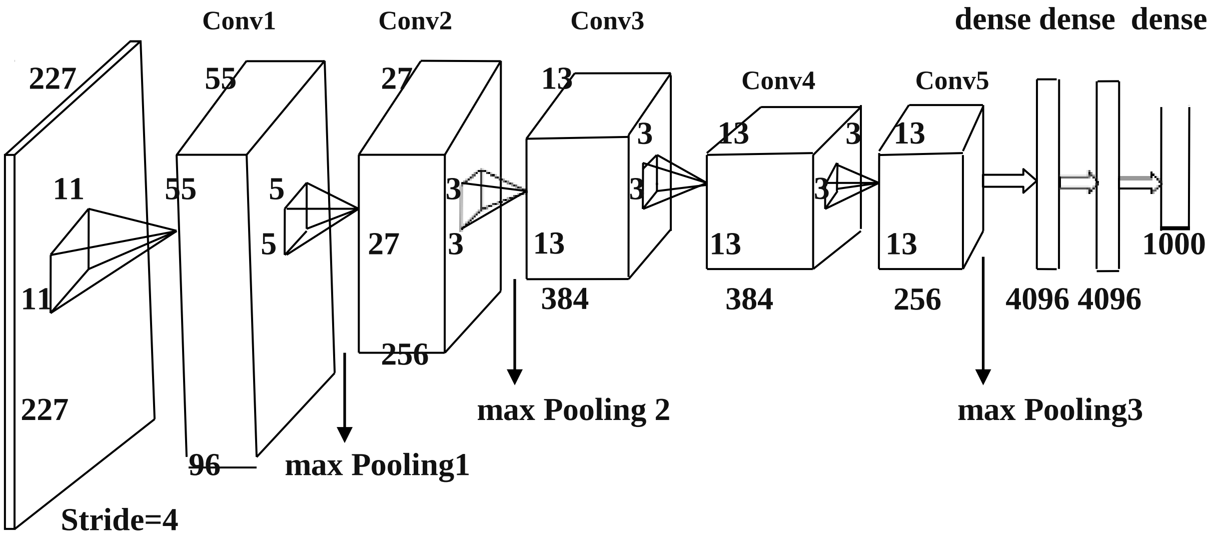


Figure 5: ALEX NET CNN Architecture

So, 227*227*3 is the input of the first convolution layer. 96 filters of size 11*11 with the 4-pixel phase will be added to the first convolution layer. We have a pooling layer after the first convolution layer where we use a window size of 3*3 with the 2 pixel phase. The output of the first convolution layer is given to the second convolution layer as the input.

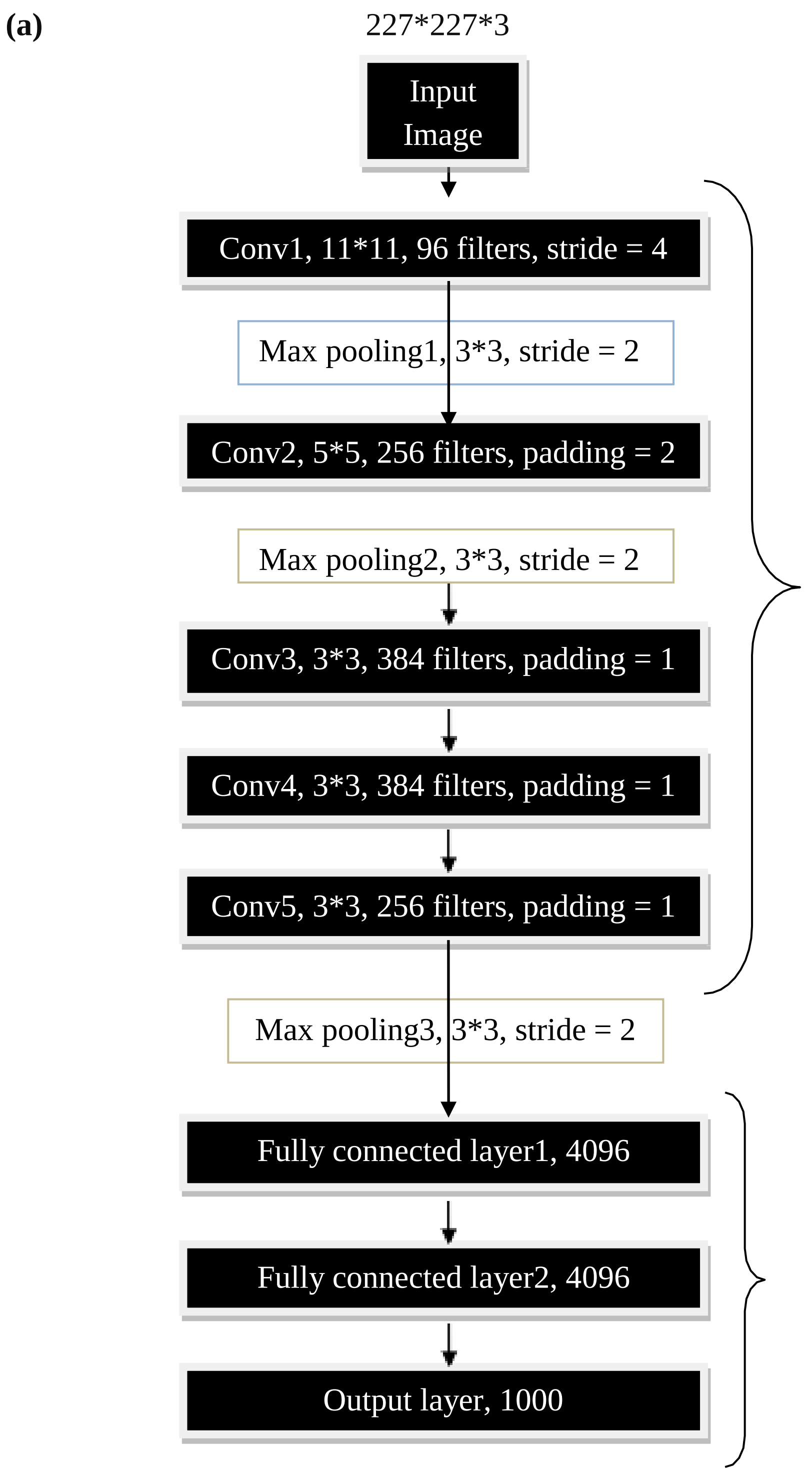

(a)
227*227*3
Input Image
Conv1, 11*11, 96 filters, stride = 4
Max pooling1, 3*3, stride = 2
Conv2, 5*5, 256 filters, padding = 2
Max pooling2, 3*3, stride = 2
Conv3, 3*3, 384 filters, padding = 1
Conv4, 3*3, 384 filters, padding = 1
Conv5, 3*3, 256 filters, padding = 1
Max pooling3, 3*3, stride = 2
Fully connected layer1, 4096
Fully connected layer2, 4096
Output layer, 1000

Figure 5)a): Alex net flow chart

With padding 2 and pooling layer 3*3 of step 2, we use 256 filters of size 5*5 in the second convolution layer. The output of the second convolution layer is given to the third

ISSN: 2278-4632
Vol-10 Issue-12 No.03 December 2020

convolution layer as the input. Three, four, five layers of convolution are related to each other without any layer of pooling between them. Third convolution layer of 384 scale 3*3 filters with padding 1. Fourth, the same. The third and fourth have the same characteristics. The scale of 256 filters in the fifth convolution layer.

We have a 3*3 size and phase 2 maxpooling after these three layers. After that, we have three completely linked layers, the last one is used for the activation function of softmax that generates a distribution over 1000 class labels. 4096 neurons in the first fully connected layer, 4096 in the second fully connected layer, and 1000 neurons in the last fully connected layer. The neurons in the last fully connected layer rely on the dataset,

Layer		**Feature map**	**Size**	**Kernel size**	**Stride**	**Activation**
Input	Image	1	227*227*3	-	-	-
1	Convolution1	96	55*55*96	11*11	4	Relu
	Maxpooling1	96	27*27*96	3*3	2	Relu
2	Convolution2	256	27*27*256	5*5	1	Relu
	Maxpooling2	256	13*13*256	3*3	2	Relu
3	Convolution3	384	13*13*384	3*3	1	Relu
4	Convolution4	384	13*13*384	3*3	1	Relu
5	Convolution5	256	13*13*256	3*3	1	Relu
	Maxpooling3	256	6*6*256	3*3	2	Relu
6	FC	-	9216	-	-	Relu
7	FC	-	4096	-	-	Relu
8	FC	-	4096	-	-	Relu
9	FC	-	1000	-	-	Relu
						Soft-max

Table 1: Parameters used in Alexnet Cnn

Calculation of layers:

Without padding

$$\frac{n-f}{s}+1*\frac{n-f}{s}+1$$

With padding

$$\frac{n+2p-f}{s}+1*\frac{n+2p-f}{s}+1$$

Where, n = Image size

f = Filter size

s = Stride

p = padding

Layer 1: Convolution1

$$\frac{n-f}{s}+1*\frac{n-f}{s}+1$$

$$\frac{227-11}{4}+1*\frac{227-11}{4}+1$$

55*55*96

Max pooling1

$$\frac{n-f}{s}+1*\frac{n-f}{s}+1$$

$$\frac{55-3}{2}+1*\frac{55-3}{2}+1$$

27*27*96

Layer 2: Convolution2

$$\frac{n+2p-f}{s}+1*\frac{n+2p-f}{s}+1$$

$$\frac{27+2(2)-5}{1}+1*\frac{27+2(2)-5}{1}+1$$

27*27*256

Max pooling2

$$\frac{n-f}{s}+1*\frac{n-f}{s}+1$$

$$\frac{27-3}{2}+1*\frac{27-3}{2}+1$$

13*13*256

Layer 3&4: Convolution 3&4

$$\frac{n+2p-f}{s}+1*\frac{n+2p-f}{s}+1$$

$$\frac{13+2(1)-3}{1}+1*\frac{13+2(1)-3}{1}$$

Layer 5: Convolution5

$$\frac{n+2p-f}{s}+1*\frac{n+2p-f}{s}+1$$

$$\frac{13+2(1)-3}{1}+1*\frac{13+2(1)-3}{1}+1$$

Layer 3&4: Convolution 3&4

$$\frac{n+2p-f}{s}+1 * \frac{n+2p-f}{s}+1$$

$$\frac{13+2(1)-3}{1}+1 * \frac{13+2(1)-3}{1}$$

13*13*384

Max pooling3

$$\frac{n-f}{s}+1 * \frac{n-f}{s}+1$$

$$\frac{13-3}{2}+1 * \frac{13-3}{2}+1$$

6*6*256

Layer 5: Convolution5

$$\frac{n+2p-f}{s}+1 * \frac{n+2p-f}{s}+1$$

$$\frac{13+2(1)-3}{1}+1 * \frac{13+2(1)-3}{1}+1$$

13*13*256

Fully connected layer1

$$6 * 6 * 256$$

9216

Convolutional layer:

To construct a feature map that summarizes the presence of detected features in the data, it applies a filter to an input. One image becomes a stack of filtered images in the convolutional layer, and the number of filtered images depends on the number of filters.

Input image * Filter = Filtered image

0	**1**	**2**
3	**4**	**5**
6	**7**	**8**

0	**1**
2	**3**

19	**25**
37	**43**

Pooling layer:

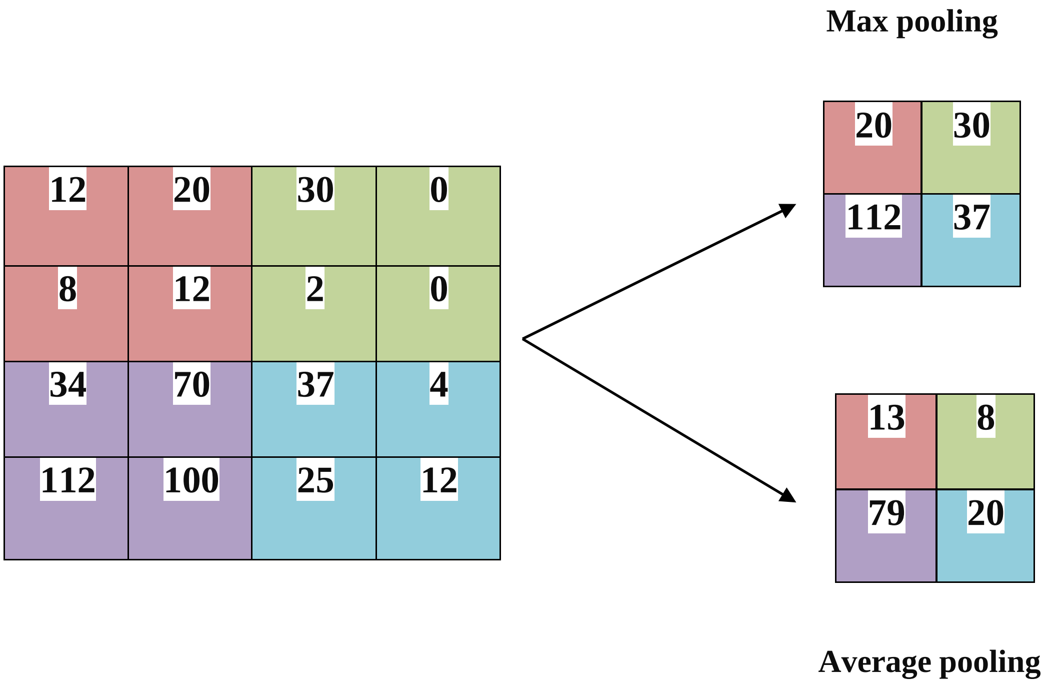
Pooling layer down samples the volume spatially, independently in each depth slice of the input volume. The most common down sampling operation is max, giving rise to max Pooling.

Max pooling with 2*2 filter and stride4

There are two types of pooling. They are:

1. Max pooling: From above example in a 2*2 window we choose max value.The process called max pooling.

2. Average pooling: From above example we take the average of 2*2 window. The process called average pooling.



12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

Max pooling

20	30
112	37

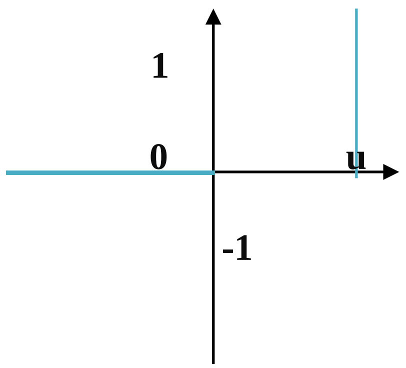
13	8
79	20

Average pooling

Rectified Linear unit（ReLU):

- ❖ Activation function of ReLU produces 0 when u < 0, and is linear with slope 1 when u > 0. Rectified linear function, f(u) = max(0,u)

f(u) = max (0,u)

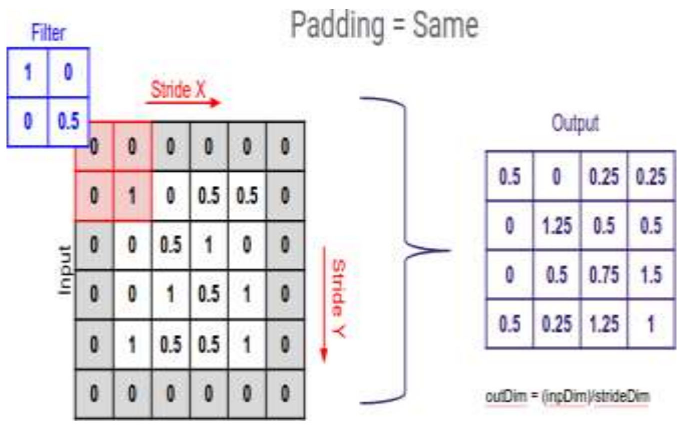


Fully connected layer:

- ❖ This is the layer where image classification actually happens and we convert our filtered images into a 1-Dimensional array.

Padding and Stride:

- Adding zero rows and columns to the image is known as padding.
- Number of columns and rows are shifting towards right and downside is known as stride.



1	0
0	0.5

0	0	0	0	0	0
0	1	0	0.5	0.5	0
0	0	0.5	1	0	0
0	0	1	0.5	1	0
0	1	0.5	0.5	1	0
0	0	0	0	0	0

0.5	0	0.25	0.25
0	1.25	0.5	0.5
0	0.5	0.75	1.5
0.5	0.25	1.25	1

Soft-max function:

The soft-max function is applied after the output layer of ALEX NET CNN in order to obtain the probability of the possible actions.

$$\boldsymbol{\sigma}\,(\mathbf{z})_{\mathbf{j}} = \mathbf{e}^{\mathbf{z}}{}_{\mathbf{j}} / \;\; {}^{\boldsymbol{N}}_{\boldsymbol{k}=\mathbf{1}}\;\; \mathbf{e}^{-\mathbf{z}}\,\mathbf{k}$$

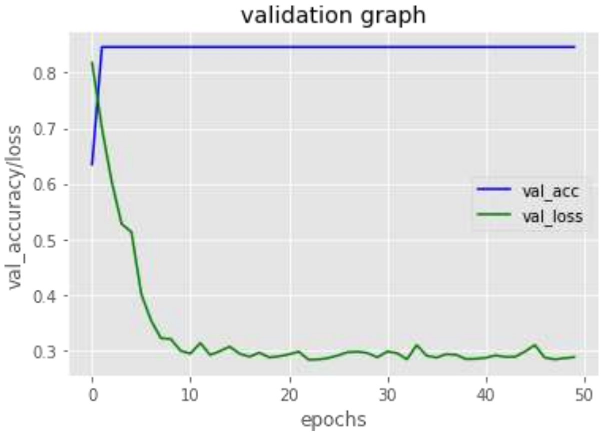
Where, j = each action

Z = network output

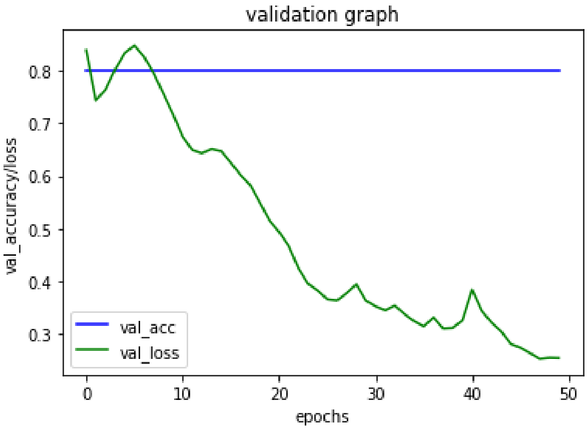
N = Total number of actions

6. SIMULATION RESULTS AND ANALYSIS

In general, the datasets used for Convolutional Neural Networks has variations in the images which yield higher accuracy levels (0.7 – 0.95). But, we have selected a dataset with higher levels of similarities in the images which possess a problem to achieve higher accuracy levels (0.3 – 0.55). The modifications are made in the algorithm such that we can achieve higher accuracy (0.84) and some preprocessing modifications has done like image resizing and image cropping such that the Region of Interest (ROI) is highlighted.



Dataset results



Real time results

Due to false positives identified in the algorithm false higher accuracy is achieved till 1 to 12 epochs then as the number of epochs increases false positives are minimized and true positives are identified.

7. CONCLUSION

We propose an intelligent human action recognition system to develop as a consumer electronics product (with low computational cost and high accuracy outcomes) for automatically monitoring and recognizing the daily activities of elderly people living alone. Moreover, this system can be utilized without any restrictions on environmental conditions or domain structures, and is also very promising for real-time applications because of the fast-processing time. The problems of view-variation (single camera) and intra-class variation have been solved in this system. The experimental results show that the proposed system is outperforming other state-of-the-art methods both on CAD-60 daily activity datasets. An intelligent human action recognition system is developed in consumer electronics product with low computational cost and high accuracy outcomes.

8. REFERENCES

[1] Cho Nilar Phyo, T. T. Zin and P. Tin, “Deep Learning for Recognizing Human Activities using Motions of Skeletal joints” DOI 10.1109/TCE.2020, IEEE transactions on consumer electronics.

[2]C. N. Phyo, T. T. Zin and P. Tin, “Skeleton motion history based human action recognition using deep learning”, in Proc. of 2017 IEEE 6th Global Conf. on Consumer Electronics, Nagoya, Japan, 24-27 Oct. 2017, pp. 784-785.

[3]J. Wang et al., “An enhanced fall detection system for elderly person monitoring using consumer home networks”, IEEE Trans. Consumer Electronics, vol. 60, no. 1,pp.23-29,Apr.2014, 10.1109/TCE.2014.6780921.

[4] A. Jalal et al., “Depth video-based human activity recognition system using translation and scaling invariant features for life logging at smart home”, IEEE Trans. Consumer Electronics, vol. 58, no. 3, pp. 863-871, Sept. 2012, 10.1109/TCE.2012.6311329.

[5] T. T. Zin, P. Tin and H. Hama, “Visual monitoring system for elderly people daily living activity analysis”, in Proc. of the Int. MultiConf. of Engineers and Computer Scientists 2017, Hong Kong, 15-17 Mar. 2017, pp. 140-142.

[6] L. Zaineb et al., “A Markovian-based approach for daily living activities recognition”, in Proc. of the 5th Int. Conf. on Sensor Networks, Rome, Italy, 17-19 Feb. 2016, pp. 214-219.

[7] L. H. Wang et al., “An outdoor intelligent healthcare monitoring device for the elderly”, IEEE Trans. Consumer Electronics, vol. 62, no. 2, pp. 128-135, Jul. 2016, 10.1109/TCE.2016.7514671.

[8] J. Wang et al., “An enhanced fall detection system for elderly person monitoring using consumer home networks”, IEEE Trans. Consumer Electronics, vol. 60, no. 1, pp. 23-29, Apr. 2014, 10.1109/TCE.2014.6780921.