International Journal of Engineering Research and Applications *www.ijera.com*
ISSN: 2248-9622, pp. 14-18

RESEARCH ARTICLE OPEN ACCESS

Review Paper on Updation Score Board

Bharath Kumar E R[1], Abhilash T S[2], R Kodhandaram[3],
Jonnadula Shalini[4], Aruna R[5]
[1234]Student, [5]Associate Professor, Department of Electronics and communication Engineering, AMC Engineering college, Bangalore, India.

ABSTRACT
We are residing in a cutting edge reality where advancements are being developed and the current innovations are being refined to work on the precision and furthermore facilitate the work by people. In cricket, additionally various advances have been as of now executed like the Hawk eye innovation, the Hotspot innovation, the Snicko, etc. But the scoreboard of a cricket match is still being calculated manually. We expect to make a framework in which the scoreboard is determined naturally by perceiving the motions that are announced the umpires. In this work, we introduce a new dataset. The results are obtained using a Convolution neural network and Deep learning.
Keywords-*Deep learning, Convolution neural network*

I. Pre-processing

1. CNN Algorithm overview:

Convolutional Neural Network were used to achieve some state of the art results and win outstanding difficulties. The use of convolutional layers comprises in convolving a sign or a picture with portions to get highlight maps... The loads of the bits are adjusted during the preparation stage by back spread, to upgrade specific qualities of the input. Since the parts are divided between all units of a similar component maps, convolutional layers have less loads to prepare than thick FC layers, making CNN simpler to prepare and less inclined to overfitting.

2. Network Architecture:

Convolutional Layer:

Input contentions for this layer are separating size, the quantity of filters, and cushioning. Here, the filter of size 10 is utilized, which decides 10 x 10 filter. The quantity of filters utilized is 10, implies 10 neurons are associated.

3. Architecture (AlexNet):

This engineering was quite possibly the earliest profound organization to move ImageNet Classification exactness by a critical step in contrast with customary systems. It is made from 5 convolutional layers followed by 3 totally related layers, as depicted in Figure. 3.1.

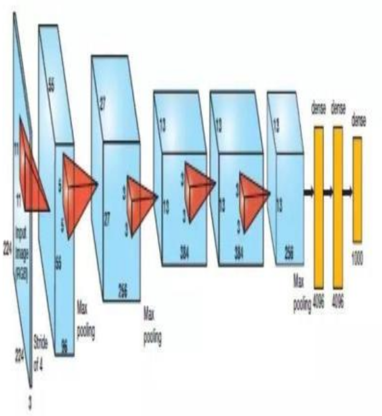

Fig3.1: Architecture of Neural network (AlexNet)

AlexNet, proposed by Alex Krizhevsky, uses ReLu(Rectified Linear Unit) for the non-linear part, instead of a Tanh or Sigmoid function which was the earlier standard for traditional neural networks. ReLu is given by: $f(x) = \max(0,x)$

The benefit of the ReLu over sigmoid is that it prepares a lot quicker than the last in light of the fact that the subsidiary of sigmoid turns out to be tiny in the immersing district and consequently the updates to the loads nearly disappear. This is called vanishing gradient problem.

Drop Out work:

The thought behind the dropout is like the model outfits. Due to the dropout layer, different courses of action of neurons which are switched off, address a substitute designing and this huge number of different models are ready in agreed with weight given to each subset and the summation of burdens being one. For n neurons connected to DropOut, the quantity of subset designs shaped is 2^n. So, it adds up to expectation being arrived at the midpoint of over these outfits of models.

Algorithm works Steps:

Step1: Convolutional Neural Networks:

Convolutional Neural Networks have a different architecture than regular Neural Networks. Ordinary Neural Networks change a contribution by putting it through a progression of stowed away layers. Each layer is comprised of a bunch of neurons, where each layer is completely associated with all neurons in the layer previously. Finally, there is a last fully-connected layer — the output layer — that represent the predictions.

CNN is composed of two major parts:

- **Feature-Extraction:**

In this part, the association will play out a movement of convolutions and pooling tasks during which the components are recognized. On the off chance that you had an image of a zebra, here the organization would perceive its stripes, two ears, and four legs.

- **Classification:**

Here, the completely associated layers will act as a classifier on top of these removed highlights. They will give out a probability for the article on the image being what the calculation predict

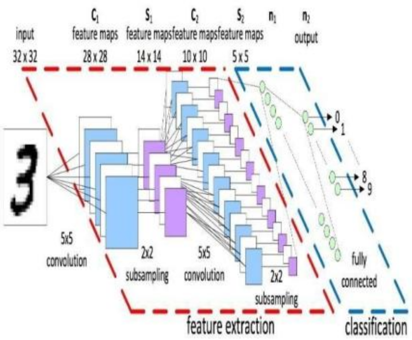

Fig3.2: Convolutional Neural Networks architecture

There are squares and lines inside the red spotted locale which we will separate it later. The green circles inside the blue spotted district named grouping is the neural network or multi-layer perceptron which acts as a classifier. The contributions to this organization come from the previous part named highlight extraction.

Highlight extraction is the piece of CNN arctiecture from where this organization determines its name.

Convolution is the numerical activity which is integral to the viability of this calculation..

Step2: Feature Extraction: Convolution:

Convolution in CNN is performed on an info picture using a filter or a section. To comprehend sifting and convolution you should filter the screen beginning from upper left to right and dropping down a cycle subsequent to covering the width of the screen and rehashing a similar interaction until you are finished checking the entire screen.

For example on the off chance that the information picture and the filter seem to be following As shown in fig 3.3

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Input

1	0	1
0	1	0
1	0	1

Filter / Kerne

Fig3.3: Input image and the filter

The filter(green) slides over the info (blue) each pixel in turn beginning from the up

Fig3.3: Filter (green) slides over the input image

In the above Fig3.3 the worth 4 (upper left) in the result framework (red) relates to the filter cross-over on the upper left of the picture which is processed as;

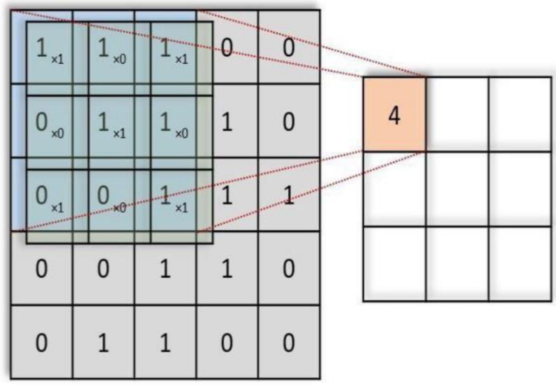

Fig3.4: 1st step of convolution

$$(1\times1+0\times1+1\times1)+(0\times0+1\times1+1\times0)+(1\times0+0\times0+1\times1)=4$$

Also, we process different upsides of the result network. Note that the upper left esteem, which is 4, in the result grid relies just upon the 9 qualities (3x3) on the upper left of the first picture framework. It doesn't change regardless of whether the other qualities in the picture change. This is the open field of this outcome worth or neuron in our CNN. Every value in our outcome network is sensitive to simply a particular district in our extraordinary picture as in fig:3.4.

Because of pictures with different channels (for instance RGB), the Kernel has the very significance as that of the information picture. Network Multiplication is performed among □□ and □□ stack ([□1,1],[□2,□2],[□3,□3]) and every one of the outcomes are added with the inclination to give us a crushed one-profundity channel Convoluted Feature Output as shown in fig:3.5.

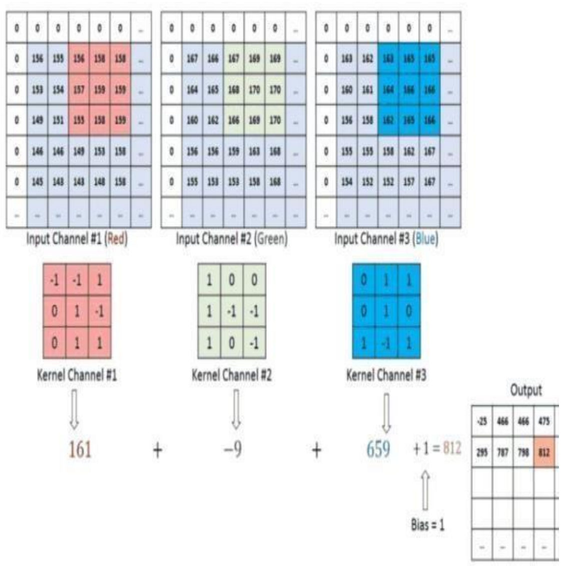

Fig3.5: Squashed one-depth channel convolution feature

Every neuron in the result framework has covering responsive fields. The Fig3.6 beneath will provide you with a superior feeling of what occurs in convolution.

Routinely, the primary ConvLayer is liable for catching the Low-Level elements like edges, variety, angle direction, and so on.

With added layers, the engineering adjusts to the High-Level elements too, giving us an organization which has the healthy understanding of pictures in the informational index, like how we would.

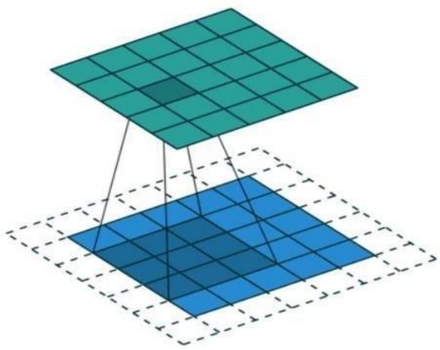

Fig3.6: Convolution example

Step3: Feature Extraction: padding:

There are two sorts of results to the activity one in which the tangled element is decreased in dimensionality when contrasted with the input, and the other in which the dimensionality is either expanded or continues as before.

In our model when we expand the 5x5x1 picture into a 7x7x1 picture and afterward apply the 3x3x1 part ready to be done, we find that the tangled grid ends up being of aspects 5x5x1. It suggests our outcome picture is with same viewpoints as our outcome picture (Same Padding).

On the other hand, if we perform the same operation without padding, in the output we'll receive an image with reduced dimensions. So our (5x5x1) image will become (3x3x1). **Feature Extraction: example:**

Lets say we have a physically composed digit picture like the one underneath. We want to eliminate out only the level edges or lines from the image.We will utilize a filter or piece which when tangled with the first picture diminishes out that multitude of regions which don't have level edges:

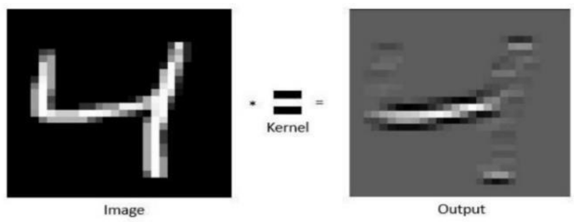

Fig3.7: Horizontal filter example

Notice how the outcome picture simply has the level white line and rest of the image is darkened. The piece here looks like a peephole which is a level cut. .Similarly for a vertical edge extractor the filter is like a vertical slit peephole and the output would look like shown in fig:3.8

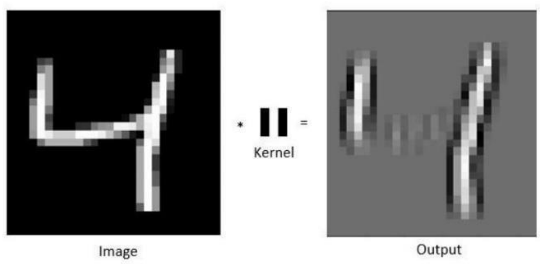

Fig3.8: Vertical filter example

Step4: Feature Extraction: Non- Linearity:

In the wake of sliding our filter over the first picture the result which we get is edge extractor and got two output images. gone through another numerical capacity which is called an actuation work. The enactment work typically utilized generally speaking in CNN highlight extraction is ReLu which represents Rectified Linear Unit. Which just believers each of the negative qualities to 0 and keeps the positive qualities the equivalent:

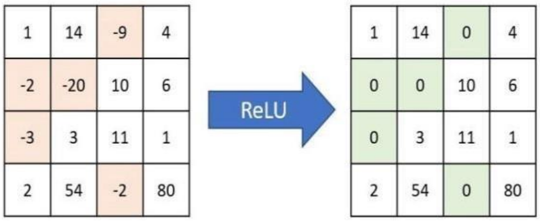

Fig3.9: CNN feature extraction with ReLu

After passing the outputs through ReLu functions they look like:

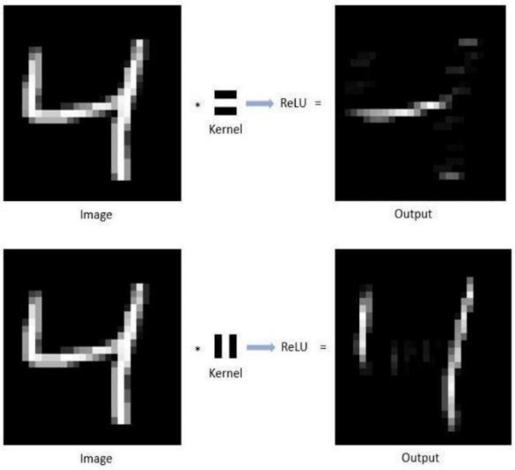

Fig3.9: Input image after filters with ReLu example
So for a single picture by convolving it with various channels we can obtain different outcome pictures.For the

handwritten digit here we applied a horizontal edge extractor and a vertical We can apply several other filters to generate more such outputs images which are also referred as feature maps.

Step5: Feature Extraction: Pooling:
After a convolution layer once you get the element maps, it is normal to add a pooling or a sub-examining layer in CNN layers. Like the Convolutional Layer, the Pooling layer is answerable for lessening the spatial size of the Convolved Feature. This is to reduce the computational power expected to manage the data through dimensionality decline. Moreover, it is valuable for extracting predominant aspects which are rotational and positional invariant, consequently keeping up with the course of successfully preparing of the model. Pooling abbreviates the preparation time and powers over- fitting.

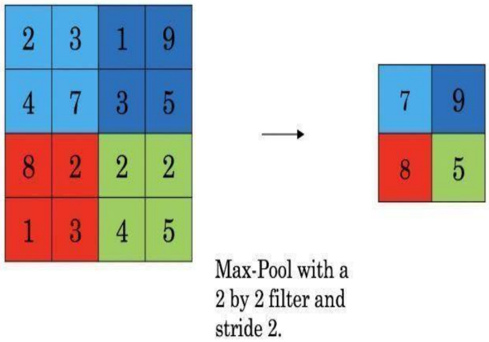

Fig3.10: Max Pooling example

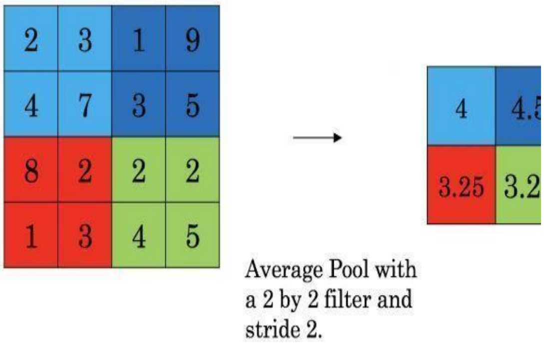

Fig3.11: Average Pooling example

Step6: Classification — Fully Connected Layer (FC Layer):
Adding a Fully-Connected layer is a (regularly) unassuming way to deal with learning nondirect mixes of the incredible level components as tended to by the consequence of the convolutional layer.The Fully-Connected layer is learning a conceivably non-direct capacity there. Illustration of CNN organization:

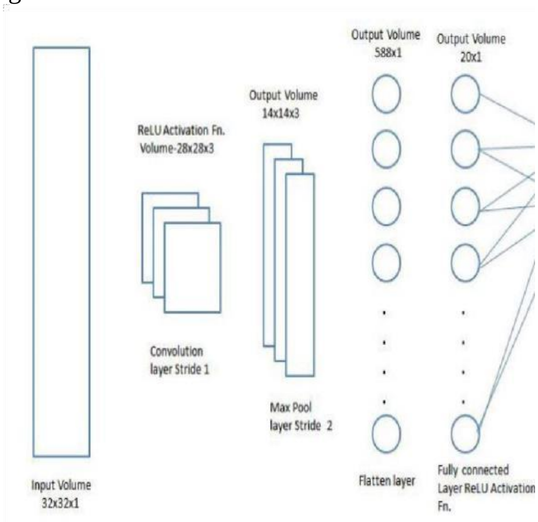

Fig3.12: Fully Connected model

Now that we have converted our input image into a suitable form, we shall flatten the image into a column
vector. The flattened output is fed to a feed-forward neural network and backpropagation applied to every iteration of training. Over a progression of ages, the model can recognize ruling and certain low-level features in pictures and group them utilizing the Softmax Classification method.

II. Conclusion:

We are encouraging an upgraded cricket score assumption system that perceives the badge of the umpires and updates the scoreboard successfully the elements of a game give the overview of critical events of that game like an objective in soccer or a wicket in cricket. a structure is made to distinguish the striking signs and positions shown by the umpire to normally create cricket highlights.

References:

[1]. Md. Asif Shahjalal, Zubaer Ahmad, “An Approach to Automate the Scorecard in Cricket with Computer Vision and Machine Learning”,2017 3 rd International Conference on electrical information and communication technology.

[2]. M. Komar, P. Yakobchuk, V. Golovko, V. Dorosh and Sachenko, "Deep Neural Network for Image Recognition Based on the Caffe Framework," 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), Lviv, 2018, pp. 102-106. Jürgen Schmidhuber, “Deep learning in neural networks: An overview,” Neural Networks, vol. 61, pp. 85-117, January 2015.

[3]. S. Mitra and T. Acharya, “Gesture recognition: A survey,” IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 37, no. 3, pp. 311–324, 2007.

[4]. “Top 10 list of the internet world‟s most popular sports,” http://www.topendsports.com/world/lists/popular - sport/fans.htm. https://www.python.or