Alzheimer's Tab: A Way to Strengthen Memory using Neural Networks

N.D. Karande[1], S.S. More[2] and P. S. Khot[3]

[1]N. D. Karande, Associate Professor, CSE, SGI, Kolhapur, Maharashtra, India; nikhilkarande18@gmail.com[1]

[2]S. S. More, Assistant Professor, CSE, SGI, Kolhapur, Maharashtra, India; more.ss@sginstitute.in[2]

[3]P. S. Khot, Assistant Professor, CSE, SGI, Kolhapur, Maharashtra, India; khot.ps@sginstitute.in[3]

ABSTRACT

Recently deep learning has been used in many fields such as image classification, text detection and recognition, object detection etc. As concerned to image processing, object detection is becoming one of the popular fields in research. When talking about computer vision tasks the powerful ability with feature learning and transfer learning of CNN has proved to be a major breakout. Due to this the convolution neural network is common interest among all deep learning community. CNN is typically feed-forward architecture and hence we propose a recurrent CNN (RCNN) for object detection. In this paper we build a simple CNN to apply object detection and achieve our goal. Firstly, the paper includes basic concept and architecture of CNN. After building a successful RCNN based object detection model using TensorFlow and testing it or object recognition dataset: MS-COCO, Flickr-8K, Flickr-30K. Querying the identified objects and locating the misplaced objects. Finally it solves the problem of Alzheimer, Dementia, Amnesia patients.

Key Words: Convolutional Neural Network, Object detection, Deep learning, TensorFlow

1. INTRODUCTION

Forgetfulness is a general ailment with most people these days, and it's only on the grow with progressive memory-related ailments such as dementia, Alzheimer's disease, amnesia, etc. due to which people tend to repeatedly be unable to find objects they use throughout their day-to-day activities, also affecting their every day life. ALZHEIMER'S TAB tracks all objects around, provided that a simple-yet-efficient method to solve a problem that's so very widespread in today's world. It's a wearable device positioned over the clothing. Capturing images as user moves around processes them over a deep-learning-framework to take out the objects within them. Deep learning is the new huge trend in machine learning. Deep learning refers to a class of artificial neural networks (ANNs) composed of many processing layers. According to Dr. Robert Hecht-Nielsen an Artificial Neural network (ANN) can be defined as "A computing system made of number of undemanding, highly organized processing elements, which process information by their dynamic state reply to external inputs". Artificial neural network have been applied to a wide range of fields such as medicine, finance, data recognition, geology, physics. Artificial neural network detect patterns too complex to be recognized by humans. ALZHEIMER'S TAB is a wearable device which uses a cloud-based deep-learning structure.

2. CONVOLUTIONAL NEURAL NETWORK

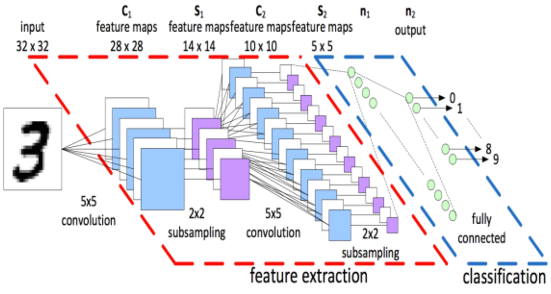

Figure 1. Architecture of Convolutional Network

Convolutional Neural Networks (CNNs) have wide applications in image and video recognition, recommender systems and natural language processing. CNNs, like neural networks, are made up of neurons with learnable weights and biases. every neuron receives numerous inputs, takes a weighted sum over them, pass it throughout an activation function and responds with an output. Convolutional neural network layer types mainly include three types, namely Convolutional layer, pooling layer and fully-connected layer.

a. Convolutional Layer:

Convolution is a arithmetical operation that's used in single processing to filter signals, find patterns in signals etc. In a convolutional layer, all neurons relate convolution operation to the inputs; hence they are call convolutional

 DOI : 10.18231/2454-9150.2018.0242 © 2018, IJREAM All Rights Reserved.

neurons. The most important factor in a convolutional neuron is the filter size; we have a layer with filter size 5*5*3. The input that's feed to convolutional neuron is an input image of size of 32*32 with 3 channels.

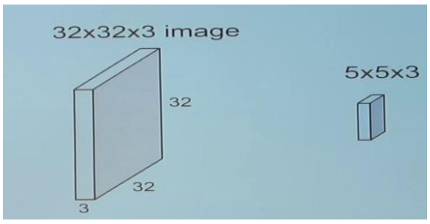

Figure 2: Convolving an image with a filter

We select one 5*5*3(3 for number of channels in a colored image) sized chunk from image and compute convolution (dot product) with our filter (w). This one convolution function will result in a single number as output. We shall also add the bias (b) to this output.

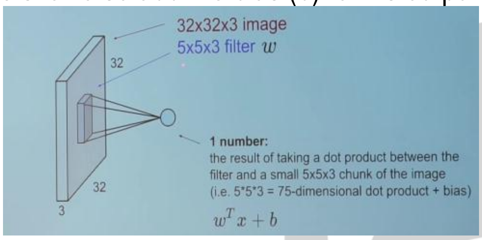

Figure 3: Sliding the filter layer over the complete image

In order to compute the dot product, it's compulsory for the 3rd dimension of the filter to be same as the number of channels in the input. i.e. when we compute the dot product it's a matrix multiplication of 5*5*3 sized chunk with 5*5*3 sized filter. If you concatenate all these outputs in 2D, we shall have an output *activation map* of size 28*28. Normally, we use more than 1 filter in one convolution layer. If we have 6 filters in our case in point, we shall have an output of size 28*28*6.

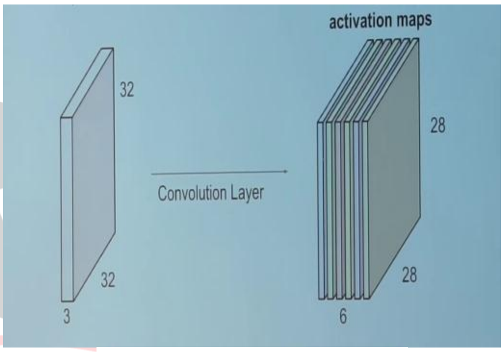

Figure 4: Final Convolution

As you can observe, after each convolution, the output reduces in size (as in this case we are going from 32*32 to 28*28). In a deep neural network with several layers, the output will suit very small this way, which doesn't work very well. So, it's a standard practice to add zeros on the boundary of the input layer such that the output is the similar size as input layer. So, in this case in point, if we add a padding of size 2 on both sides of the input layer, the size of the output layer will be 32*32*6 which works immense from the implementation point as well. Let's say you have an input of size N*N, filter size is F, you are using S as stride and input is added with 0 pad of size P. Then, the output size will be: (N-F+2P)/S +1

b. Pooling Layer:

Pooling layer is typically used immediately following the convolutional layer to decrease the spatial size (only width and height, not depth). This reduces the number of parameters, hence computation is reduced. Also, with a reduction of number of parameters avoid over fitting. The most general form of pooling is Max pooling where we take a filter of size F*F and apply the maximum operation over the F*F sized part of the image.

If you take the average in place of taking maximum, it will be call average pooling, but it's not especially popular. If your input is of size w1*h1*d1 and the size of the filter is f*f with stride S. Then the output sizes w2*h2*d2 will be:

w2= (w1-f)/S +1
h2= (h1-f)/S +1
d2=d1

Most ordinary pooling is done with the filter of size 2*2 with a stride of 2. As you can compute using the above procedure, it essentially reduces the size of input by half.

c. Fully Connected Layer:

If every neuron in a layer receives input from all the neurons in the earlier layer, then this layer is called fully connected layer. The output of this layer is computed by matrix multiplication followed by bias offset.

3. OBJECT DETECTION BASED ON RCNN

CNN object detectors have a theory of anchor boxes. This is a set of prototype shapes that is fixed throughout training relatively than learned. The learned behavior is to dynamically according to the particular image recommend offsets that transform these anchor boxes into detections (rectangular regions hopefully containing an object). The structural design which has that behavior is often called "Region Proposal Networks" and is apply in a sliding-window

 DOI : 10.18231/2454-9150.2018.0242 © 2018, IJREAM All Rights Reserved.

(v.i.z. convolutional) style across deep convolutional feature-maps. If you're familiar with vignette, Faster-RCNN/SSD connects the region proposal network to conv {3_3,4_3,5_3} features.

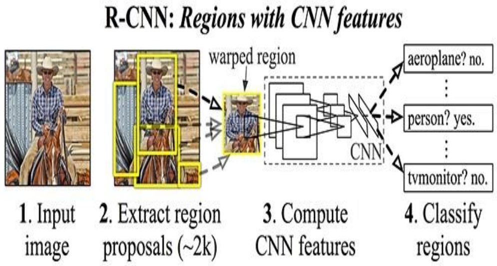

Figure 5: R-CNN Architecture

The reason of connecting region proposal networks to features of numerous depths is to take benefit of the high-resolution earlier in the network and deeper features later on in the network. A classification output is then computed from the features related with a spatial region within feature maps related with the detection in a way depending on the particular structural design (SSD has a higher throughput than Faster-RCNN because it incorporates categorization of all detected regions within a single forward pass with no any resampling). The loss used to train this network is a weighted sum of the categorization error (usual softmax) and the localization error. This loss is in common a sum over non-maximally-suppressed detected regions for the reason that the region proposal network gives dense predictions for each probable spatial region and for each probable anchor box but realistically we anticipate objects to occur more sparsely than that. Using thousands of anchor boxes is a typical feature-engineering choice that is required for the model to accommodate objects of varying sizes and aspect ratios, and Faster-RCNN can process ~5 images per second on a practical GPU whereas SSD can process tens per second with pretty much the similar accuracy.

4. DATASETS

Datasets plays a very significant role in research. CNN requires a huge amount of labeled data for training model. There are not a lot of datasets available but there are a small number of which work well. Common datasets used for object detection are Image Net, PASCAL VOC and MS COCO. It consist of 5 stages which includes classify, capture, server, cloud and cell phone. Objects usually found around homes are pre-categorized into the User objects and reference objects. Camera will capture images on the base of user movements. By performing the object analysis captured image will help to decide user objects within their location with help of reference object around them. A voice-search-mechanism that's integrated with Google-Now helps the user locate their objects when they require them, and render its image if requested.

Name	# Images (trainval)	# Classes	Last updated
ImageNet	450k	200	2015
COCO	120K	80	2014
Pascal VOC	12k	20	2012
Oxford-IIIT Pet	7K	37	2012
KITTI Vision	7K	3	2014

Figure 6: Datasets Used

The sequence includes three objects, the flow of process starts from wearable camera and ends at deep learning server in which the cloud serve will interact with other objects to perform the tasks.

5. SYSTEM DESIGN

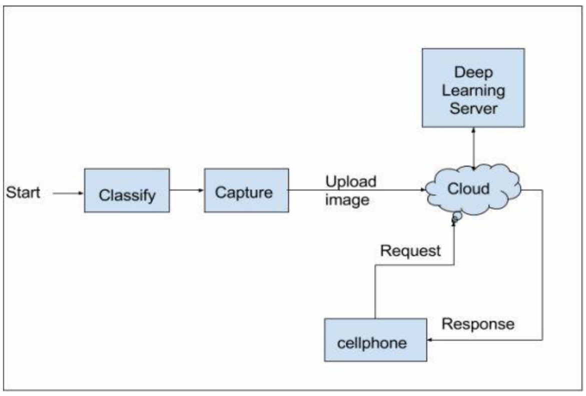

Figure 7: System Design

DOI : 10.18231/2454-9150.2018.0242
© 2018, IJREAM All Rights Reserved.

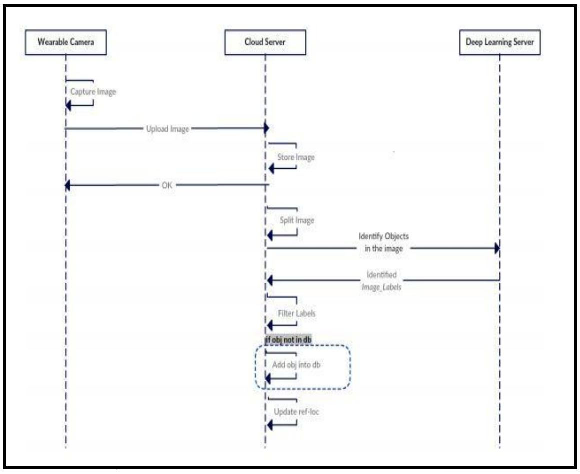

Figure 8: Sequence Diagram

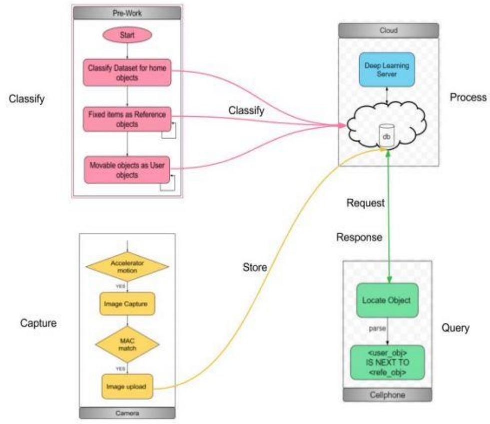

Figure 9: Activity Diagram

Classify: Objects usually found around us are pre-classified into subsequent two categories:

i. User Objects.

ii. Reference Objects.

This will be done prior to detection of object and then the capturing process will begin.

2. Capture: Every time user movement is observed, it will be observed by an 3-axis accelerometer which will observe the movement of the person so as to keep away from camera to be on at all times. The wearable camera acquires images of the objects in the region of it.

3. Process: An object examination is performed on the image captured. The CNN model will classify and detect all the objects captured and further will detect objects in close proximity them. This will decide user-objects within, and their location exactly from the reference-objects present close to them - and this data is stored into the cloud database.

4.Query: A voice-search-mechanism generally Google API thats integrated with Google-Now helps the user locate their objects when they require them, and render its image if requested.

6. EXPERIMENTAL RESULTS

The following screenshot shows the obtained experimental results. In results with respect to objects detected at the upper it shows object name with its accuracy.

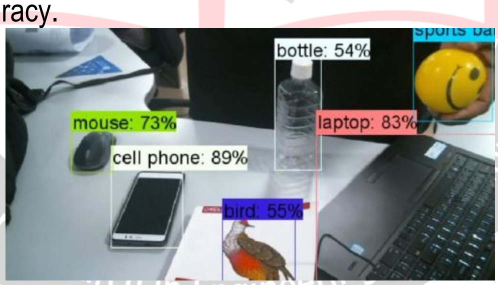

Figure 10: Experimental Result

7. CONCLUSION

We provided a proof of idea for the problem statement and be of assistance out people suffering from forgetfulness, and ultimately reduce their mental stress levels. By using standard hardware (Camera, Accelerometer) and software (CNN, TensorFlow, Google API) we have been intelligent to give a build solution to this diseases.

REFERENCES

[1] C. Szegedy, A. Toshev and D. Erhan, Deep Neural Networks for object detection, Advances in Neural Information Processing Systems, 2013, 26: 2553-2561.

[2] Ham F & Kostanic I, Principles of neuron Computing for science and engineering. McGraw- Hill 2001.

[3] Fundamentals of Artificial Neural Networks-Mohamad H. Hassoun. (Cambridge, MA: MIT Press, 1995, 544pp,hardbound, ISBN 0-262-08239-X).

 DOI : 10.18231/2454-9150.2018.0242 © 2018, IJREAM All Rights Reserved.

[4] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, and F.F. Li. ImageNet: A large-scale hierarchical image database, in Proc. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. 2009, 248-255.

[5] A. Krizhevsky, I. Sutskever and G.E. Hinton. ImageNet classification with deep convolutional neural networks, in Proc. International Conference on Neural Information Processing Systems. 2012, 1097-1105.

[6] H. Wu and X. Gu. Max-Pooling Dropout for Regularization of Convolutional Neural Networks: Springer International Publishing, 2015.

[7] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in ICLR, 2015.

[8] A. Shrivastava, A. Gupta and R. Girshick, Training Region-Based Object Detectors with Online Hard Example Mining, 2016.

With Best Compliments From

Climax

ADVERTISING AGENCY

Kolhapur • Pune • Belgaum • Goa

E-mail : climaxad@gmail.com

Phone : (0231)-2652976, 2664207

 DOI : 10.18231/2454-9150.2018.0242 © 2018, IJREAM All Rights Reserved.