IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Volume 10, PP 28-33

Automated Neural Image Caption Generator for Visually Impaired People

Yash Badani[1], Gunjan Bagayatkar[2], Sandesh Bhat[3], Sanket Benke[4], Prof. Sonali Jadhav[5]

[1, 2, 3, 4](Student, Computer Engineering, Rajiv Gandhi Institute Of Technology, Mumbai 400053, India)
[5](Professor, Computer Engineering, Rajiv Gandhi Institute Of Technology, Mumbai 400053, India)

Abstract: *Automated Neural Image Caption Generator Can Generate The Content Of An Image Using Well-Formed Meaningful English Sentences. Image Is Continuously Captured Real-Time Using User's Camera/Mobile Phone. Our Models Use A Convolutional Neural Network (CNN) To Extract Features From An Image. These Features Are Given To A Recurrent Neural Network (RNN) Or A Long Short-Term Memory (LSTM) Network To Generate A Valid Image Description In English. To Extract Features From The Image, We Use A CNN. Cnns Have Been Used And Studied For Variety Of Image Tasks, And Are Currently State-Of-The-Art Methods For Object Recognition And Detection. Concretely, For All Input Images, We Extract Features From The Fc7 Layer Of The VGG-16 Network Pre-Trained On Imagenet, Which Is Well Tuned For Object Detection. After Obtaining A 4096-Dimensional Image Feature Vector, We Reduce It Using Principal Component Analysis (PCA) To A 512-Dimensional Image Feature Vector Because Of Computational Constraints In LSTM. We Feed These Features Into LSTM Network To Generate A Description Of The Image In Valid English, Which Could Then Converted To Audio Using Text-To-Speech Technology.*
Keywords- *Caption Generator, Feature Extraction, LSTM, Neural Network, Object Detection*

I. INTRODUCTION

Visual Impairment Is Something That Any Person Does Not Opt To Have And It Does Not Have Any Kind Of Temporary Fixes. Visual Impairment Is Nothing But Disability To See Which Makes The Problem Not Fixable By Temporary Means. According To The World Health Organization, 285 Million People Are Visually Impaired Worldwide, Including Over 39 Million Blind People [1]. Living Without One Of The Most Useful Sensory Organ In A Technologically Developing World Where Even The Smallest Piece Of Work Would Require Sight Is Very Difficult. Living In The Era When The Technology Sector Is Booming There Can Be Many Developments Made That Could Improve The Lives Of Visually Impaired. There Are Several Ways In Which The Technology Can Provide An Aid To The Visually Impaired One Way Is By Detecting The Objects From An Image And Providing A Meaningful Caption That Would Be Read Out Loud Which Would Help The Person Using This System To Relate All The Objects In The Image.

The Challenges Faced Are Being Able To Automatically Describe The Content Of The Image Into Well-Formed English Sentences And The Caption Must Not Only Read Out The Objects Present In The Image But Also Must Express How These Objects Relate To Each Other Along With Their Attributes And The Activities They Are Involved In. These Descriptions Must Be Expressed Using A Natural Language Like English That Would Require A Language Model. This Can Be Achieved By Using Convolution Neural Network, Recurrent Neural Network, Imagenet Dataset, Text To Speech Convertor In Interrelation To Each Other.

II. DATASET

Dataset Is Collection Of Related Items, Which Would Help The System To Classify Different Categories Of Items And Grouping Them Together By Comparing It With The Dataset. In This Paper, We Use Two Types Of Dataset Imagenet And Microsoft COCO. Imagenet Consists Of Variable-Resolution Images, While Our System Requires A Constant Input Dimensionality. Therefore, We Downscale The Image To Some Fixed Resolution. Given A Rectangular Image, We First Rescale The Image Such That The Shorter Side Was Of Length 256, And Then Cropped Out The Central 256x256 Patch From The Resulting Image. Therefore, We Trained Our Network On The Raw RGB Values Of Pixels. For This Exercise, We Will Use The 2014 Release Of The Microsoft COCO Dataset, Which Has Become The Standard Testbed For Image Captioning. The Dataset Has 80,000 Training And 40,000 Validation Images, Each Annotated With 5 Captions Written By Amazon Mechanical Turk Workers [2].

III. CONVOLUTIONAL NEURAL NETWORKS

In Machine Learning, A Convolutional Neural Network (CNN, Or Convnet) Is A Class Of Deep, Feed-Forward Artificial Neural Networks That Has Successfully Been Applied To Analyzing Visual Imagery. CNN Compares Any Image Piece By Piece And The Pieces That It Looks For In An Image While Detection Are Called As Features. By Finding Rough Feature Matches In Roughly, The Same Position In Two Images CNN Gets Trained. Every Neuron In CNN Will Be Connected To Small Region Of Neurons Below It This Would Allow Handling Less Amount Of Weights And Number Of Neurons Required Will Also Be Less.

3.1 Convolution Layer

The Convolution Layer Is The Core Building Block Of A Convolutional Network That Does Most Of The Computational Heavy Lifting. It Preserves Spatial Relationship Between Pixels Thereby Extracting And Learning Features Out Of Them. The Image Is Represented As A Matrix And A Filter, Which Is Also A Matrix Is Used To Obtain The Convolved Feature Map Or Activation Map By Sliding The Feature Matrix Over The Image Matrix As Shown In Fig. 1. We Can Perform Operations Such As Edge Detection, Sharpen And Blur Just By Changing Values In The Filter Matrix. It Captures The Local Dependencies In The Original Image [3].

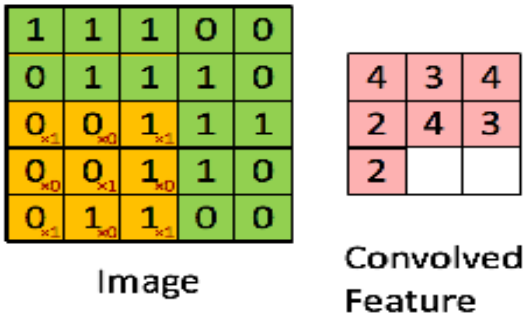

Figure 1: Obtaining Convolved Feature Map From Image Matrix By Sliding Filter Matrix Sequentially

3.2. Rectified Linear Unit Layer

It Is Called As Rectified Linear Unit(Relu), Which Is Nothing But An Activation Function That Activates A Node If An Input Is Above A Certain Quantity; While If The Input Is 0 Then The Output Will Be 0. But, If The Input Is Above Certain Threshold It Has A Linear Relationship With The Dependent Variable.

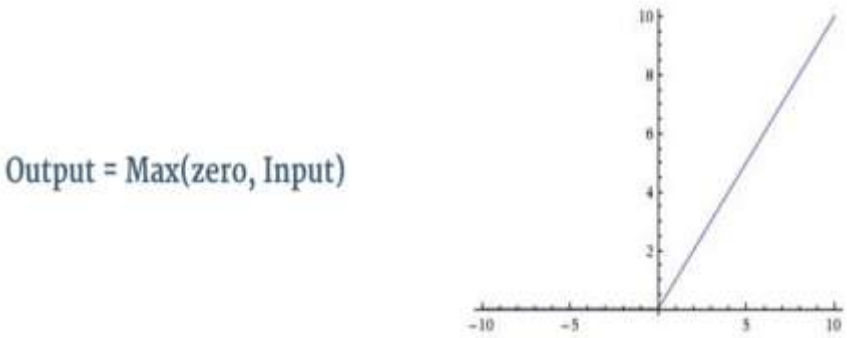

Figure 2: Relu Operation To Obtain Rectified Feature Maps

It Replaces All The Negative Values In The Feature Map By Zero And Generates A Rectified Feature Map As Shown In Fig. 2. Relu Introduces Non-Linearity In The Convnet Since Most Of The Image Data Are Non-Linear In Nature In The Real World [3].

3.3. Pooling Layer

In This Layer, We Reduce The Dimensionality Of The Feature Map To Get Smaller Or Shrinked Maps That Would Reduce The Parameters And Computations. Pooling Can Be Max, Average Or Sum Pooling From The Rectified And Downsized Feature Map. Number Of Filters In Convolution Layer Is Same As The Number Of Output Maps From Pooling As Shown In Fig. 3. It Also Makes The Network Invariant To Small Transformations, Distortions And Translations In The Input Image [3].

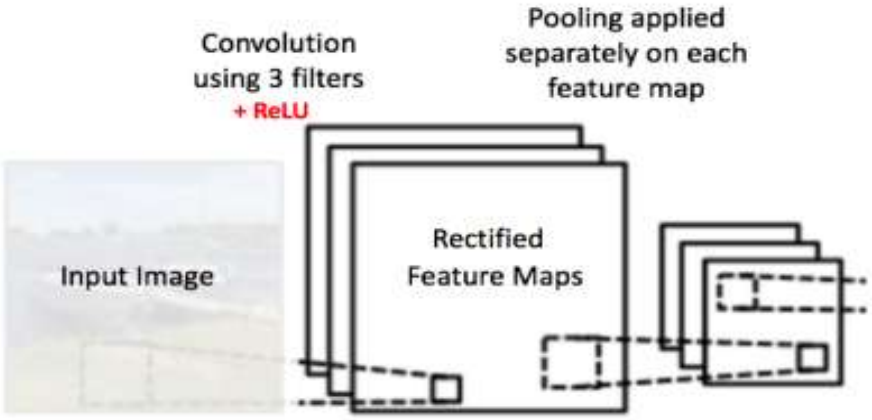

Figure 3: Pooling Applied To Rectified Feature Maps.

3.4. Fully Connected Layer

This Is The Final Layer Where The Actual Classification Occurs Where We Take Our Downsized Or Shrinked Images Obtained After Processing Through Convolution, Relu And Pooling Layer And Put Them Into Single List Or A Vector.[3]

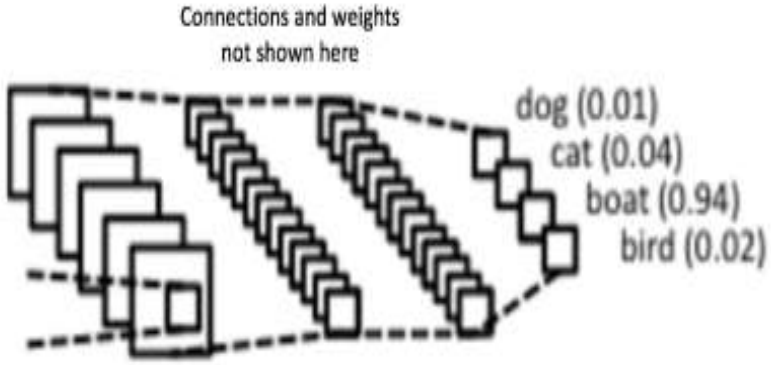

Figure 4: Fully Connected Layer

It Is A Traditional Multi-Layer Perceptron Uses A Softmax Activation Function. Convolution And Pooling Layers Generate High-Level Features. The Purpose Of The Fully Connected Layers Is To Use These Features To Classify The Data Into Various Classes Based On The Dataset.

3.5. Training Of CNN Using Back-Propagation

- Initialize All Filters And Parameters With Random Variables
- Take Input Images For Training, Go Through The Forward Propagation And Find Output Probability For Each Class.
- Calculate Total Error By The Formula:

  Total Error = $\sum$ ½ *(Target Probability – Output Probability)* ² (1)
- Calculate Gradients Of The Error With Respect To The Weights And Use Gradient Descent To Update The Filter Values And Parameters To Minimize The Output Error.
- Repeat 2-4 For All Images In The Training Set [3].

IV. RECURRENT NEURAL NETWORK

The Main Purpose Of RNN Is To Make Use Of Sequential Information. In Order To Predict The Upcoming Words In A Sentence It Is Very Much Necessary To Know The Previous Words. Rnns As The Name Suggests, Are Recurrent As They Perform Same Operations On Every Element Of A Sequence And The Current Output Is Dependent On Output Of Previous Computations.

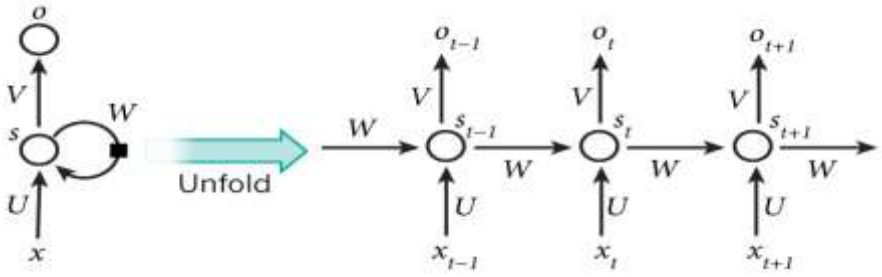

Figure 5: Recurrent Neural Network Forward Computation

Fig. 5 Shows That RNN Being Unfolded Into Full Network. Unfolding Is Nothing But Representing Network For Every Element In The Sequence. For Example, If A Sequence Is A Sentence With 7 Words Then The Network Will Be Unfolded Into 7 Layers, One Layer For Each Word. The Formulas That Govern The RNN Process Are As Follows: X_t Represents The Input At Time Step T. S_t Represent The Hidden State At Time Step't'. O_t Represents The Output At Step T. The Hidden State S_t Is The Memory Of The Network That Stores The Information Of The Previous Time Steps And Is Calculated As Previous Hidden State And Input At Current State:

$$S_t=F(Ux_t+Ws_{t-1}) \quad (2)$$

The Function *F* Is A Non-Linearity Function Such As Tanh Or Relu. S_{-1}, Which Is Required To Calculate The First Hidden State Is Initialized To Zeros O_t, The Output At Step T, Is Calculated Completely On The Basis Of Memory At Time T.

$$O_t=Softmax(Vs_t) \quad (3)$$

Unlike The Traditional Approach That Uses Different Parameters At Every Layer RNN Makes Use Of The Same(U,V,W) Vectors Across Every Layer Which Means That The Same Process Is Carried Out At Every Layer With Different Inputs. Rnns Have Shown Great Success In Natural Language Processing. The Most Commonly Used RNN Is Lstms Which Are Much Better At Capturing Long Term Dependencies And No Vanishing Gradient Problem. LSTM's Way Of Development Is Same But The Way Of Calculating The Hidden State Is Different [4].

V. LONG-SHORT TERM MEMORY

LSTM Are Special Kind Of Recurrent Neural Network And These Recurrent Neural Network Are Capable Of Learning Long Term Dependencies. LSTM Also Have A Chain Like Structure Like RNN [5]. The Network Takes 3 Inputs: X_t Is Current Input, H_{t-1} Is The Output From Previous LSTM Unit And C_{t-1} Is Previous Unit Memory Which Is Very Important. It Gives 2 Outputs: H_t Is The Output Of Current Network And C_t Is The Memory Of Current Unit. A Single LSTM Unit Makes Decision Based On Current Input And Previous Output As Well As Memory. Fig. 6 Depicts The LSTM Architecture Along With Its Various Gates.

- Forget Gate: It Removes Unnecessary Information From The Cell State, Which Is Not Required. This Is Done By Multiplication Filter. It Is Controlled By Single Layer Neural Network With A Sigmoid Activation Function, Output Of Which Is Applied To Old Memory To Decide If It Should Be Kept Or To Forget.
- New Memory Gate: It Is A Also A Single Layer Neural Network Whose Inputs Are Same As Forget Layer. It Controls The Influence Of New Memory Over Old Memory. Another Neural Network With Tanh As Activation Function Generates The New Memory. The Output Of This Network Multiplies With The New Memory Gate And Adds To Old Memory To Form New Memory
- Output Gate: The Output Valve Is Controlled By New Memory, The Previous Output Ht-1, Input Xt And Bias Vector. It Controls The Amount Of New Memory That Flows To Next LSTM Unit.
- The Following Are The Formulas To Derive The Value At Each Gate:

$$I_t = \Sigma(Wx_ix_t + Wh_ih_{t-1} + Wc_ic_{t-1} + B_i) \quad (4)$$

$$F_t = \Sigma(Wx_fx_t + Wh_fh_{t-1} + Wc_fc_{t-1} + B_f) \quad (5)$$

$$C_{t=} F_tc_{t-1} + I_ttanh(Wx_cx_t + Wh_ch_{t-1} + B_c) \quad (6)$$

$$O_t = \Sigma(Wx_ox_t + Wh_oh_{t-1} + Wc_oc_{t-1} + B_o) \quad (7)$$

$$H_t = O_ttanh(C_t) \quad (8)$$

Where,

I_t Is Input Gate Vector

F_t Is The Forget Gate Vector

C_t Is The Cell State Vector

H_t Is Output Of LSTM Cell

O_t Is The Output Of 3 Different Single Layer Neural Nets Having Values Between -1 And 1
Following Are The Steps For A LSTM Cell To Perform:

- *Step 1:* The First Step In LSTM Is To Decide What Information We Are Going To Throw Away From Cells State. This Decision Is Made By Sigmoidal Layer Called As Forget Gate Layer

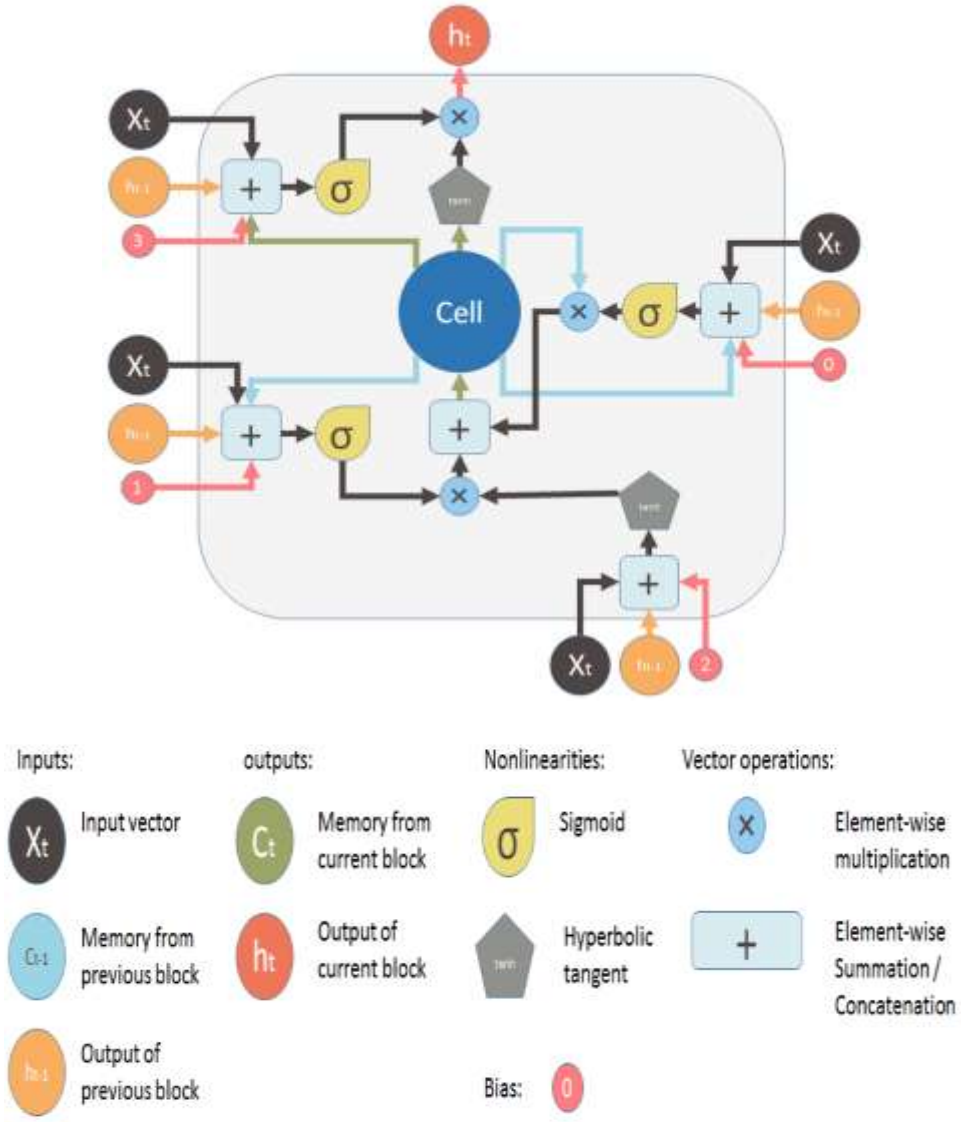

Figure 6: LSTM Architecture With Gates

- *Step 2:* In This Step We Decide What New Information We Are Going To Store In The Cell State.
- This Has 2 Parts, First, Sigmoid Layer Or Input Gate Layer Decides Which Values Will Be Updated. Then, The Tanh Layer Creates A Vector Of New Values That Could Be Added To The State.
- *Step 3:* Now We Will Update The Old Cell State Into The New Cell State.
- *Step 4:* We Will Run A Sigmoid Layer, Which Decides The Parts Of The Cell State We Are Going To Output. Then We Put The Cell Through *Tanh* And Multiply It By The Output Of The Sigmoid Layer, So That We Only Output The Parts We Decided To [5].

VI. TECHNICAL FRAMEWORK

The Proposed Architecture Involves All The Above Stated Techniques To Achieve An Almost 80% Accuracy In Scene Description. Fig. 7 Shows The Basic Block Architecture Of The Current System.

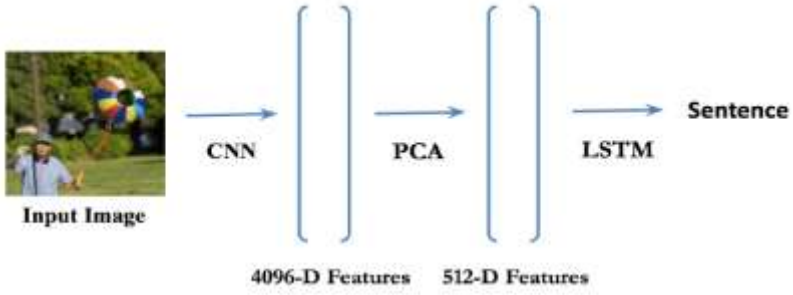

Figure 7: Image Extraction And Language Generation Pipeline Architecture

- The Features Are Extracted From Images Using A CNN And Are Extracted From Fc7 Layer Of The VGG-16 Network Previously Trained On Imagenet Dataset
- A 4096-Dimensional Feature Vector Is Obtained Which Is Reduced Using Principal Component Analysis (PCA) To A 512-Dimensional Vector Due To Computational Constraints For LSTM Input.
- Though, Vanilla Rnns Are Successful In Speech Recognition And Text Generation, It Is Difficult To Train Them To Learn Long-Term Dependencies Because Of Vanishing And Exploding Gradient Problem That Result From Propagating Gradients Through Many Layers Of Recurrent Networks [2].
- Therefore, LSTM Provides A Solution By Adding Memory Units, Which Allow The Network To Learn When To Forget Or Update Previous And Current Hidden States When Given New Information. Thus LSTM Is The Best Recurrent Network Solution Against The Vanilla RNN To Get The An Almost Perfect Caption [2].

VII. CONCLUSION

We Presented This Paper To Help Visually Impaired People By Using Deep Learning Techniques. Techniques Like Convolutional Neural Networks (CNN) And Feature Maps That Get Generated Using Such Neural Nets Help Us To Recognize Objects And Later Generate Sentences Using Recurrent Nets Such As Long-Short Term Memory (LSTM). The CNN And LSTM Are Currently The State-Of-The-Art Techniques For Object Detection, Scene Representation And Scene Description Such That The Generated Captions Are Highly Descriptive Of The Objects Depicted On The Images. Because Of The High Quality Of The Generated Image Descriptions, Visually Impaired People Can Greatly Benefit And Get A Better Sense Of Their Surroundings Using Text-To-Speech Technology. The Further Study And Research For This Project Is To Generate Captions For The Live Video To Give Real-Time Understanding And Describing The Scene To The User Instead Of Describing The Captured Static Images That Can Only Provide Blind People With Information About One Specific Instance Of Time.

Acknowledgement

We Sincerely Thank Our Project Guide Mrs. Sonali Jadhav For Her Guidance And Encouragement In Carrying Out This Research. We Wish To Express Our Sincere Gratitude To The Principal And The Head Of Department Of Computer Engineering Of Rajiv Gandhi Institute Of Technology For Providing Us An Opportunity To Do Our Research On "Automated Neural Image Caption Generator For Visually Impaired People". This Research Bears On Imprint Of Many People. Finally, We Would Like To Thank Our Colleagues And Friends Who Helped Us In Every Possible Way For Completion Of This Research Paper.

REFERENCES

[1] Bourne RRA, Flaxman SR, Braithwaite T, Cicinelli MV, Das A, Jonas JB, Et Al, "Magnitude, Temporal Trends, And Projections Of The Global Prevalence Of Blindness And Distance And Near Vision Impairment: A Systematic Review And Meta-Analysis," The Lancet Global Health, Sep. 2017 Vol. 5

[2] Christopher Elamri, Teun De Planque, "Automated Neural Image Caption Generator For Visually Impaired People," Stanford University, Department Of Computer Science CS224d, 2016

[3] Ujjwalkarn.Me, "An Intuitive Explanation Of Convolutional Neural Networks," 2016. [Online]. Available: Https://Ujjwalkarn.Me/2016/08/11/Intuitive-Explanation-Convnets/. [Accessed: 28-Feb-2018]

[4] Wildml.Com, "Recurrent Neural Networks Tutorial," 2015. [Online]. Available: Http://Www.Wildml.Com/2015/09/Recurrent-Neural-Networks-Tutorial-Part-1-Introduction-To-Rnns/. [Accessed: 17-Sep-2015]

[5] Medium.Com, "Understanding LSTM And Its Diagrams," 2016. [Online]. Available: Https://Medium.Com/Mlreview/Understanding-Lstm-And-Its-Diagrams-37e2f46f1714. [Accessed: 14-Mar-2016]