LifeLens: Deep Learning Based Application for the Visually Impaired

Balasubramanian S
Dept. of ECE
PES University
Bengaluru, India
balavish580@gmail.com

Surya Teja CVN
Dept. of ECE
PES University
Bengaluru, India
cvnsuryateja@gmail.com

Aakash Tomar
Dept. of ECE
PES University
Bengaluru, India
tomaraakash16@gmail.com

Thanikonda Sai Kiran
Dept. of ECE
PES University
Bengaluru, India
tskiran242000@gmail.com

Niranjana Krupa
Dept. of ECE
PES University
Bengaluru, India
bnkrupa@pes.edu

TENCON 2021 - 2021 IEEE Region 10 Conference (TENCON) | 978-1-6654-9532-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/TENCON54134.2021.9707409

Abstract— The swift rise in the field of Deep Learning and Computer Vision is transforming how we approach important problems. This paper aims at aiding the visually impaired with environment navigation using a novel prototype which utilizes prevalent solutions to the Image Captioning and Visual Question Answering problems. The IC model was implemented using two approaches - an encoder-decoder framework that incorporates a visual attention mechanism and a Reinforcement Learning (RL) Based Policy-Value Network. The performance of the two models were evaluated using the BLEU score metric. The visual attention model performed relatively better and attained an average score of 0.47. This model was combined with a Text-to-speech mechanism and was deployed on the Raspberry Pi thus facilitating the visually impaired with environment navigation. Further, a Long Short Term Memory (LSTM) network was implemented for Visual Question Answering applications.

Keywords— Image Captioning, Feature Extraction, Deep Learning, Transfer Learning, Reinforcement Learning, Raspberry Pi, Visual Question Answering

I. Introduction

According to the statistics obtained from the World Health Organization, 39 million people in the world have been diagnosed with visual impairment and 246 million people have been affected by low vision. Visually Impaired people face several challenges in their daily lives and one of the most important problems is the identification of objects and obstacles.

The proposed work comprises two models namely, IC and VQA. The implementation of the Image Captioning (IC) using Deep Learning networks aids the visually impaired by enhancing their capability of environment perception and navigation. The incorporation of local attention mechanisms helps the IC model on focusing towards important regions by assigning weight to different sections of the captured image, thus leading to meaningful descriptions [5] and a RL based Policy Value Network was implemented, which helps in determining the word generated at each timestamp based on the reward obtained by the action taken on the input image [7]. An interactive application, Visual Question Answering (VQA) model was developed using LSTM network thus aiding the visually impaired by generating answers based on the queries asked by them [9], [10], [11].

Novelty in this paper focuses on comparing different IC models, evaluating their performances and deploying it on Raspberry Pi. Further, a VQA model with the integration of text-to-speech was experimented. These two applications provide the visually impaired with a feeling of real time experience and inclusiveness within the society.

II. Related Work

There has been significant research and contribution done in the field of Artificial Intelligence towards Image Captioning applications. The authors in [1] were able to achieve a good correlation between the image and it's description after performing analysis on different Convolutional (CNN) and Recurrent Neural Networks (RNN) architectures. It was seen that several transfer learning networks like VGG-16, AlexNet which were able to achieve a high accuracy for object classification tasks were able to provide a very efficient internal representation of the image during the feature extraction phase [2]. Different frameworks, approaches and model designs were proposed for solving the IC tasks. One such approach was the construction of a CNN and a bi-directional RNN model which were used to represent the image features and words respectively and were mapped to the same multimodal embedding space [3]. A unique encoder-decoder framework was introduced in [4], where the CNN was used as an encoder and the RNN as the decoder to generate the image description. An innovative local attention mechanism was proposed in [5] which focuses on providing more weights on certain sections of the image, thus providing clear and relevant captions. The RL mechanism was introduced in [6], [7]. It uses a policy and value network which work concurrently to determine the upcoming word at each timestamp. The BLEU score metric helps in evaluation by identifying the closeness or the correlation between the generated caption and the reference caption [13].

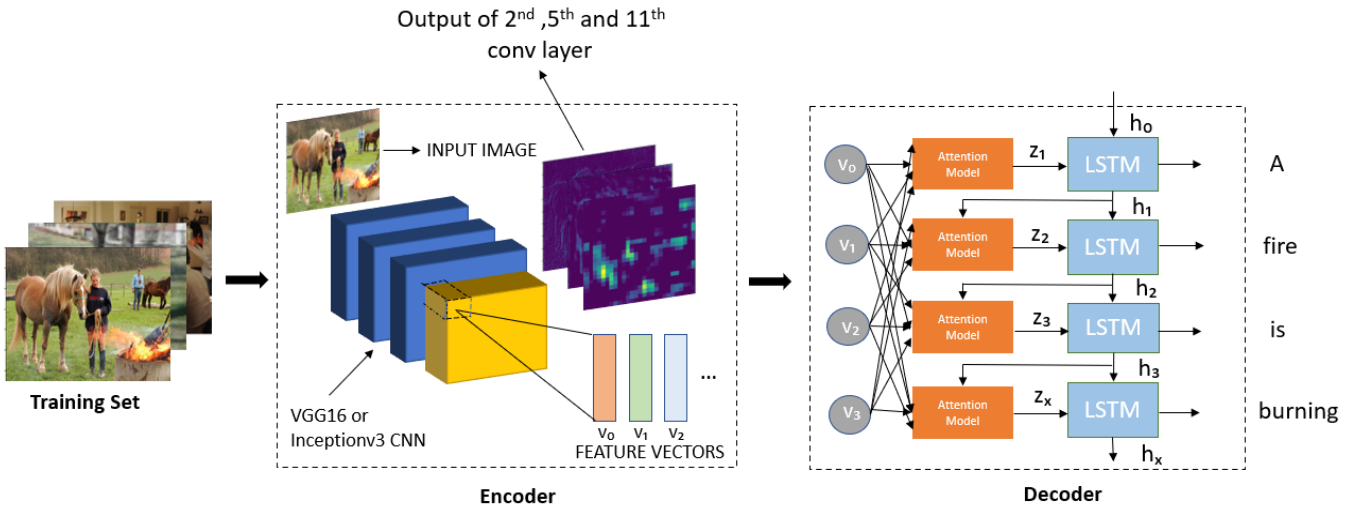


Fig1. Working of Image Captioning Model

The VQA helps in providing more clarity about the surroundings to the visually impaired. The authors in [8], [9], [10] have focused on dealing with different types of questions asked and proposed ways on improving the performance of VQA models by utilising the shared knowledge between the process of generating captions and that of VQA. Since a caption contains an overall description, it can be used to answer questions pertaining to the image. In [11], the VQA model was constructed using the CNN LSTM network where the CNN was used for extracting the essential features and the LSTM was used for embedding the question features. These two feature representations which are independent of each other were combined and passed to the softmax layer to generate appropriate answers. Several insights about the importance of different components used in developing a prototype for the visually impaired were gained from [14].

The work performed in this paper is an extension of [4] where the focus is on capturing the essence of the image by identifying important features through *Bahdanau* attention principle, thus leading to concise and relevant captions. Further, a RL based policy - value model was implemented as in [6],[7], to compare its performance with the visual attention model.

III. METHODOLOGY

The two eminent approaches to Image Captioning - encoder-decoder based visual attention mechanism and RL with embedding reward are discussed below. The objective of this section is to discuss the components and steps revolving around the implementation of these techniques in detail. Fig 1 shows an overview of the IC attention model where an image is passed through a pre-trained VGG16 or Inceptionv3 network for feature extraction. The representation of the different features are shown in the encoder block in Fig 1. The extracted features are then passed through an attention module to recognize important features and map them to the ground truth caption.

A. Image Captioning

1. Dataset

The IC model was trained using the MSCOCO 2014 dataset. The dataset contains 91 object categories and three image splits - 40,504 validation, 82,783 training and 40,755 testing images with 5 captions associated with each image [15].

2. Text Preprocessing

In the text preprocessing phase , the captions associated with each image are prefixed with a <start> token and suffixed with an <end> token. The tokens are added to facilitate the encoding decoding operation - on the encoding side, the <end> token signals to the encoder that the input sentence is complete; on the decoding side, the tokens help to avoid emission of random length sequences. All the captions are tokenized, converted to lowercase, punctuation marks and words that do not play a significant role in describing the image are removed. A vocabulary of words is created containing the most frequently occurring words present among all the captions of the MSCOCO dataset. Every word in the vocabulary is identified by a unique integer (index). This is done in order to ensure all the sequences of text can be converted into sequences of numbers. Words that are not part of the vocabulary are represented by a <UNK> token. The caption having maximum length is identified. All sequences are padded with zeros for the calculated maximum length in order to ensure a standard length is maintained among the captions.

3. CNN RNN Visual Attention Model

An encoder - decoder framework architecture [4] was used in building an image captioning module where the encoder is used to capture the salient and prominent features and the decoder uses these feature vectors to generate relevant captions. An attention model [5] was integrated into our neural network and was trained using a back propagation algorithm in order to ensure the optimized weights that correspond to minimum loss were obtained.

a) Encoder

i) VGG 16: It is a 16 layer CNN architecture with pre-trained weights of the ImageNet dataset that provides a very fine internal representation of an image. The images that are passed as input are resized to a resolution of 224 X 224 X 3 which is the defined input size format of the VGG 16 model [2]. A feature vector of 4096 dimensional length is extracted from the last hidden layer and is given as input to the decoder as shown in Fig 1.

ii) Inception-v3: The Inception-v3 is another transfer learning network that helps in capturing the important features effectively. The images that are passed as input to this network are resized to a format of 299 X 299 X 3 and its corresponding pixels are normalized to a range of -1 to -1 which is the defined input size format for Inception-v3 [2]. Feature vector of 2048 dimensional length is extracted from this transfer learning network and is passed as input to the decoder as shown in Fig 1.

b) Decoder

A local attention mechanism was deployed in the decoder to ensure relevant portions of the image are captured for generating captions. The encoded image features which are obtained from the transfer learning networks are passed through a dense layer consisting of 256 nodes before being used as input to the attention module. At each timestep, the attention module takes the encoded image feature vectors and the hidden state obtained from the previous timestep of the decoder as inputs and computes a attention score by assigning weights to each pixel present in the image. Since each encoded feature vector represents different sections of an image, the corresponding weights represent the attention and focus towards that region. The attention weights are calculated using the Bahdanau attention principle [5] which are then used in the generation of the context vector. The context vector is concatenated with the output of the decoder from the previous time step (In the first timestep there would be no previous output from the decoder and therefore the <start> token is concatenated with the context vector) and is fed as input to the LSTM decoder which consists of a recurrent neural network built with gated recurrent units to produce an output sequence along with a hidden state. This cycle is repeated until an <end> token is predicted to generate a caption that maintains the essence of the image.

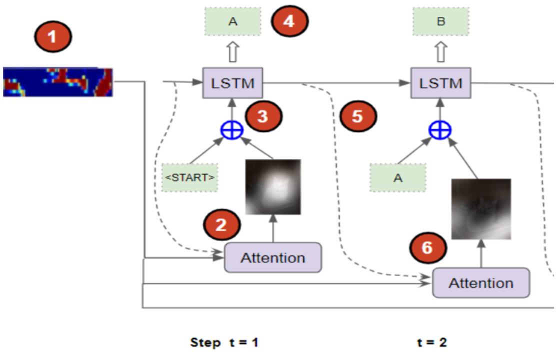


Fig 2. Working of Decoder with Attention Mechanism

c) Local Attention (Bahdanau)

The intuition behind the local attention mechanism can be understood better with a very good example. The transfer learning encoders ensure that all the features of an image are extracted by generating hidden states (**h1, h2...etc**). Each hidden state represents a particular region of an image. If **h1** is the hidden state that corresponds to the **frisbee** region and contains information about it, then more attention and focus should be given to that state by the decoder compared to the other state while generating that particular word.

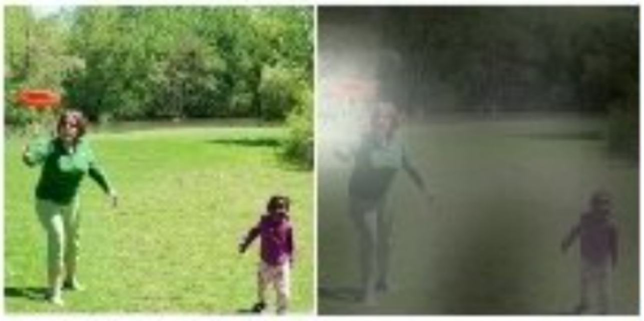


Fig 3. An example for Bahdanau Attention Principle

A feed forward neural network is trained to understand the mapping between the different sections of the image and its corresponding words by generating a high score for the relevant state and low score for those that are going to be ignored. If **s1,s2,s3**..etc are the scores generated for the corresponding hidden states **h1,h2,h3**..etc then the score **s1** should be high while predicting the word frisbee while the scores **s2,s3**..etc should be low. Once the scores are generated, the softmax function is applied to produce the attention weights **e1, e2, e3**..etc. The advantage of the softmax function is that all weights are normalized between 0 and 1 and their sum is equal to unity. This means that while generating the word frisbee the attention weight **e1** would be maximum. The context vector is then generated by multiplying these weights with the corresponding hidden states as shown in Eq (1). Once the context vector is formed, the steps mentioned in the **"Decoder"** section are followed.

$$\text{Context Vector} = e1 * h1 + e2 * h2 \ldots.. \quad (1)$$

4. *Reinforcement Learning*

The use of a decision making framework while solving a problem forms the basis of RL. RL solutions typically involve 3 components - policy, value and reward. The state and action space can be considered as the two other components here, with the state being observed after every action. The key differentiating factor in this approach to image captioning problem is the presence of both global and local guidance for making a decision, policy network being the local guidance by providing means to predict the next word and value network being the global guidance by calculating rewards for all possible states [6].

Any decision-making process always involves an agent that reacts with the environment and performs a set of actions to achieve a target. In an IC application, the target or goal is to generate a meaningful sentence (caption) $\mathbf{S = \{w_1, w_2, ..., w_T\}}$ that provides a relevant description for an image '**I**' where '$\mathbf{w_j}$' is a word in caption '**S**' generated at each timestamp and '**T**' is the length. The caption generator can be considered the agent. The state can be considered as the image features and the caption generated up till that point of time. The action space is the set of actions available to the agent, and hence a fitting mapping for it would be the available vocabulary.

The policy network $\mathbf{p_\pi}$ is built using a CNN-RNN architecture, similar in nature to the architecture utilised in the conventional encoder-decoder framework. The CNN is used to encode visual information in **I**, after which it is fed to the RNN's input node for recurrent processing of semantic information.

The value network $\mathbf{v_\pi}$ is built using a CNN-RNN-MLP architecture. The visual and semantic processing is performed at the CNN-RNN level respectively. The value network finds it's major utility in the computation of the reward with the additional architecture, as mentioned above.

The goal of a RL based approach is to maximize the decided reward, and hence the coordinated working of the different components mentioned above aims to maximize the similarity between the visual and semantic embedding of the image and the ground truth captions.

B. VQA

1. Dataset

The VQA model was trained using the MS COCO V2 VQA training set which has 82,783 base images, with at least 5 different questions associated with image and 10 annotation/answers for each question of a particular image. The dataset consists of 3 different types of questions, which are Numerical type, Yes or No type and Other types such as What, When, Where etc. and these types would help the model in being robust and in giving answers/solutions to any kind of questions/problems.

2. Preprocessing

The input images from the dataset have been resized to a resolution of 224x224x3 to ensure that it is compatible with VGG-16. The extraction of features from the images is performed by VGG-16. Among the 17000 unique answers that exist in the dataset, only the 1000 most frequently occurring answers are chosen, thus making it a classification problem by considering the 1000 answers to be 1000 unique classes.

3. LSTM Based VQA Model

VGG-16 architecture is being used to extract images and the question features are obtained by using a 3-layer LSTM network. A feature vector of 4096 dimensions is extracted from the last hidden layer of the VGG-16 network. The extracted vector consisting of features is then passed to a dense hidden layer consisting of 1024 nodes. Each question is forwarded to a word embedding layer which generates the corresponding word vector before being passed to the multi-layer LSTM network. The LSTM has hidden states embedded in them. The question feature vector is obtained from the LSTM's terminal hidden state which is then directed to the dense layer. Thereby, a new dense layer is created and is formed by the integration of the image and the question dense layer. The new dense layer is given to the fully-connected layer that consists of 1000 output nodes. Finally, the softmax layer is used to compute and predict the probabilities of the answers generated [16]. Weights in the network are adapted and updated periodically using the back propagation mechanism.

C. Hardware and Proposed Prototype

The primary goal while creating the prototype is to facilitate a three fold process - capturing periodic images, describing the captured images through a summarised caption, and answer the user's questions pertaining to the image. The sub-systems comprising the prototype, therefore, should include units like visual input, audio input, computational unit and speech output.

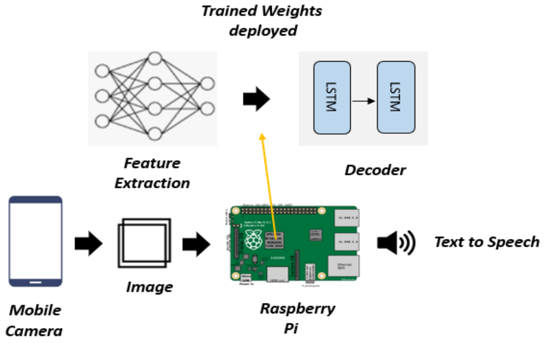


Fig 4. Real time working principle of the IC prototype.

Visual Input

Amongst the several available options for a visual input device, the prototype utilises a mobile phone camera to capture images from the environment. This method is chosen primarily due to an enhanced UX, wireless mode of operation and affordability.

Audio Input

Speech input can be obtained through conventional microphones. It is, however, necessary to convert the obtained speech to text, so as to facilitate computation. This can be done internally.

Computational Unit

The computational unit is the central unit of the prototype. It is utilised for various tasks like predicting a caption using trained model checkpoints, providing relevant answers to input questions, speech to text conversion and text to speech conversion. The prototype utilises Raspberry Pi Model 3B+ for this purpose.

Audio Output

The caption and the computed answer must be presented to the user via audio (speech) output. This is accomplished using headphones, connected to the computational unit.

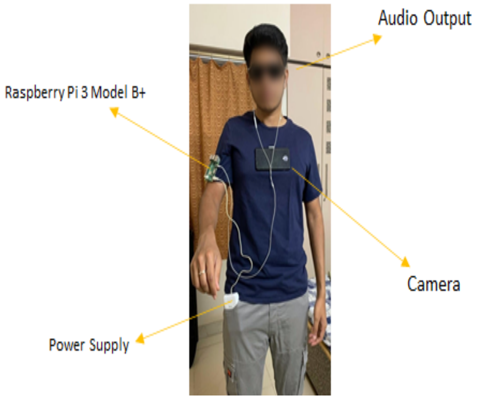


Fig 5. The proposed prototype and overview design

V. RESULTS

A. *Image Captioning*

1. *Training and Optimization*

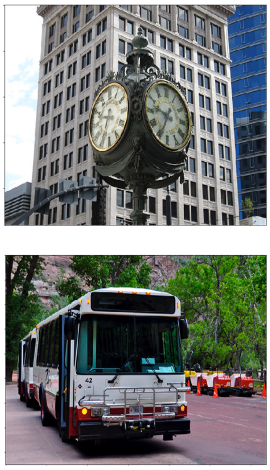
The training was performed through a **"Teacher Forcing"** principle which helps the IC model in learning the correct statistical properties of a sequence by passing the target word of the ground truth sequence as the next input to the decoder at each time step instead of the predicted word. There might be scenarios where the predicted word at a particular time step might be wrong and passing that as input in the next time step would propagate error into the network. Teacher forcing solves this problem. The loss at each timestamp was computed using the categorical cross - entropy technique. The Adam stochastic gradient descent method was used to compute the different adaptive learning rates. The IC visual attention model was trained on 50,000 images from the train split (82,783) of the MSCOCO 2014 dataset. As for the RL based embedding reward technique, the best loss of 0.3 was taken with a batch size of 50, and a backpropagation loss reduction methodology was followed for the captions generated via the policy network (the rewards for which were subsequently computed).

2. *Experimental Results*

The IC module's performance was evaluated using the **Bilingual Evaluation Understudy** metric also known as BLEU scores. The basic principle of bleu metric is to compare a set of generated captions and reference captions , count the matches (correlation) between them and assign scores between 0 and 1, a score of 0 indicating the generated captions are irrelevant and a score of 1 indicating that the generated captions are perfectly same as reference captions [13]. A score greater than 0.4 is interpreted as high quality translations. The trained IC models were evaluated on 100 randomly selected images from the test split of the MSCOCO dataset and their corresponding BLEU scores were averaged and tabulated below (TABLE 1). After evaluating the different IC models and comparing their performance it was observed that Visual Attention model with **Inception v3 encoder framework** was able to achieve the highest BLEU Scores (**0.47471, 0.34059, 0.32660, 025883**) compared to its counterparts.

TABLE I. IC Model Performance Average Scores

Metric Scores	**RL**		**Visual Attention**	
	VGG 16	Inception v3	VGG 16	Inception v3
BLEU 1	0.33745	0.43791	0.40046	0.47471
BLEU 2	0.18381	0.32136	0.28749	0.34059
BLEU 3	0.12767	0.31405	0.28825	0.32660
BLEU 4	0.10370	0.21094	0.22930	0.25883



Visual Attention Model : A clock post has a massive clock for its surroundings on a brick building

RL : A close up of a clock tower

Visual Attention Model : This bus is walking down a bus stop with passengers across the side.

RL : Two public transit buses parked on a city street.

Fig 6. Generated Captions from the two different IC Models

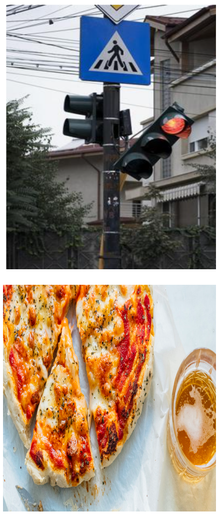
B. *VQA*

1. *Training and Optimization*

The LSTM model was trained on 50000 random samples present in the training set (82,783) of MSCOCO VQA 2014 dataset. An adaptive Adam optimizer was used to minimize loss during the training phase by using the gradient descent method to adjust weights with 0.001 as the suitable rate of learning. The loss at each timestamp was computed using the categorical cross entropy.

2. *Experimental Results*

The VQA model requires two inputs : The captured image and a speech input which is the query from the user .The model then processes these inputs and outputs the relevant answers through an audio which is obtained by combining the model with text to speech mechanism. An accuracy of 50.8% was obtained through the CNN LSTM VQA based approach.



Question 1: What colour is the signal?
Answer : Dark

Question 2: What is in the image?
Answer: Double Decker

Question 3: What does the traffic light indicate?
Answer: Pole

Question 1: What is the colour of pizza?
Answer: Yellow

Question2 : How many pieces of pizza are there
Answer: 2

Question 3: Is there a drink next to the pizza?
Answer: No

Fig 7. Predicted Answers from the LSTM based VQA Model

V. CONCLUSION

This paper compares the prevalent approaches to solving the image captioning and visual question answering problems. It also utilises a chosen optimum approach in a novel prototype which facilitates environment navigation for the visually impaired.

It is observed that although conventional CNN-LSTM networks serve as a good starting point to solving the problem of image description, they are not necessarily the best approach. Indeed, the more optimum approaches involve a more localised attention based solution to the problem. A primary characteristic of such techniques is the identification of interest areas in an image by assigning weights to them. A RL reward based Policy - Value Network was also explored and implemented.

The performance of the two IC models with different encoder frameworks were evaluated using the BLEU score metric and it was observed that the Visual attention model with an average score of **0.47** performed relatively better than its counterparts and was hence combined with text to speech mechanism and was deployed on the hardware.

The paper also explores solutions to the problem of query based description of an input image through a LSTM network. The model achieved an accuracy of **50.8%**. It is observed that the problem is inherently more complex and subjective compared to the captioning problem, due to the varied nature of the input questions. While binary questions requiring a 'Yes/No' answer are easier to deal with, open ended questions need a more sophisticated approach.

It is also observed that in order to act as an efficient environment navigation tool, optimization would be a primary focus for the prototype. The interval between two consecutive predictions is an important facet of it's practical operation.

While some prevalent solutions to the problems mentioned were compared, the insights and results discussed invite future research on how these methods can be optimised to achieve better results. Optimum integration and coordination of the IC and VQA subsystems can also be studied and worked upon. The implementation of a robust speech input system and the translation of human speech into a form more conducive for analysis are some of the other potential areas of endeavour.

REFERENCES

[1] Sutskever, I., Vinyals, O., & Le, Q.V. (2014), "Sequence to Sequence Learning with Neural Networks," *NIPS*.

[2] M. Shaha and M. Pawar, "Transfer Learning for Image Classification," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018, pp. 656-660, doi: 10.1109/ICECA.2018.8474802."

[3] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," *arXiv:1412.2306 [cs.CV]*, pp. 3128–3137, 2015.

[4] O. Vinyals, A. Toshev, S. Bengio and D. Erhan, "Show and tell: A neural image caption generator," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, 2015, pp. 3156-3164, doi: 10.1109/CVPR.2015.7298935

[5] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, Yoshua Bengio, "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention," Proceedings of the 32nd International Conference on Machine Learning, PMLR 37:2048-2057, 2015.

[6] Z. Ren, X. Wang, N. Zhang, X. Lv and L. Li, "Deep Reinforcement Learning-Based Image Captioning with Embedding Reward," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 1151-1159, doi: 10.1109/CVPR.2017.128.

[7] N. Xu et al., "Multi-Level Policy and Reward-Based Deep Reinforcement Learning Framework for Image Captioning," in IEEE Transactions on Multimedia, vol. 22, no. 5, pp. 1372-1383, May 2020, doi: 10.1109/TMM.2019.2941820.

[8] S. Antol et al., "VQA: Visual Question Answering," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, pp. 2425-2433, doi: 10.1109/ICCV.2015.279.

[9] Wu, Jialin & Hu, Zeyuan & Mooney, Raymond. (2019), "Generating Question Relevant Captions to Aid Visual Question Answering," 3585-3594. 10.18653/v1/P19-1348.

[10] S. Sarath & J. Amudha, "Vısual question answering models Evaluation," 2020, *International Conference for Emerging Technology (INCET),* 2020, pp. 1-5, doi: 10.1109/INCET49848.2020.9154094.

[11] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu, "Are you talking to a machine? dataset and methods for multilingual image question,"NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems, pp. 2296–2304, 2015

[12] Jiang, Y., Natarajan, V., Chen, X., Rohrbach, M., Batra, D., & Parikh, D. (2018). "Pythia v0.1: the Winning Entry to the VQA Challenge 2018." *ArXiv, abs/1807.09956*.

[13] Papineni, Kishore & Roukos, Salim & Ward, Todd & Zhu, Wei Jing. (2002), "BLEU: a Method for Automatic Evaluation of Machine Translation," 10.3115/1073083.1073135

[14] H. AlSaid, L. AlKhatib, A. AlOraidh, S. Al Haider and A. Bashar, "Deep Learning Assisted Smart Glasses as Educational Aid for Visually Challenged Students," 2019 2nd International Conference on new Trends in Computing Sciences (ICTCS), Amman, Jordan, 2019, pp. 1-6, doi: 10.1109/ICTCS.2019.8923044

[15] Lin, T.-Y., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P. & Zitnick, C. L. (2014). "Microsoft COCO: Common Objects in Context.. *CoRR"*, abs/1405.0312.

[16] Ren, Mengye, Ryan Kiros and R. Zemel, "Exploring Models and Data for Image Question Answering," *NIPS* (2015).