Atlantis Highlights in Computer Sciences, volume 4
Proceedings of the 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021)

Deep Cross of Intra and Inter Modalities for Visual Question Answering

Rishav Bhardwaj[1,*]

[1]*Cambridge Institute of Technology*
[*]*Corresponding author. Email:rishav.16cs127@citech.edu.in*

ABSTRACT
Visual Question Answering (VQA) has recently attained interest in the deep learning community. The main challenge that exists in VQA is to understand the sense of each modality and how to fuse these features. In this paper, DXMN (Deep Cross Modality Network) is introduced which takes into consideration not only the inter-modality fusion but also the intra-modality fusion. The main idea behind this architecture is to take the positioning of each feature into account and then recognize the relationship between multi-modal features as well as establishing a relationship among themselves in order to learn them in a better way. The architecture is pretrained on question answering datasets like, VQA v2.0, GQA, and Visual Genome which is later fine-tuned to achieve state-of-the-art performance. DXMN achieves an accuracy of 68.65 in test-standard and 68.43 in test-dev of VQA v2.0 dataset.

Keywords: *Deep Learning, Inter-Modality Fusion, Intra-Modality Fusion, Visual Question Answering*

1. INTRODUCTION

The field of deep learning has been consistently pushing its boundaries further to achieve better results. There are different tasks which are tackled using deep learning and one such is Visual Question Answering (VQA). Other tasks like object classification which involve uni-modal features or even tasks like image captioning which require multi-modal features are easier compared to VQA. In VQA, intricate details from both the features are important to get the desirable result.

The extraction of features from the image and question independently needs to happen and then the features need to be examined and combined. The favourable outcome of convolutional neural networks in the field of vision was identified in ILSVRC 2012 challenge [1]. Instead of taking features from the entire image into consideration like ResNet-152 [2], attention mechanism is used. It is seen that attention mechanism for vision [3], text [4] and speech [5] produce better results for the uni-modal tasks. Similar idea was introduced in VQA and the results improved.

The Figure 2 shows technique used to answer the question related to the image. It uses attention mechanism to extract only the reasonable features from the image and then the features of the question extracted. Section 3 explains how the features from both the modalities are made to achieve required solution.

Figure 1. Showing few results of DXMN network on the VQA 2.0 dataset.

Compared to other papers, the entire focus is to construct the relationship between the two modalities. Here in DXMN, instead of just prioritizing an inter-modality relationship, similar priority is given to intra-modality relationship. Figure 1 shows few results of the DXMN network on the VQA 2.0 dataset. It takes as input an image and a question related to the image and finally outputs the answer to question.

2. RELATED WORK

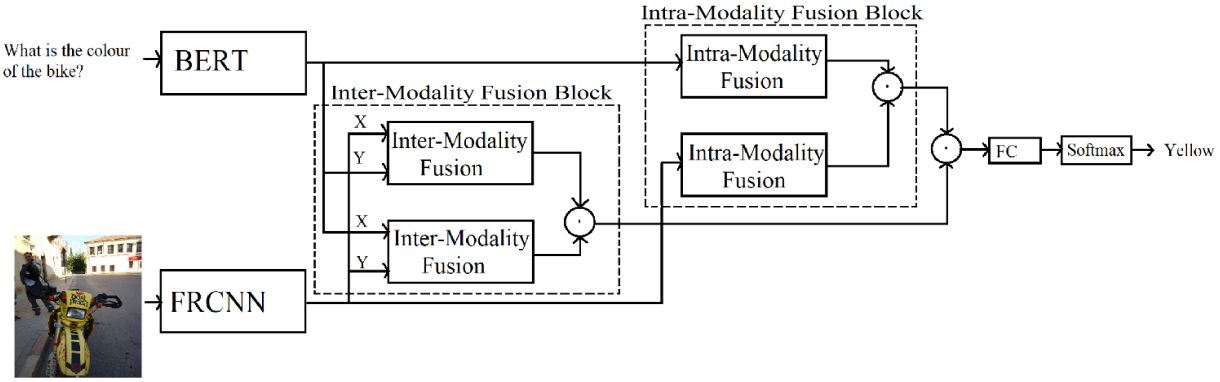

Figure 2. Overview of the proposed DXMN architecture that answers question about the image. The Inter-Modality Fusion block takes X and Y as inputs, while Intra-Modality Fusion block takes A as input.

The methods proposed to tackle the VQA problem kept improving as the time passed by. During the initial stages, bag of words model was used to decode the question meaning, later the embeddings of words were passed through gated recurrent unit (GRU) [6] or long short term memory (LSTM) [7] to get the question semantic. The image was passed through VGG [8] or ResNet [2] to get the spatial features of the image.

The VQA task achieved better result compared to traditional approaches after the bottom-up and top-down approach [9] which used Faster R-CNN [10]. In bilinear fusion [11], the features from both the modalities required higher number of parameters which made it fused using bilinear pooling. However, bilinear pooling is computationally expensive. Then few more fusion approaches were proposed like multimodal compact bilinear (MCB) pooling [12] and Multimodal Tucker Fusion (MUTAN) [13] which used comparatively lesser number of parameters and achieved better results.

In coattention mechanism [14], different regions from the images and question were merged independently to establish unique relationship among themselves. [15] tried to further extend this mechanism, the main idea was to obtain the language concept of the image instead of just focusing on the features of image.

Recently, LSTMs [7] and GRUs [6] were replaced by Bidirectional Encoder Representations from Transformers (BERT) [16], which improved the accuracy. One of the issues earlier with the word embeddings that were fed to LSTMs [7] and GRUs [6] was that for a particular word it was always the same irrespective of taking into account the entire question semantic. This problem was solved after introducing BERT [16]. In Learning Cross-Modality Encoder Representations from Transformers (LXMERT) [17], the usage of BERT [16] was taken one step further. Instead of just using BERT [16] for question, it was also used for processing the different objects present in the image.

Apart from coming up with newer architecture to tackle the VQA problem, there are training techniques [18, 19, 20, 21] put forward which might help to increase the accuracy. A special care is taken in the dataset while training which takes the semantic changes in the input data into consideration that might affect the output.

3. PROPOSED MODEL

The final objective of any VQA architecture is to find an answer to a question about an image. It takes into account, an image and a question as inputs and outputs a most appropriate answer to the question. This section proposes the architecture used in this paper. It is pictorially represented in the Figure 2.

As shown in the Figure 2, the image is represented using sequence of objects which comes from the output of Faster R-CNN [10] and the pretrained BERT [16] is used for extracting the token embeddings which represent the words in the question. For extracting the token embeddings, sum of last four layers of the BERT [16] output is taken.

The entire architecture is mainly divided into two components (1) Inter-Modality Fusion Block and (2) Intra-Modality Fusion Block which will be explained in the following subsections in greater detail.

3.1. Inter-Modality Fusion Block

In further discussion, each image feature corresponds to individual object which is obtained from Faster R-CNN [10] and feature of question is nothing but the token which is obtained from BERT [16]. In this paper, from each image top 36 objects were taken from the output of Faster R-CNN [10] based on their confidence values, and the question which was severed as input was represented in 20 tokens which is nothing but the output of BERT [16] after passing question through it. After taking the sequence of objects from the image which comes from

the output of Faster R-CNN [10] and the tokens from the question which comes from the output of BERT [16], this section represents how the relationship between the two modalities is established.

The inputs taken are represented as, $X\epsilon\mathbb{R}^{m\times n}$ and $Y\epsilon\mathbb{R}^{p\times q}$, these inputs X and Y are shown in Figure 1, in the Inter-Modality Fusion Block section. All the features of X are concatenated with each individual feature of Y, in Equation(1). This concatenation of $X\epsilon\mathbb{R}^{m\times n}$ in Equation(1) happens for every feature of Y one by one. For instance, if X represents the output of Faster R-CNN [10] and Y is the output of BERT [16], for y_1 in Equation(1), all the features of X will be concatenated with Y_1, where Y_1 is the first token of output of BERT [16]. Similar process will happen from y_1 to y_p in Equation(1), where p is the total number of tokens. Then, a function is implemented, $\psi: x\epsilon\mathbb{R}^{m\times(n+q)}$ which defines new set of parameters for each of the concatenation of features taking place independent of the other concatenations, $W\epsilon R^{(n+q)\times n}$ and b, where b is the bias, in Equation (2). In the above mentioned example, each of the concatenation undergoes the ψ function defined in Equation (2).

$$y_k = \psi_k([X, Y_k]) \tag{1}$$

Where,

$$\psi(x) = \tanh(xW + b) \tag{2}$$

The computed y_k in Equation(1), for each of the feature concatenation are then cascaded in Equation(3). From all these cascaded values of y in Equation(3) an average is computed out of it and stored in z, which happens in Equation(4). In the above mentioned example, where the y_1 to y_p is computed in Equation(1), all the values are then cascaded in Equation (3) and their average is computed in Equation (4).

$$y = [y_1, y_2, y_3, \dots y_p] \tag{3}$$

$$z = \frac{1}{m}\sum_{i=1}^{p} y_i \tag{4}$$

z from Equation (4) needs to be brought to a fixed dimension, so that the outputs from both the inter fusion blocks have the similar dimension. s_i represents the result from each block in computed in Equation (5). Θ from Figure 1 performs the Hadamard product, which is nothing but the element-wise product of two matrices. Result from both the blocks, mentioned in Equation(5), are taken and Hadamard product is performed on them, in Equation(6), which is stored in $\overline{Y_1}$. In the above mentioned example, where if X represented the output of Faster R-CNN [10] and Y was the output of BERT [16], it finally computed s_1, now the exact opposite will happen where X would represent the output of BERT [16] and Y would represent the output of Faster R-CNN [10], all the computation would repeat from Equation(1) to Equation(5) and this would finally lead to s_2 in Equation(5). Then, the computed s_1 and s_2 would undergo Hadamard product and store their value in $\overline{Y_1}$ in Equation(6).

$$s_i = ReLU(zW_o + b_o) \tag{5}$$

$$\overline{Y_1} = s_1 \circ s_2 \tag{6}$$

Where, $W_o\epsilon\ \mathbb{R}^{n\times K}$ and b_o are the learned weights and bias respectively.

3.2. Intra-Modality Fusion Block

The intention of this section of the paper is to establish the relationship between the features of the same modality. It takes either the sequence of tokens from the BERT [16] or the objects of the images at a time and sets up an association among themselves.

Inputs to the Intra-Modality Fusion blocks can be shown as $A\epsilon\mathbb{R}^{\alpha\times\beta}$ and each feature of A is made to concatenate with other features, in q_j, in Equation (7). For instance, if A represents the output of Faster R-CNN [10], then in q_1 of Equation (7), all the features of A will be concatenated with the first object of the output of Faster R-CNN [10] and similar process will happen from q_1 to q_α, where α is the total number of objects detected. Similar to what was done in Inter-Modality Fusion, a function is defined $\varphi(x): x\epsilon\mathbb{R}^{\alpha(\beta+\beta)}$, which defines new set of weights $W_r\epsilon\mathbb{R}^{(\beta+\beta)\times\beta}$ and b_r, where b_r is the bias, for each of the concatenation happening, in Equation (8).

$$q_j = \varphi_j([A, A_j]) \tag{7}$$

Where,

$$\varphi(x) = \tanh(xW_r + b_r) \tag{8}$$

For each of the concatenated values passed through φ function, in Equation (8), it is then cascaded in Equation (9), and its average would be computed for the same, in Equation(10). In the above mentioned example, the values q_1 to q_α computed in Equation (7) are cascaded in Equation(9) and their average is found in Equation (10),

$$q = [q_1, q_2, q_3, \dots, q_\alpha] \tag{9}$$

$$z_o = \frac{1}{\alpha}\sum_{i=1}^{\alpha} q_i \tag{10}$$

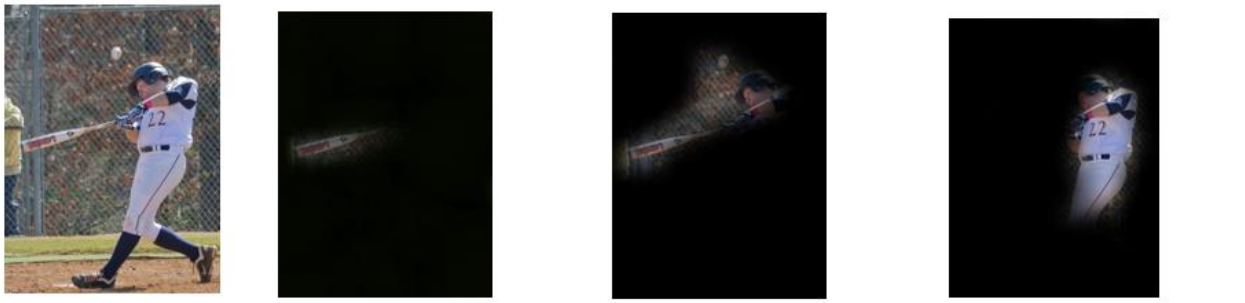

Figure 3. Showing how attention on an image is adjusted based on question.

Finally, z_o from Equation (10) is brought to the same dimension of z from Equation (4) using the parameters $W_e \epsilon \mathbb{R}^{\beta \times K}$ and b_o. Again, result from both the intra fusion blocks are represented in Equation (11) and finally Hadamard Product on both the outputs of Equation (11) are performed on Equation (12). In the above mentioned example, where A represented the output of Faster R-CNN [10], it led to the computation of t_1. The entire similar process will take place from Equation(7) to Equation(12), but this time with A being the output of BERT [16] and this would compute t_2. Then, the Hadamard product of t_1 and t_2 would take place in Equation(12).

$$t_i = ReLU(zW_e + b_e) \tag{11}$$

$$\bar{Y_2} = t_1 \circ t_2 \tag{12}$$

4. SIMULATION RESULT AND DISCUSSION

4.1. Datatsets

DXMN model uses three datasets, VQA v2.0 [24] and GQA [25] and Visual Genome [26] for evaluation. There were considerable number of images that were common across different datasets. These images were not repeated and all the questions from that image across different datasets were combined. Thus, this merged data consists of 180K distinct images and 3M questions. A large number of distinct answers were found in this combined dataset. Only the most frequent 9000 answers were selected which cover almost all the questions from the above mentioned datasets. Partially inspired by LXMERT [17], instead of having a large validation set to avoid overfitting of model, a pruned validation set is used of 5K images and 36K questions and rest of the data is used as training set.

4.2. Image Features

Instead of passing the entire image through convolutional neural networks to extract its spatial features, the image is passed through Faster R-CNN [10]. 36 objects are extracted from each image which are passed through the image.

4.3. Question Embedding

The question is passed through pre-trained BERT [16] bert-base-uncased model where its words are converted into tokens. The length of the number of tokens is set as 20 and each token has a default size of 768. Questions with number of tokens less than 20 are end-padded using the pad token.

4.4. Implementation

The model is implemented using Pytorch [27] and all the initializations are default Pytorch [27] initialization. DXMN is pre-trained on the aggregated dataset till the accuracy on validation set keeps increasing to avoid overfitting. Pre-training on the aggregated dataset happens for 26 epochs. The model uses Stochastic Gradient Descent [28-34] optimizer for training and reducing the loss. For the first 7 epochs the learning rate is set to $1e-3$ and $1e-4$ epoch 8 onwards. To avoid exploding gradients gradient clipping technique is used. Also, Dropout is used in the model [35-43].

4.5. System Requirement

The system that was used for training and inference had Intel Xeon CPU of 2 GHz 16 cores with the GPU of Nvidia Tesla T4. It took nearly 8.5 days for DXMN to train from scratch on the 180K images questions and 3M questions.

Table 1. Accuracy achieved by different models across test-standard of VQA v2.0.

VQA 2.0				
Models	**Yes/No**	**Number**	**Other**	**Overall**
CSS [19]	73.25	39.77	55.11	59.91
LMH [20]	77.85	40.03	55.04	61.64
SCR [23]	77.40	40.90	56.50	62.30
UpDn + MUTANT	82.07	42.52	53.28	62.56

[18]				
Caption VQA [21]	82.6	43.9	56.4	65.8
JE-MHA [22]	84.6	48.3	58.0	66.7
DXMN	85.30	51.04	58.18	68.65

4.6. Qualitative Analysis

In order to understand how the model adjusts its attention Figure 3 can be visualized. The attention that is focused on the image is based on the question being asked is shown in image. The entire attention in the image is focused on the bat, as the question that is asked is related to bat. Similarly, the attention keeps adjusting itself depending on the question asked in the image. This process can be clearly visualized in Figure 3.

Unlike the other older methods, which took the entire image into consideration to focus its attention on depending on the question, here in DXMN, the top 36 objects which are given as output from the Faster R-CNN [10] are taken into account in order to decide the attention. This process increases the accuracy which can be clearly visualized in Figure 3.

4.7. Comparative Analysis

DXMN achieves state-of-the-art result in the VQA v2.0 dataset. The in-depth analysis across various categories of how DXMN performs compared to other models who have achieved state-of-the-art results is shown in Table 1. The technique or the formula used for calculating the accuracy is mentioned in Section 4.8.

4.8. Final result

Table 2. This is the accuracy achieved by DXMN in VQA 2.0 dataset across various categories.

VQA 2.0				
Test Set	**Yes/No**	**Number**	**Other**	**Overall**
Test-Standard	85.30	51.04	58.18	68.65
Test-Dev	85.13	50.88	58.24	68.43

DXMN was able to achieve an accuracy of 68.43 in test-dev and 68.65 in test-standard of the VQA v2.0 dataset after training it on the datasets mentioned in section 4. The result is explained in detail in Table 2.

Yes/No, Number and Other are the different categories of answers existing in VQA 2.0 dataset and the respective accuracies achieved across those categories. The accuracy is calculated using the following the formula:

$$\text{Acc(ans)} = \min\left\{\frac{\#\text{humans that said ans}}{3}, 1\right\} \quad (13)$$

5. CONCLUSION

In this paper a novel architecture is proposed for visual question answering. The main idea proposed is to take the relationship between same modality into consideration along with relationship between same modality into consideration along with relationship between different modalities. This model achieves state-of-the-art result for VQA v2 dataset. Also, this paper introduces novel method of learning the features in the modalities. It is believed that there is room of improvement in this field and this paper might contribute to enhance researches happening in future.

AUTHORS' CONTRIBUTIONS

Rishav Bhardwaj has done the entire project, right from coming with this novel approach till bringing it into implementation.

ACKNOWLEDGMENTS

Would like to thank Jayanthi Singh and Geetha P, Computer Science Department at Cambridge Institute of Technology for guiding through this project.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou and K. Q.Weinberger, editors, Advances in Neural Information Processing Systems (2012).

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (2016).

[3] Volodymyr Mnih, Nicolas Heess, Alex Graves, Koray Kavukcuoglu. Recurrent Models of Visual Attention. In: Advances in neural information processing systems (2014).

[4] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In: International Conference on Learning Representations (2015).

[5] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In: Advances in neural information processing systems (2018).

[6] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Empirical Methods in Natural Language Processing (2014).

[7] S. Hochreiter and J. Schmidhuber. Long short-term memory. In: Neural Computation (1997).

[8] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2018).

[9] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In: IEEE Conference on Computer Vision and Pattern Recognition (2018).

[10] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems (2019).

[11] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell. Compact bilinear pooling. In: IEEE Conference on Computer Vision and Pattern Recognition (2018).

[12] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (2019).

[13] H. Ben-younes, R. Cadene, M. Cord, and N. Thome. MUTAN: Multimodal Tucker Fusion for Visual Question Answering. In IEEE International Conference on Computer Vision (2018).

[14] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering. In: International Conference on Neural Information Processing Systems (2016).

[15] D. Yu, J. Fu, Y. Rui, and T. Mei. Multi-level attention networks for visual question answering. In: International Conference on Computer Vision and Pattern Recognition (2017).

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova: Bert: Pre-training of deep bidirectional transformers for language understanding. In: North American Chapter of the Association for Computational Linguistics HLT (2019).

[17] Hao Tan, Mohit Bansal. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In: Empirical Methods in Natural Language Processing (2019).

[18] Tejas Gokhale, Pratyay Banerjee, Chitta Baral and Yezhou Yang. MUTANT: A Training Paradigm for Out-of-Distribution Generalization in Visual Question Answering. In: Proceedings of the 2020 Conference on Emperical Methods in Natural Language Processing, (2020)

[19] Long Chen, Xin Yan, Jun Xiao, Hanwang Zhang, Shiliang Pu, and Yueting Zhuang. Counterfactual samples synthesizing for robust visual question answering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020).

[20] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In Emperical Methods in Natural Language Processing, (2020)

[21] J. Wu, Z. Hu, R. Mooney, Generating question relevant captions to aid visual question answering, In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (2020).

[22] K. Yu, L. Tan, S. Mumtaz, S. Al-Rubaye, A. Al-Dulaimi, A. K. Bashir, F. A. Khan, "Securing Critical Infrastructures: Deep Learning-based Threat Detection in the IIoT", IEEE Communications Magazine, 2021.

[23] K. Yu, Z. Guo, Y. Shen, W. Wang, J. C. Lin, T. Sato, "Secure Artificial Intelligence of Things for Implicit Group Recommendations", IEEE Internet of Things Journal, 2021, doi: 10.1109/JIOT.2021.3079574.

[24] H. Li, K. Yu, B. Liu, C. Feng, Z. Qin and G. Srivastava, "An Efficient Ciphertext-Policy Weighted Attribute-Based Encryption for the Internet of Health Things," IEEE Journal of Biomedical and Health Informatics, 2021, doi: 10.1109/JBHI.2021.3075995.

[25] L. Zhen, A. K. Bashir, K. Yu, Y. D. Al-Otaibi, C. H. Foh, and P. Xiao, "Energy-Efficient Random Access for LEO Satellite-Assisted 6G Internet of Remote Things", IEEE Internet of Things Journal, doi: 10.1109/JIOT.2020.3030856.

[26] L. Zhen, Y. Zhang, K. Yu, N. Kumar, A. Barnawi and Y. Xie, "Early Collision Detection for Massive Random Access in Satellite-Based Internet of Things," IEEE Transactions on Vehicular Technology, vol. 70, no. 5, pp. 5184-5189, May 2021, doi: 10.1109/TVT.2021.3076015.

[27] L. Tan, K. Yu, A. K. Bashir, X. Cheng, F. Ming, L. Zhao, X. Zhou, "Towards Real-time and Efficient Cardiovascular Monitoring for COVID-19 Patients by 5G-Enabled Wearable Medical Devices: A Deep Learning Approach", Neural Computing and Applications, 2021, https://doi.org/10.1007/s00521-021-06219-9

[28] Jung-Jun Kim, Dong-Gyu Lee, Jialin Wu, Hong-Gyu Jung and Seong-Whan Lee. Visual Question Answering based on Local-Scene-Aware Referring Expression Generation. In: Neural Networks Volume 139 (2021).

[29] Jialin Wu and Raymond J Mooney. Self-critical reasoning for robust visual question answering. In NeurIPS (2019).

[30] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh: Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In: IEEE Conference on Computer Vision and Pattern Recognition (2017).

[31] Drew A Hudson and Christopher D Mannin: Gqa: A new dataset for compositional question answering over real-world images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019).

[32] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al: Visual genome: Connecting language and vision using crowdsourced dense image annotations. In: International Journal of Computer Vision (2017).

[33] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. De-Vito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch (2017).

[34] Llya Sutskever, James Martens, George Dahl and Geoffrey Hinton. On the importance of initialization and momentum in Deep Learning. In: Proceedings of the 30th Interenation Conference of Machine Learning (2013).

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. In: The Journal of Machine Learning Research (2014).

[36] Puttamadappa, C., and B. D. Parameshachari. "Demand side management of small scale loads in a smart grid using glow-worm swarm optimization technique." *Microprocessors and Microsystems* 71 (2019): 102886.

[37] Parameshachari, B. D., H. T. Panduranga, and Silvia liberata Ullo. "Analysis and computation of encryption technique to enhance security of medical images." In *IOP Conference Series: Materials Science and Engineering*, vol. 925, no. 1, p. 012028. IOP Publishing, 2020.

[38] Rajendran, Ganesh B., Uma M. Kumarasamy, Chiara Zarro, Parameshachari B. Divakarachari, and Silvia L. Ullo. "Land-use and land-cover classification using a human group-based particle swarm optimization algorithm with an LSTM Classifier on hybrid pre-processing remote-sensing images." *Remote Sensing* 12, no. 24 (2020): 4135.

[39] Hu, Liwen, Ngoc-Tu Nguyen, Wenjin Tao, Ming C. Leu, Xiaoqing Frank Liu, Md Rakib Shahriar, and SM Nahian Al Sunny. "Modeling of cloud-based digital twins for smart manufacturing with MT connect." *Procedia manufacturing* 26 (2018): 1193-1203.

[40] Bhuvaneswary, N., S. Prabu, K. Tamilselvan, and K. G. Parthiban. "Efficient Implementation of Multiply Accumulate Operation Unit Using an Interlaced Partition Multiplier." *Journal of Computational and Theoretical Nanoscience* 18, no. 4 (2021): 1321-1326.

[41] Bhuvaneswary, N., S. Prabu, S. Karthikeyan, R. Kathirvel, and T. Saraswathi. "Low Power Reversible Parallel and Serial Binary Adder/Subtractor." *Further Advances in Internet of Things in Biomedical and Cyber Physical Systems* (2021): 151.

[42] Seyhan, Kübra, Tu N. Nguyen, Sedat Akleylek, Korhan Cengiz, and SK Hafizul Islam. "Bi-GISIS KE: Modified key exchange protocol with reusable keys for IoT security." *Journal of Information Security and Applications* 58 (2021): 102788.

[43] Nguyen, Tu N., Bing-Hong Liu, Nam P. Nguyen, and Jung-Te Chou. "Cyber security of smart grid: attacks and defenses." In *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, pp. 1-6. IEEE, 2020.