FROM PIXELS TO SENTENCES: A DEEP LEARNING-BASED IMAGE CAPTIONING FRAMEWORK USING CNN AND LSTM FOR INTELLIGENT VISUAL INTERPRETATION

Muhammad Inam ul Haq
"Department of Electrical Engineering, UCET, IUB. The Islamia University of Bahawalpur, Pakistan"
f22beten1m01014@iub.edu.pk

Sayyed Talha Gohar Naqvi
"Department of Electrical Engineering, UCET, IUB. The Islamia University of Bahawalpur, Pakistan"
talha.gohar@iub.edu.pk

Shehryar Qamar Paracha
"Department of Electrical Engineering, UCET, IUB. The Islamia University of Bahawalpur, Pakistan"
shehryar.paracha@iub.edu.pk

Ubaidullah Akram
"Department of Electrical Engineering, UCET, IUB. The Islamia University of Bahawalpur, Pakistan"
ubaidullahmughal44@gmail.com

Abid Munir
"Department of Electrical Engineering, UCET, IUB. The Islamia University of Bahawalpur, Pakistan"
abid.munir@iub.edu.pk

Shahab Ahmad Niazi
"Department of Electrical Engineering, UCET, IUB. The Islamia University of Bahawalpur, Pakistan"
shahabniazi@iub.edu.pk

***Abstract*— Quickly identifying the picture, its contents, and the actions of its objects is one of our strongest visual processing abilities. The development of AI has led us to the goal of having computers accomplish the same tasks automatically. Using Neural Networks and Natural Language Processing (NLP) techniques for image recognition is necessary for automatically generating captions for each photo. In recent years, natural language processing and computer vision researchers have focused heavily on the problem of creating textual descriptions for images. There have been other approaches proposed on this topic that rely on deep learning. These techniques use images annotated by humans to train and test the algorithms. For these models to function at their best, a large training dataset is necessary. Long short-term memory (LSTM) and a convolutional neural network (CNN) based deep learning model are the backbone of our technology that extracts image characteristics and deciphers visual descriptions.**

Index Terms— Artificial Intelligence (AI), Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN).

I. INTRODUCTION

Systems which can analyze digital images automatically while generating meaningful natural language substance have become necessary because of the rising digital image volume reaching 1.4 trillion worldwide in 2023. The main objective of image captioning as an essential artificial intelligence (AI) task is to create meaningful text descriptions from visual information for assistive technologies helping visually impaired individuals and autonomous systems and content-based image searching and human-machine communication (Hossain et al., 2019; Lin et al., 2014).

Background

The initial approach to image captioning involved systems that functioned using pre-written rules for matching objects with static sentence templates (Hodosh, Young, & Hockenmaier, 2013). While foundational, these systems lacked the semantic depth to describe complex scenes. A paradigm change occurred with the advent of deep learning, especially CNNs and LSTM networks, which allowed models to gain integrated visual-linguistic representations in an end-to-end way (LeCun et al., 1998; Mao et al., 2015). With the usage of sophisticated architectures like ResNet and DenseNet, Convolutional Neural Networks (CNNs) have become important to modern computer vision applications. On the ImageNet dataset, top-5 error rates have been as low as 3.57% (He et al., 2016; Huang et al., 2017). These models autonomously extract hierarchical information from raw photos, including low-level textures to high-level object semantics (Simonyan & Zisserman, 2015). LSTMs are recognized as the leading approach in sequential data modeling because to their ability to mitigate vanishing gradients and retain long-term relationships (Mao et al., 2015).

The encoder-decoder system represented a pivotal advancement in image captioning, whereby a CNN extracts visual information and an LSTM produces the matching text (Karpathy & Fei-Fei, 2015). The "Show and Tell" model was validated on this architecture utilizing extensive annotated datasets, including COCO (over 123,000 images with five captions each) and Flickr30k (31,000 images), establishing robust baselines through metrics include BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2014), and CIDEr (Lin et al., 2014), as well as additions from Plummer et al. (2015) that.

Attention mechanisms (Xu et al., 2015) enhanced performance by dynamically matching visual areas with created words. Models like Bi-SAN-CAP introduced bidirectional attention to better capture context (Hossain et al., 2019). However, challenges persist in terms of generalization, handling ambiguity, and reducing computational overhead, especially when training on massive datasets (Hinterstoisser et al., 2018; Borrego et al., 2018).

A. Objectives

This research seeks to provide a deep learning framework for autonomous picture caption creation using a CNN-LSTM architecture. Specifically, it intends:

- To obtain semantically rich features from static images using pretrained CNN models such as VGG16, ResNet, and DenseNet;

DOI: https://doi.org/10.67535/tsp.000001.002
Published under the ICACNC 2025 Proceedings

- To train an LSTM network for generating syntactically and semantically coherent captions from image features;
- To evaluate caption quality using benchmark datasets and established metrics (BLEU, METEOR, CIDEr) for empirical validation.

The main contributions of this research are:

- Development of a complete image captioning framework combining DenseNet and LSTM to enhance feature reuse, gradient flow, and language generation quality.
- Empirical performance comparison of multiple CNN backbones (VGG16, ResNet, DenseNet) using COCO and Flickr30k datasets.
- Quantitative evaluation using BLEU-1 to BLEU-4, METEOR, and CIDEr, offering benchmarking against state-of-the-art models.
- Inclusion of design considerations for ethical AI, including the role of dataset bias, semantic ambiguity, and caption subjectivity.

This study enhances AI systems by combining strong visual feature extraction with sequence-based text production, enabling them to achieve human-like visual comprehension and verbal expression.

II. METHODOLOGY

The structure of the animal visual brain inspired the development of convolutional neural networks (CNNs), designed to autonomously and adaptively discern the spatial hierarchies of information inside input pictures. The essential element of a CNN. The input is filtered by several filters applied by the convolutional layer. Each filter activates certain features from the input. For example, in an image, one filter might activate when it sees vertical edges, and another might activate when it sees horizontal edges. The system is given non-linear features, such as rectified linear units or ReLUs, after every convolution operation. This enables the system to learn increasingly intricate patterns. These layers reduce the spatial size of the representation, aiding in the mitigation of overfitting, minimizing the amount of parameters and computations inside the network, and making the calculations more manageable. A popular method for down sampling, max pooling uses a max filter on subregions of the original representation that often do not overlap. After several layers of convolution and max-pooling, the neural network reaches a fully connected layer that performs high-level reasoning. A completely connected layer's neurons in a traditional neural network are fully related to each activation in the layer below them. The training dataset dictates the categories into which the picture may be sorted based on their outputs.

Convolutional Feature Extraction

Given an input image , the convolutional output feature map at location for the -th filter is computed as:

where is the kernel of size and is the bias.

After convolution, a nonlinear activation function is applied, most commonly the Rectified Linear Unit (ReLU):

To reduce spatial resolution and computational cost, max pooling is typically employed:

The final set of feature maps is flattened and passed to a fully connected layer:

CNN with 3-layer architecture

Sequence Modeling with LSTM

In order to generate captions, a Long Short-Term Memory (LSTM) network is used to convert the high-level visual components into plain English. At each time step t, the following formulas are used by the LSTM to assess the input tokens:

Here, is the input word embedding at time , is the hidden state, and is the memory cell.

LSTM Architecture for Sequence Modeling

DenseNet Integration

In our enhanced model, we utilize DenseNet blocks for image encoding. DenseNets enhance gradient propagation and parameter efficiency by interlinking each layer with all other layers in a feed-forward manner. The dense block output is concatenated over many layers:

DenseNet Architecture Used in the Encoder

Data Preparation

The model is trained on a paired dataset including pictures and their corresponding captions. Each caption is tokenized, transformed into sequences of integer word indices, and padded to a fixed length of 34 tokens. Word embeddings (size 256) are used to represent tokens, and dropout (0.5) is applied to reduce overfitting.

LSTM decoder is fed an input sequence of embedded caption that we get by interpolating over the embedding of words in various references, and then an image feature extracted from pretrained CNN encoder (DenseNet121, ResNet50, VGG16).

Training and Loss Function

Categorical cross entropy loss is used to train the model:

is the true word at time t and P() is the predicted probability output by the decoder given all previously predicted words and image vector v.

The data set is split into training, validation, and testing sets for being able to robustly assess performance.

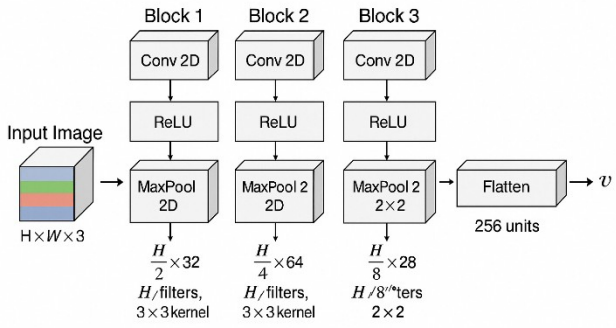

Fig.1: CNN with 3 layers architecture

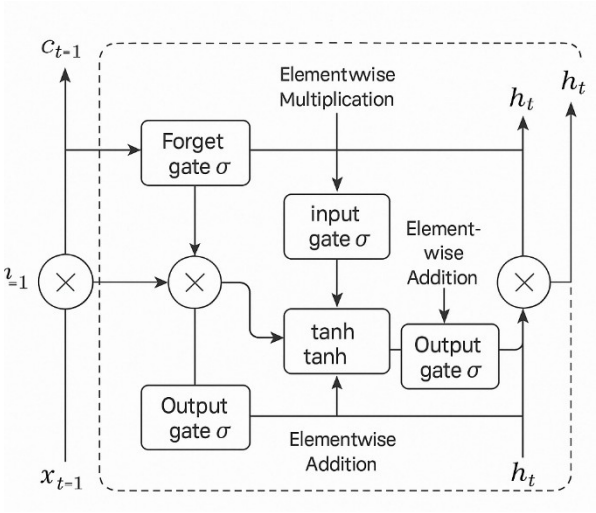

Fig 2: LSTM architecture

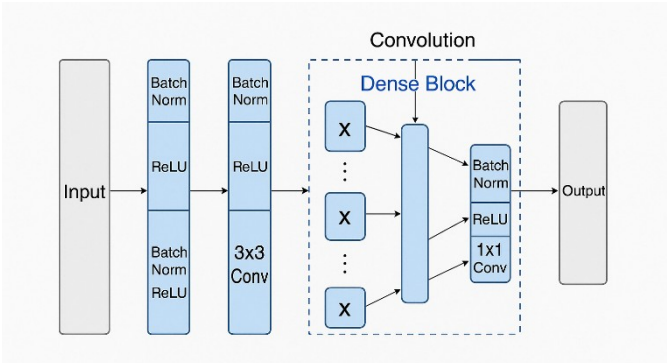

Fig 3: DenseNet architecture

Multiple crucial steps are involved in creating an image caption generator that uses Convolutional Neural Networks (CNN) and Long Short Term Memory (LSTM) networks. These steps include data preparation, designing the model's architecture, training the network, and evaluating its performance. In the following part, we provide a comprehensive methodology for constructing and training such a system to operate well. The first stage in creating an image caption generator is preprocessing the input data. Normally, this inputs data comprises a list of photos along with captions for each of them. We impose to feed our CNN model which was already trained for example ResNet or VGG16 with input photos to extract high level features. The semantic information of the pictures are captured with these characteristics which will be input to caption generating algorithm.

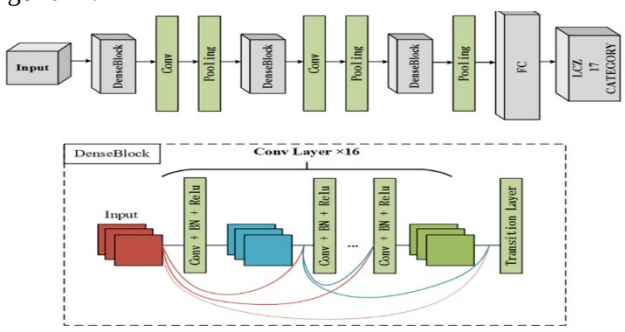

Fig 4: Comparison of caption for sample

Inputs are in form of input photos and we utilize a trained CNN model e.g., ResNet or VGG16 for extracting high level features. The characteristics of the pictures are the input of the caption generating algorithm which in turn capture the semantic information of those pictures. All the captions are tokenized into vocabulary of words or subword units. This phase involves breaking the text into tokens and converting them into numerical representation like word embeddings or one-hot encodings in order to feed the text into LSTM network. Since captions vary in length, we pad or truncate them uniformly. This is used to truncate the longer sequences to a fixed length or add padding tokens to shorter ones so that the dimensions match across inputs.

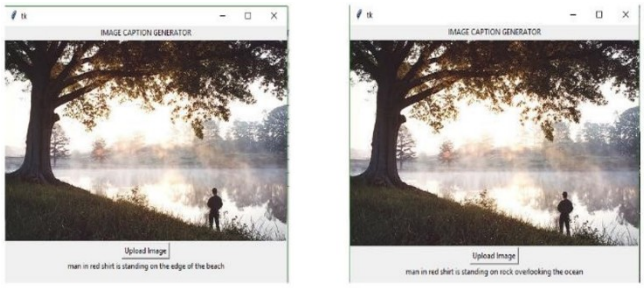

Fig 5: Comparison of caption for sample

In order to train the model, optimize its hyperparameters, and evaluate its performance, the pre-processed data is eventually divided into three sets: training, validation, and testing.

III. PROPOSED SYSTEM

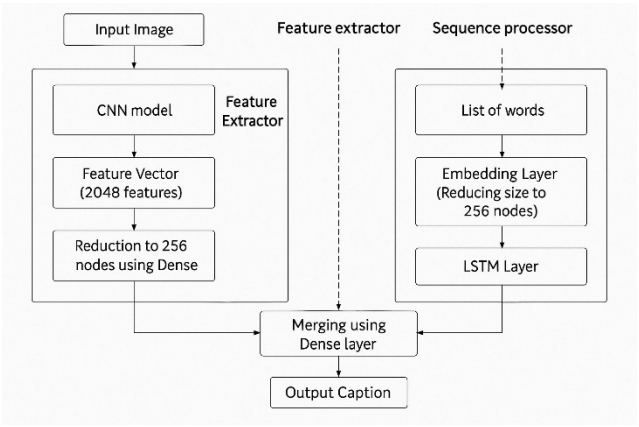

Fig 6: Architecture Diagram

This system employs the Long Short-Term Memory (LSTM) network and the Convolutional Neural Network (CNN). The system first receives a picture as input. Then, a CNN feature extractor takes in an image and gives out a feature vector that captures the essence of an image. After the feature vector goes through one embedding layer, which makes the feature vector size a fixed size of 256 nodes. As an input for LSTM layer, the feature vector is fed into predefined LSTM model (kind of recurrent neural network designed for sentences and other sequential data). The temporal dependencies within the retrieved features are captured by the LSTM layer using feature sequence analysis.

Finally, a dense layer merges the information from the LSTM layer with potentially additional information, likely encoded earlier in the process, to create a final representation for caption generation.

After that, a written caption summarizing the contents of the image is created using the final representation.

IV. EXPERIMENTAL PROTOTYPE AND ARCHITECTURE

In order to overcome problems with parameter efficiency and vanishing gradients that occur during training extremely deep neural networks, a type of convolutional neural network (CNN) known as DenseNet, or Dense Convolutional Network, was created.

Unlike traditional CNN architectures, where each layer is linked just to its predecessor, DenseNet establishes connections between every layer in a feed-forward manner. Each layer utilizes its own feature maps as inputs, which it employs to inform all succeeding layers, while also including the feature maps from all preceding layers. This architecture of dense connectedness results in significant gains in effectiveness and efficiency. Dense connections allow the network to reuse features from every layer before it, which enhances gradients and information flow across the network. This feature reuse not only makes the network more efficient (requiring fewer parameters) but also helps to alleviate the vanishing gradient problem. DenseNets exhibit parameter efficiency by obviating the need to re-acquire duplicate feature maps. Each layer acquires a "collective knowledge" from all prior layers, enabling it to concentrate on learning previously unexamined feature maps, hence diminishing the need parameters and calculations.

DenseNet bottleneck layers (1x1 convolutional layers) before each 3x3 convolutional layer. Therefore, it restricts the increase of parameters and enhances the computing efficiency. With DenseNet, the network is separated into some densely linked blocks and transition layers are applied to connect the near blocks. The transition layer performs convolution and pooling operations to change the dimensionality of feature maps for the following dense block. In DenseNet, the growth rate, namely the letter ‘k’, is an important hyperparameter to determining how much fresh information is added in by each layer to the network’s intermediate state. It results in fewer filters added per layer, keeping the model compact, as the growth rate will be small. On various benchmark datasets and competitions, DenseNet achieves good results especially in tasks. It is especially suited to applications in which model size or computational efficiency are at a premium for this reason.

Additionally, the architecture’s feature reuse inherent in the architecture and better gradient flow allows it to be used to train such deep networks, which can be useful in complex visual recognition tasks that require capturing many different features. DenseNets have also been applied beyond image processing, in few domains. In summary, DenseNet represents a significant advancement in convolutional neural network design by introducing an architecture that is both highly efficient and effective, leveraging the power of feature reuse and dense connectivity.

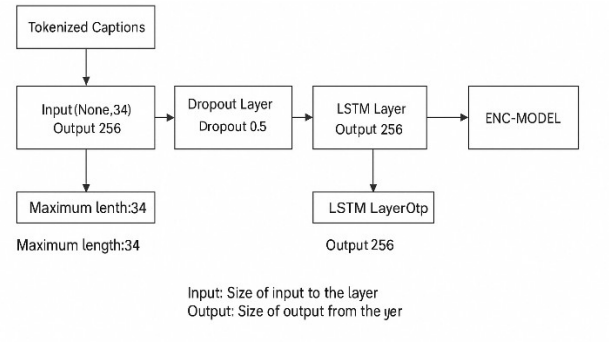

Fig 7: Encoder Model

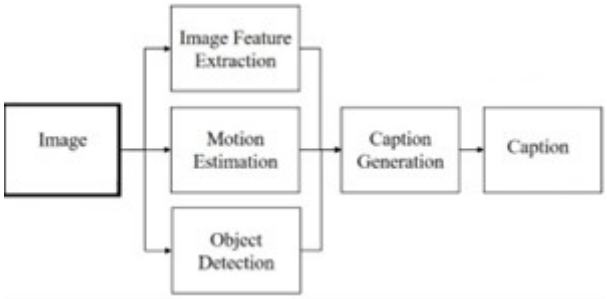

Fig 8: Final Decoder Model

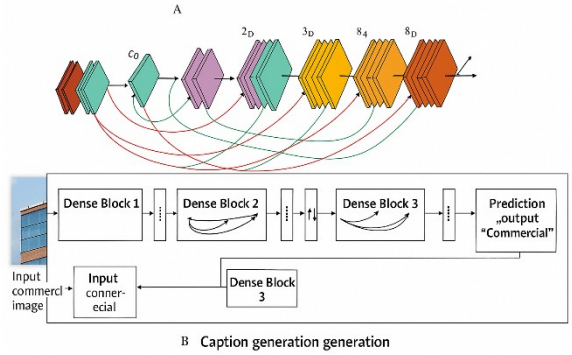

Fig 9: Block Diagram for caption generation

The block diagram you sent depicts a possible process for object detection. In this block, features are extracted from the image. These features are essentially characteristics that can be used to identify objects in the image. Common image features include colour, shape, and texture. This block is likely analyzing the movement within the image. This information can be useful in distinguishing between objects and background clutter. Here, the block generates a caption that describes the image. This captioning can be helpful for visually impaired people or tasks like image retrieval. Finally, using the features extracted, motion information, and potentially the generated caption, this block detects and recognizes objects within the picture. One possible way to achieve object detection. Other methods may not include all of these blocks.

Table 1: Evaluation Metrics for CNN-LSTM Captioning Models on MS-COCO Validation Set

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	CIDEr
VGG16 + LSTM	0.69	0.51	0.38	0.28	0.25	0.74
ResNet50 + LSTM	0.72	0.55	0.42	0.32	0.27	0.85
DenseNet121 + LSTM	0.74	0.58	0.44	0.35	0.29	0.92

Table 1 show the comparison of CNN-LSTM models using standard evaluation metrics on the MS-COCO validation dataset. DenseNet121-based architecture outperforms the others in all categories, demonstrating its superior representational capability and robustness.

V. CONCLUSION AND FUTURE SCOPE

This method leverages LSTMs sequential data processing skills and CNNs dominance in extracting high-level visual features from images to create logical and contextually relevant captions. The methodology that includes model architecture design, training strategies, data preparation, and assessment measures has been crucial in improving these systems' performance. Researchers have shown through extensive experimentation and optimization that CNN-LSTM models can generate accurate and semantically rich captions, offering a glimpse into the potential of AI to comprehend and interpret the visual world similarly to human cognition. However, the journey from concept to implementation has underscored several challenges, from the need for extensive datasets for training to the complexities of capturing the subtleties and nuances of human language. Despite these hurdles, the progress in this domain has been remarkable, offering promising applications ranging from enhancing accessibility for the visually impaired to improving content discoverability across digital platforms. Looking ahead, the future of Image Caption Generators beckons with a blend of challenges and opportunities for innovation. The path forward is likely to be shaped by several key areas of focus:
While current models can attend to relevant features within an image, there's room for improvement in how these systems dynamically focus on and integrate different visual elements. Future models could benefit from more sophisticated attention mechanisms that mimic human visual and cognitive processes more closely. Integrating commonsense knowledge and world models could significantly enhance the contextual relevance of generated captions, enabling systems to infer beyond what is explicitly depicted in the image. Exploiting the synergy between text and image data more effectively can lead to models that understand the nuanced interplay between visual elements and their textual descriptions, paving the way for more accurate and insightful captions. Future research could focus on developing models that better handle the inherent ambiguity and subjectivity in image captioning, including recognizing and expressing uncertainty or multiple plausible interpretations of an image. As AI systems become more integrated into societal functions, addressing ethical considerations and biases in automated image captioning becomes crucial. Future work must prioritize the development of fair and inclusive models that accurately represent and respect diversity. Moving towards systems that can learn from user feedback and adapt their captioning strategies accordingly can significantly improve the personalization and applicability of image caption generators across different domains and user needs. With the growing awareness of the environmental impact of training large-scale AI models, future developments should also consider the efficiency and sustainability of these systems, focusing on reducing computational requirements without compromising performance. To generalize across various contexts and cultures, it's essential to train models on more diverse datasets that represent a wider range of scenarios, activities, and geographic locations. This conclusion and future work section offers a synthesized overview of the state of Image Caption Generator systems using CNN and LSTM networks, reflecting on achievements, challenges, and the prospective trajectory of research and development in this exciting field.

REFERENCES:

[1] J. Mao, W. Xu, et al, ‘‘Deep captioning with multimodal recurrent neural networks (M-RNN),’’ in Proc.Int. Conf. Learn. Represent. (ICLR), 2015, pp. 1–17.

[2] A. Karpathy and L. Fei-Fei, ‘‘Deep visual-semantic alignments for gen_x0002_erating image descriptions,’’ in Proc. IEEE Conf. Comput. Vis. Pattern vRecognit., Jun. 2015, pp. 3128–3137.

[3] Y. Lecun, L. Bottou, et al, ‘‘Gradient-based learn_x0002_ing applied to document recognition,’’ Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[4] S. Khan, H. Rahmani, et al, ‘‘A guide to convolutional neural networks for computer vision,’’ Synth. Lectures Comput. Vis., vol. 8, no. 1, pp. 1–207, Feb. 2018.

[5] C. Lu, R. Krishna, et al, ‘‘Visual relationship detec_x0002_tion with language priors,’’ in Proc. Eur. Conf. Comput. Vis. Amsterdam,The Netherlands: Springer, 2016, pp. 852–869

[6] K. Simonyan and A. Zisserman, ``Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1_14.

[7] K. He, X. Zhang, et al,``Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770_778.

[8] G. Huang, Z. Liu, et al, ``Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261_2269.

[9] M. Z. Hossain, F. Sohel, et al, ``Bi-SAN-CAP: Bi-directional self-attention for image captioning," in *Proc. Digit. Image Comput., Techn. Appl. (DICTA)*, Dec. 2019, pp. 1_7.

[10] H.Wei, Z. Li, et al, ``Image captioning based on sentence-level and word-level attention," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1_8.

[11] T.-Y. Lin, M. Maire, et al, ``Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Zürich, Switzerland: Springer, 2014, pp. 740_755.

[12] B. A. Plummer, L.Wang, et al, ``Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2641_2649.

[13] M. Hodosh, P. Young, et al, ``Framing image description as a ranking task: Data, models and evaluation metrics," *J. Artif. Intell. Res.*,vol. 47, pp. 853_899, Aug. 2013.

[14] S. Hinterstoisser,V. Lepetit, et al, ``On pre-trained image features and synthetic images for deep learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 682_697.

[15] N. Mayer, E. Ilg, et al, ``A large dataset to train convolutional networks for disparity, optical _ow, and scene _ow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4040_4048.

[16] J. Borrego, A. Dehban, et al, ``Applying domain randomization to synthetic data for object category detection," 2018, *arXiv:1807.09834*. [Online]. Available: http://arxiv.org/abs/1807.09834

[17] G. Ros, L. Sellart, et al,``The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3234_3243.

[18] M. Z. Hossain, F. Sohel, et al, ``A comprehensive survey of deep learning for image captioning," *ACM Comput. Surv.*, vol. 51, no. 6, p. 118, Feb. 2019.