

Received November 13, 2021, accepted November 25, 2021, date of publication November 30, 2021, date of current version December 10, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3131343

Image Captioning With Positional and Geometrical Semantics

ANWAR UL HAQUE¹, SAYEED GHANI¹, AND MUHAMMAD SAEED²

¹Institute of Business Administration, Karachi 75270, Pakistan

²Department of Computer Science, University of Karachi, Karachi 75270, Pakistan

Corresponding author: Anwar ul Haque (ahaque@iba.edu.pk)

ABSTRACT The last 5 to 6 years have seen tremendous progress in automatic image captioning using deep learning. Initial research focused on the attribute-to-attribute comparison of image features and texts to describe the image as a sentence, the current research is handling issues related to semantics and correlations. However, current state of art research suffers from insufficient concepts when it comes to positional and geometrical attributes. The majority of research relying on CNN's (Convolutional Neural Networks) for object feature extractions has no clue about equivariance and rotational invariance which leads towards the orientation-less understanding of objects for captioning along with longer training time, and larger dataset. Furthermore, CNN's based image captioning encoders also fail to understand the geometrical alignment of object attributes within the image and hence mislabels distorted as correct. To cater to the above issues, we propose ICPS (*Image Captioning with Positional and Geometrical semantics*) a capsule network-based image captioning technique along with transformer neural networks as the decoder. The proposed ICPS architecture handles various geometrical properties of image objects with the help of parallelized capsules while the object-to-text decoding is done by Transformer Neural Networks. The inclusion of cluster capsules provides better object understanding in terms of position, equivariance, and geometrical orientation with more augmented object understanding over a small dataset in comparatively less time. The extracted image features provide a better understanding of image objects and help the decoding stage to narrate effectively with positional and geometrical details. We trained and tested our ICPS over the Flickr8k dataset and found ourselves to be better at captioning when it comes to describing the positional and geometrical transitions as compared to other current state-of-the-art research.

INDEX TERMS Capsule networks, deep learning, image captioning, transformer neural networks.

I. INTRODUCTION

The field of computer vision is experiencing new challenges and ventures on a day-to-day basis both in academia and in industry. These challenges are targeting various segments of human life at a scale. Among them, Image captioning is a burning and hot topic. Image captioning is just like modeling the human behavior of seeing and describing. Having a robust and human-like image captioning model would ensure that the machine understanding of the scene is just like us which will be a brave step towards a mature and effective artificial intelligence. Various researchers around the world are trying to create meaning full and comprehensive image-to-text generation with the help of deep learning. However,

it requires global content understanding within the images and efficient state-of-the-art natural language modeling techniques. Both are challenging within their domains. The general approach adopted by researchers is to have an encoder-decoder architecture: the encoding job is done with the help of convolutional neural networks while the decoding role is played by either recurrent neural networks or its variants such as LSTM/GRUs [1]–[3]. The inherent capabilities of convolutional neural networks have been promising for image captioning encoders and have given satisfactory results while the natural language capability of recurrent neural networks has been effective in performing the job of decoding. However, the encoding module suffers from covariance and geometrical understanding while the decoding part suffers from simultaneous visual representation mapping and language model learning.

The associate editor coordinating the review of this manuscript and approving it for publication was Charalambos Poullis¹.

These have been major challenges towards making a more human-like image captioning model till today to the best of our knowledge.

There has been much research available to make a robust and human-like image captioning with the help of variants of convolution neural networks and recurrent neural networks. These works have been successful and somehow significant [4], [5]. Considerable progress in the field of image captioning has been made due to the inclusion of semantic conceptualization over image and image objects. The semantic understanding somehow reflects the human cognition behavior and makes the captioning more human-like [5], [6]. The semantic understanding of objects has enabled the encoding stage to provide better multi-label classification support towards the decoding stage and helps the decoder to mine better language concepts from the training data. The detected concepts allow the decoder to generate superior captions especially working on the test data coming from the same training dataset.

However, we believe that the existing state-of-the-art works providing semantical conceptualization suffer from positional and orientational details. This causes the entire captioning meaningless in terms of activity understanding in the image of objects. An example of our argument is given in Figure. 1.



FIGURE 1. An image from flicker8k dataset.

Figure 1. is an image captioned by the various state-of-the-art works in the domain of image captioning. A few captions are “Dog swims in the water,” “Dog standing in water,” etc. Mentioned captions are great in terms of the detail of feature to text translation, however, lack positional and orientational details. We propose a technique termed ICPS in this paper, our ICPS algorithm captions the same image as “Dog standing near the water” which indicates the relative position of the dog from the perspective of the water in the image. The reason for losing the positional and orientational semantics into the captioning work is due to the training captions used for training these models. Generally, training captions are acquired by conducting random captioning by humans which

tries to describe the objects and their activity within the image instead of going through the pain to describe the positional and orientational details. The annotation data having no positional and orientational details is used to train the model. This training leads to an insufficient understanding of orientations and positions. To cater to the issue either we can re-generate annotation with required information by investing time and money or we can leverage the orientational and geometrical capabilities of capsule network-based feature extraction in the encoder stage. A capsule network is composed of capsules that store information in the form of vectors instead of a scalar. The vector storage helps to store a feature along with its orientation or angle. The standard capsule network uses the concept of routing by agreement to pass the information from the previous layer to the next layer as a replacement for the pooling layer. The combined magnitude and orientation information passed to the next layer helps to understand the affine transformation and geometrical behavior of the object within the image. Despite the effectiveness of capsule network and out-of-box performance over MNIST dataset as presented in [32], the major drawback is the training time when complex data is fed to the model. The routing by agreement algorithm takes a significant amount of time when the network parameters are increased as compared to any available convolutional neural network variants. Running capsule network over Flicker8k dataset also possesses the same challenge of training time exponentiation and large parameters networks which are not tractable and tangible for research. To cater to this issue, we have used a fusing technique over parametric information coming from parallelized capsules. The fusing behavior allows the parameters to be reduced for tractable processing while the parallelization enables the network to learn for more depth over the dataset. The ICPS encoder architecture is composed of stack convolutional neural networks [7]–[9] along with skip connections to provide better convolution over the input images; in addition, the parallelized capsules handle various details for example object magnitude, orientation, spatial information, etc. The parallelization significantly increases the performance and depth of understanding along with the reduction in parameters at the fusing point. We tested out our ICPS over flicker8k [10] dataset and found it quite interesting and comparable to the current state-of-the-art research work in image captioning. The key milestones which we achieved in our research work are:

- An innovative Encoder network capable of learning the spatial, geometrical, and orientational details due to parallelized capsules network with improved performance for image captioning.
- Improved baseline capsule network with more complex datasets training in a tractable amount of time for image captioning.

The rest of the paper is organized as follows. Section 2 discusses the related work and existing issues; Section 3 provides details of our methodology while section 4 provides descriptions of our experimentation, testing,

and evaluation. Section 5 provides a conclusion and future research direction.

II. RELATED WORK

Most of the latest research work in the domain of image captioning uses the fundamental architecture known as encoder-decoder architecture or framework. The framework executes in the way that the image to feature extraction is done using the encoder module which can be constructed in several ways with various deep learning networks e.g., convolutional neural networks, auto-encoders, GAN, transformers, capsule networks, etc. The job of the decoder module is to map the features with the provided annotations during training to learn the conversion of features into human language. Once trained the network can humanly annotate a randomly provided image with various details.

Basic encoder-decoder architecture is composed of CNN (as encoder) and RNN (Recurrent Neural Networks) (as a decoder). The image is fed to CNN for feature conversion while features are fed to RNN for mapping against the annotation words [12]–[15]. To make the network more innovative and efficient various additions are done in the model for example incorporation of visual attention mechanisms [16], [17], region of interests, and attention behaviors [18], [19]. A significant group of researchers believes that attention and visual attention would help in better understanding objects and their behaviors in the process of image captioning. Since visual attention is due to the higher-order convolutional work which reduces the spatial and localization information and reduces the semantic impact on the output. Similarly, the region of interest application over images during the encoding phase is also prevalent in the field of image captioning. The idea is to use multiple R-CNN-based object detectors and extract features from those regions for captioning. This helps in generating more verbose captions for each region separately but at the same time losses all the semantics and spatial relations among objects lying in the inter-region spaces of an image. However, despite the incorporation of effective techniques, there has been a serious gap persisting between the image and its generated caption for general-purpose use in daily life. This enables the requirements of handling semantic concepts of images and objects inside the image and making use of them while performing the captioning. Semantic understanding requires more than just mapping the representations during training and being able to produce an output during a test. The latest research focuses on semantically-oriented image captioning which concerns the object behavior, posture, and attribute during training and use them while doing the evaluation [20]–[22].

Among the list of the current state-of-the-art works in the field of image captioning which also resembles our idea up to a certain extent only is the use of graph convolution neural networks to understand the global and regional context of an image and its objects [23]–[25]. The graph convolution neural networks are used to understand the semantic and spatial relationship which helps the captioning model to

generate spatial tokens e.g., towards, inside, near, etc. Another approach used in the work [26] utilizes scene graph understanding along with objects to find the possible correlation of background/scene along with objects.

Current trends in the research community towards image captioning are more focused on using the visual mapping and correlations among the objects of the image and using these visual representations in the generation of captions. The usage of visual relation enables the captioning network to work on the semantics that helps in predicting the object and its behavior based on the subject [27], [39]. The latest shift in the learning of directions and geometrical understanding of objects in the image for captioning is critical towards more human-like caption generation and is being worked out on a global scale in the research community [28], [29], [39].

Despite extensive research and trends of using various techniques to understand the underlying semantics of objects in the image and translate it to a human-like text, we find a large gap in the research, pertaining to handling the geometrical and orientational details of the objects and mapping them in the generated caption.

Our research is aimed at providing a more thorough and in-depth approach towards finding the connections between spatial and geometrical semantics of image objects and converting them into respectable features for training the decoder stage.

III. APPROACH

This section discusses our approach used in ICPS for handling spatial and geometrical features of objects for image captioning. Our proposed architecture is composed of an encoder-decoder framework. The encoder is composed of parallelized capsule networks while the decoder is based on simple transformer neural networks. The use of capsule networks is based on the objective of being able to understand the geometrical and spatial details of objects while transformer neural network is due to it being the current state of the art in NLP [30].

A. ENCODER NETWORK

The baseline of our approach is the inclusion of a parallelized capsules network as an encoder stage for image captioning. The parallelized capsules network architecture allows for feature-specific learning in terms of spatial and geometrical contents on image objects. Our proposed architecture is given in Fig. 7.

A fundamental architecture of capsule network is composed of $2 \times$ convolution layers on the input with a generic 256 channels each with a 9×9 filter box and over a stride of 1. The activation layer is ReLu. The proceeding layer is also a convolution capsule with 6×6 of primary capsule grid along with 32 channels. This layer receives scalar input from the previous convolution layer and produces an 8-D vector over output. The squashing function handles non-linearity and outputs a 16-D vector for 10 classes of MNIST. The next layer performs probability calculation from the input fed by

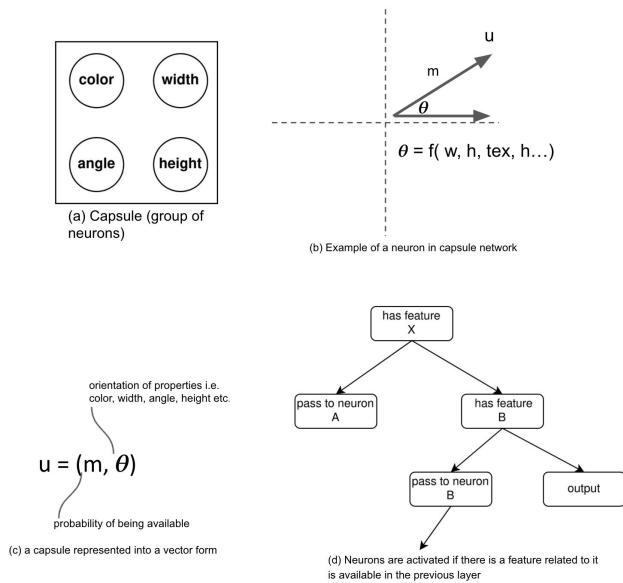


FIGURE 2. A fundamental building block of a capsule within the capsule network. (a). represents the group of neuron termed as capsule. (b). example of neuron as vector showing angle and magnitude. (c). vector representation. (d). activation over presence of feature.

the squashing unit over a 16-D vector. Later the reconstruction of the image is done using the final FC (fully connected) layers. This is the fundamental architect given in [32], and it is termed capsule networks with dynamic routing. The architecture executes in the way that the features are extracted using convolution layers and then fed to the primary capsule layer. Capsules in primary capsule layer has an associated activity vector to encoder spatial and geometrical information. The output from the primary capsule layer is then fed to the digitcaps layer which performs dynamic routing over the activation value along with coupling coefficients and then yields class probabilities which are then passed through fully connected layers to reconstruct the output image. Generally, there are 3 fully connected layers.

Every capsule in the capsule network architecture is responsible for focusing on a small area of interest and provides object details as a vector to the next capsule in the next layer [31]–[33]. Capsules are just a group of neurons having activation vectors as their instantiating parameters while the length of vectors defines the probability of the existence of a feature. Fig. 2. provides a simple description of the fundamental building blocks of the capsule network.

The relation of likelihood-based routing works well in the case of a simple dataset of images i.e. MNIST, Fashion MNIST where there is only 1 channel and a single object inside the image. However, the usefulness of capsule networks decreases when used for complex datasets i.e. Flickr, MSCOCO, CIFAR, and SVHN, etc. The presence of multiple channels and complex images having multiple objects not only increases the training time exponentially but also yields below state-of-the-art results [33]. To cater to multiple

channels & objects simultaneously our proposed encoder architecture handles the complexity issue with the help of parallelization on the input. In our architecture, we have used $16 \times$ capsules as a parallelized encoder input stage, each responsible for at least 1 object at a time, as a primary capsule cluster. The input convolution layer takes 3-dimension input having channels, kernel height & kernel width. After the extraction of features from convolution layer & passed through the activation layer, the input data is then converted into the batches of 16 and termed as D_{pc} , one for each capsule in the primary capsule layer, which results in a 4-dimensional block having channels, kernel width, kernel height and block size. Each capsule, from the set of 16 parallel capsules in the primary capsule layer, is fed with the input and expected to calculate object magnitude along with their geometrical & positional information by the help of angle. The calculation yields better in the way that there are 16 possible views for position calculation in our proposed cluster capsule network architecture as compared to simple capsule network architecture. The resulting matrix from all capsules in the primary capsule cluster is then aggregated and flattened to be fed to the feature capsule cluster which then yields the feature probability matrix based on the dynamic routing and squashing function. The feature capsule generates a feature matrix of size N . The generation of features instead of classes is way faster and provides better traction in training the model since features provide more drill-down information and eliminate the need to have a flattened and sigmoid layer for calculating the class probabilities. The primary capsule cluster layer is fed by 512 feature maps convolution neural networks. This combination provides a $512 + 36$ feature engine leading towards a better understanding of the input image features in terms of the relative position and geometry. Furthermore, the architecture uses the skip connection in the convolution layer to ensure that discriminating features are fed to the primary capsule cluster layer to generate a better-performing matrix. Fig. 3 provides a detailed view of our proposed architecture for encoder modules using capsule network clusters.

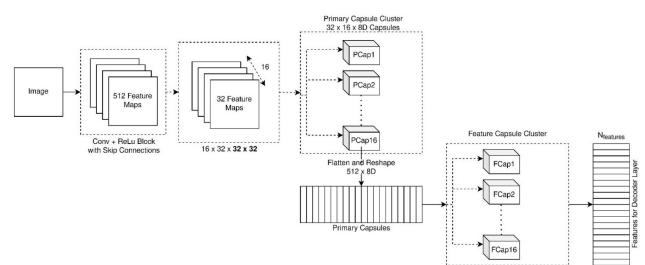


FIGURE 3. Proposed architecture of encoder module for ICPS.

Figure. 4 provides a description of a single capsule with details of its dimension in our proposed primary capsule cluster for ICPS. The actual input dimension is 3 having kernel width, kernel height and channels however, the batch size serves as the dimension as well to generate the cluster as per the input feed.

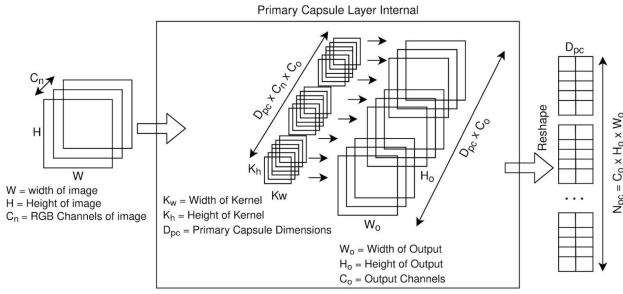


FIGURE 4. Detailed architecture of a single capsule used in our ICPS.

The input parameters are calculated using the equation $D_{pc} * C_n * k_w * k_h$, where:

- D_{pc} is the dimension of the capsule network
- k_w and k_h are the width and height of kernels respectively
- C_n is the number of input channels.

B. DECODER

The decoding stage of our architecture is based on the transformer neural networks. The selection of Transformer is done due to large dataset handling along with parallelized behavior for caption generation and self-attention mechanism [34], [35]. Features are injected into the Transformer Neural Network during the training. Figure 5. shows a simple architecture of our transformer neural network.

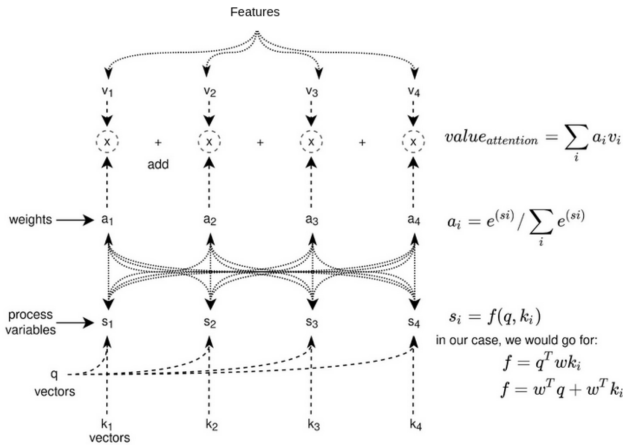


FIGURE 5. Baseline architecture of our proposed dynamic head transformer neural network for image captioning decoder.

The image features having object detail, shape, geometry, and location is processed through the input embedding layer to reduce the dimension from D_n to D_m . This reduction allows a smaller computation time during the training stage over the cost of smaller loss in the input information. One major difference in our approach is that the decoder stage is *dynamic* as per the output features of the capsule networks-based encoder stage. This will enable the Transformer Neural

Network to learn with the dynamic behavior of input and will also provide a sufficient boost in the learning behaviors.

The operation in the encoder stage of the Transformer for a single layer works in the way that each feature F_i is fed to the input embedding layer. The input embedding layer converts the feature vectors into embedding vectors.

This feeding behavior is not recurrent as of RNN/LSTM, instead, the application of the Multi-headed attention layer expects concurrent injection of feature vectors. This concurrent behavior provides a massive boost in the performance and also helps in ensuring long-term dependencies in the text. However, the simultaneous flow of features forces the transformer neural network towards losing the position or order of feature words/vectors within the text. The order or position of words/features are critical for grammar, linguistic, and making sense of the sentence. The workaround is to attach a position-dependent signal with each feature and is termed as positional encoding in the transformer neural networks.

The embedding vectors are then added with positional encoding to generate a positional embedded vector. This position embedding vector has information of features and its position in the series of features or text. The positional encoding in Transformer is done with the use of equations 1 and 2 for odd and even positions of features in the series of vectors, respectively.

$$PE(pos, 2i + 1) = \cos(pos/10000^{2i/dmodel}) \quad (1)$$

$$PE(pos, 2i) = \sin(pos/10000^{2i/dmodel}) \quad (2)$$

In equations 1 and 2, pos refers to the position of the feature in the sequence of features, the positioning starts from the feature at index 0. $dmodel$ defines the depth of feature embedding, i.e. the total number of words/tokens in the sequence and i refers to the index number for each dimension of the embedding. The term PE is short for positional encoding while the $pos, 2i + 1$ and $pos, 2i$ are odd and even representations.

The application of cosine and sin functions helps linear models to learn easily about positional encoding. sin and cosine functions can be taken as the continuous counter effects of binary positions in the domain of floats. It is just like alternating bits over the wave. However, the alternating position will appear again and again for various features/words so the inclusion of i will vary the frequency of the wave and change the positional argument for every feature/word. The same statement applies to the cosine function as well. Fig. 6, provides a descriptive behavior of the discussion. Each dimension corresponds to a *sin* wave and creates a geometric progression from 2π to 10000.2π . The value 10000 is a scaling factor in the positional encoding which helps in creating a fairly large circle to compensate approx. 1k or fewer features/words.

The positional embedding input is passed through the Multi-headed Attention layer. The number of heads in

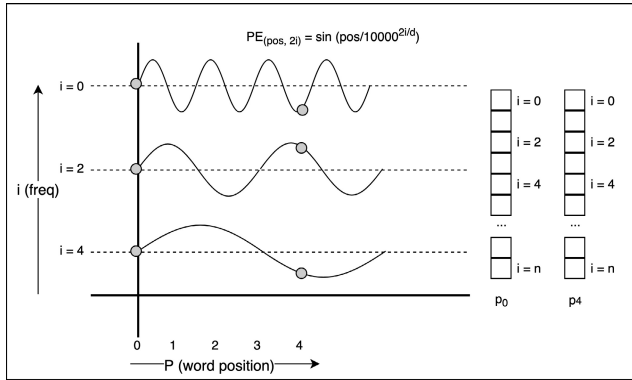


FIGURE 6. An over view of the working of positional encoding in transformer neural networks.

multi-headed attention depends upon the number of parameters we would like to learn.

The entire architecture of our proposed research can be viewed in Figure 7. The results acquired over Flickr8k datasets are significantly better than the current state-of-the-art papers in terms of positional and geometrical semantics. Along with the positional information our ICPS has better captioning by giving adverb information in the caption. A few examples of our ICPS output are given in Figure 8.

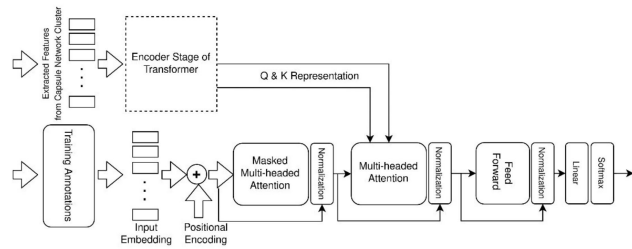


FIGURE 7. Integrated view of ICPS with capsule network cluster as encoding stage and transformers neural network as decoding stage.



FIGURE 8. Test results from Flickr8k Images using ICPS after going through a training of limited epochs.

Figures 9 and 10. provides a comparison of BLEU-4 and METEOR scores against some state-of-the-art research



FIGURE 9. Comparison of BLEU-4 score with some state of the art research work over for 100 epochs only.

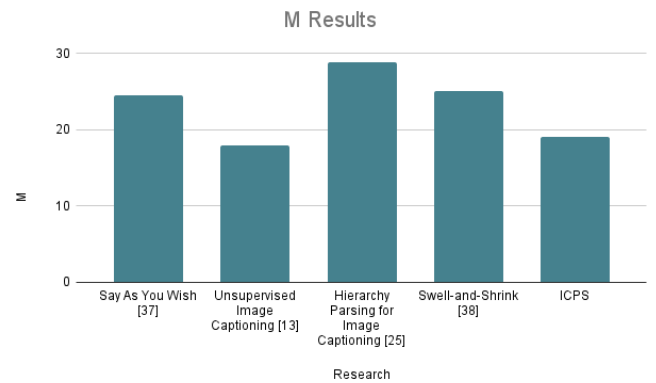


FIGURE 10. Comparison of METEOR score with some state of the art research work over for 100 epochs only.

works in the field of captioning. Our results are acquired over a training round of 100 epochs only while the state-of-the-art works run their model up to 6000 epochs. This provides the traction in our model that despite running for a few epochs it provides a more comparative and, in some cases, significantly better results. Despite the BLEU-4 scores being higher than [37], [13] in Fig. 9 the response over text generation is far superior and provides the understanding of position and geometry within it over a very limited training resource consumption. Similarly, the METEOR score in Fig. 10 is slightly higher than of [13], while the image feature geometrical and positional understanding is superior as compared to the state-of-the-art works. BLEU and METEOR scores are benchmarks in image captioning research, however, they do not have the tendency to account for positional and geometrical information within the text. BLEU performs text to text comparison while METEOR tries to find the recall and precision of the uni-grams within the text. Both are textual evaluation metrics. The scores are used to provide the relation of comparison between ICPS and the mentioned state-of-the-artworks.

IV. EXPERIMENTATION

The capability of our ICPS model having capsule clusters for geometrical and positional inferences is demonstrated over

Flicker8k and Flicker30k datasets. The selection of flicker datasets is done on the basis of simplicity, size, and density of the image matrix. The architecture of capsule networks requires a significant amount of processing resources when used for complex datasets i.e. MSCOCO [11]. In our case, we are using a cluster of capsule networks which becomes more resource-intensive during training and computation. To have tractable results with limited resources we evaluated our methodology over flicker8k and flicker30k datasets only. The resource constraints have limited us to train our model up to 100 epochs over NVIDIA Quadro P4000 GPU with 64GB of RAM and Xeon processor. The batch size was selected as 16 while the learning rates were 0.001 and Adam was used as the optimization function. The images were scaled down during the initial run to understand the PoC while in the final training the scaling factor was removed. The training, validation, and test sets were taken 80%, 10%, and 10% respectively for each dataset. So for flicker8k 4800 images were taken for training while 600 were for validation and 600 images for testing. A quantitative comparison of generated captions over the same images is given in Table 1. It is worth noting that all the test images used in [13], [25], [38] are from the MSCOCO dataset and our ICPS being only trained over the flicker dataset provides a more comprehensive position and orientation caption against them.

TABLE 1. Quantitative comparison of generated captions over same test images by the state-of-artworks and ours.



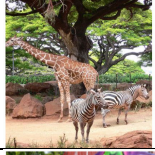

Image	Caption
	<ul style="list-style-type: none"> • [13] a black and white cat on a wooden background • [Ours] a cat eating in a wooden bowl
	<ul style="list-style-type: none"> • [13] top view of a bowl of healthy food • [Ours] two bowls next to each other on a table
	<ul style="list-style-type: none"> • [25] two zebras and a giraffe standing next to a tree • [Ours] two zebras with a giraffe standing in front of a tree
	<ul style="list-style-type: none"> • [13] a fire hydrant sitting on the side of a road • [Ours] a fire hydrant next to a colorful wall

Table 2. provides a comparison of our ICPS results for BLEU-4 & METEOR with the mentioned state-of-the-arts research. The comparison is done in a standard way to reflect the outcome of our ICPS with a minimum amount of training and resource consumption. The novelty of our ICPS lies in being able to be trained with a limited amount of data in a

TABLE 2. Comparison of metrics along with encoder architecture & pre-trained models.

Research	Training		Encoder	
	BLEU-4	METEOR	Architecture	Pre-Trained
[37]	23.0	24.5	FRCNN	ResNet-152
[13]	18.6	17.9	CNN	Inception-V4
[25]	39.3	28.8	FRCNN/MRCNN	ResNet-101
[38]	31.6	25.1	FRCNN	VGG/GoogleNet
Ours	24.1	19	CapsNet	None

very short amount of training time for a comparatively better result.

We used the publicly available code [36] for baseline CapsNet and modified the setup to run as a cluster while in the case of the decoding module the default transformer neural network code was updated to meet the requirements of dynamic attention. The following 2× public datasets were used for our experiments:

A. Flicker8k

Flicker8k dataset is small and can easily be used for training a complex model over small processing hardware. The dataset is labeled with 5 captions per image. Flicker8k consists of 8092 JPEG images with varying shapes and sizes. Out of 8092 images, 6k images are used for training, 1k for validation, and 1k for testing. The dataset comes with a caption file in text having a total of 40,460 captions for all 8092 images.

B. Flicker30k

As the name suggests, Flicker30k is of 31k images collected from the Flicker group and each image contains 5 associated captions in text. The captioning is done via manual human annotation. The total captions are 158k with 244k referencing chains. The flicker30k also comes with bounding boxes which are 276k in the count.

Our initial round of experiments is in comparison to the [36] and follows the same ethics to understand the Capsule Network baseline better, however, we never moved towards final classification and reconstruction instead we limited ourselves to feature extraction only. Then we updated the code to perform parallelization behavior in feature extraction and compared the results with non-parallel feature extraction with original code over the Flicker8k dataset. The graph in Figure. 11 and 12 provide a comparison of feature extraction accuracy and speed between simple capsule network [36] and our parallelized capsule networks. The higher the value of accuracy the better the result is while the lower the value of time consumption shows a better performing network.

Table. 3 & 4 contains a few values in bold which are the best ICPS results in terms of accuracy and efficiency for the Flicker8k and 32k datasets and demonstrated 4-5% improvement over the existing result. The results are promising and have a 98% recall rate over 10 tries of execution with the same conditions.

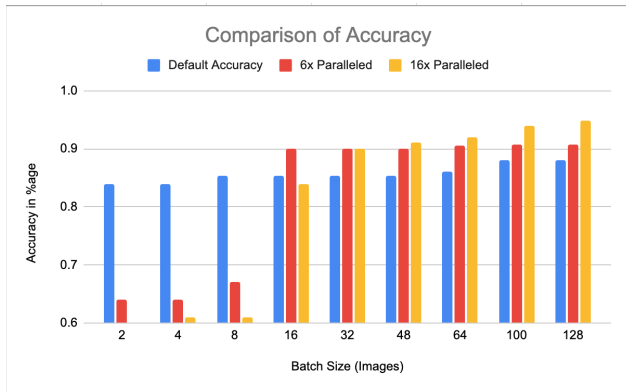


FIGURE 11. Comparison of feature extraction accuracy between original CapsNet model and parallelized capsule network.

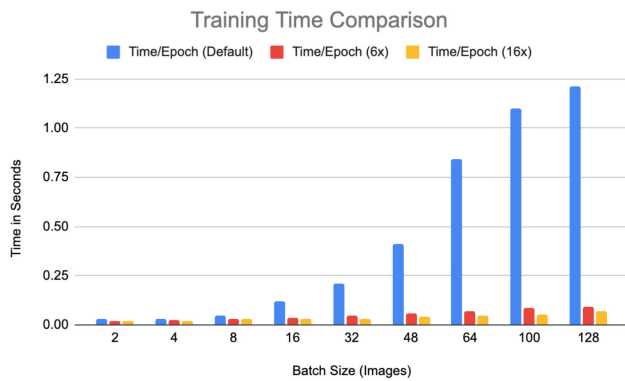


FIGURE 12. Comparison of feature extraction training time consumption per epoch among default, 6x paralleled and 16x paralleled CapsNet for Image captioning.

TABLE 3. Comparison of default capsule network with our cluster capsule in terms of accuracy & training time consumption per epoch. The training time is measured in seconds. Table data is acquired over Flickr8k dataset.

Batch	Default		6x Parallel		16x Parallel	
	T/Ep	Acc.	T/Ep	Acc	T/Ep	Acc
2	0.031	0.84	0.022	0.64	0.021	0.6
4	0.032	0.84	0.024	0.64	0.022	0.61
8	0.048	0.853	0.031	0.67	0.028	0.61
16	0.121	0.853	0.037	0.9	0.031	0.84
32	0.21	0.854	0.048	0.9	0.033	0.9
48	0.412	0.854	0.059	0.9	0.04	0.91
64	0.84	0.86	0.067	0.905	0.048	0.92
100	1.1	0.881	0.089	0.907	0.054	0.94
128	1.21	0.881	0.091	0.908	0.067	0.948

Tables 3 & 4 provide comparative analysis in terms of the accuracy and efficiency of our capsule network cluster designed for the image captioning job. The results reflect that our ICPS has significantly better efficiency in performing the feature extraction from images as compared to the original capsule networks. The delta in terms of seconds is huge and provides an above 100% boost in the processing of a single epoch for a batch of 128 images.

TABLE 4. Comparison of default capsule network with our cluster capsule in terms of accuracy & training time consumption per epoch. The training time is measured in seconds. Table data is acquired over Flickr30k dataset.

Batch	Default		6x Parallel		12x Parallel	
	T/Ep	Acc	T/Ep	Acc	T/Ep	Acc
2	0.031	0.88	0.022	0.78	0.021	0.74
4	0.032	0.88	0.024	0.78	0.022	0.74
8	0.048	0.891	0.031	0.84	0.028	0.83
16	0.121	0.895	0.037	0.851	0.031	0.9
32	0.21	0.901	0.048	0.89	0.033	0.93
48	0.412	0.91	0.059	0.92	0.04	0.94
64	0.84	0.92	0.067	0.94	0.048	0.948
100	1.1	0.92	0.089	0.96	0.054	0.96
128	1.21	0.931	0.091	0.965	0.067	0.97



(13-A) A group of trucks parked on the hill with a mountain in the background



(13-B) A person lying on a rock covered mountain with a mountain range in the background



(13-C) A big bus and couple of cars standing at a station



(13-D) A herd of sheep walking down a road next to a forest.

FIGURE 13. Output of ICPS model captioning on random google images.

C. REAL WORLD EXPERIMENT

In addition to test and validation over Flickr8k and Flickr30 datasets, we tested our model over random images taken from google images for the purpose of real-world validation tests and the results are given in Figure 13. Although the results provide better captioning in terms of positional and geometrical information, however, in some cases the model can be wrong in captioning and understanding the image. For example, the caption done in 13-B is wrong as compared to the original image. The image is of a mountain range with curves, which our model understands as a person.

V. CONCLUSION

Our research has introduced an innovative approach for image captioning with the use of capsule networks. We made the model tractable and tangible in time and resource consumption which is currently a big bottleneck for capsule network-based computer vision solutions. The idea we put forth is that of parallelization of capsules and extraction of features instead of classes which reduced the time of training by removing the final stage in the capsule network. The inclusion of transformer neural networks with dynamic heads also performed better and provided human-level captioning. The feature extraction from capsule clusters helped in finding the positional and geometrical semantics from the image, which relates better in captioning and making it more verbose and human-like. The achieved values of BLEU and METEOR

scores indicate that the captioning has improved in comparison to the current state-of-the-art works. The 16-capsules cluster achieved 94% accuracy over the Flickr8k dataset in extracting features in a small training epoch time.

We expect to update our model for handling more complex datasets e.g., MSCOCO and ImageNet in our future works and would attempt to provide a more comprehensive open-source solution for generic image captioning with positional semantics for the research community.

REFERENCES

- [1] H. Al Fatta and U. Fajar, "Captioning image using convolutional neural network (CNN) and long-short term memory (LSTM)," in *Proc. Int. Seminar Res. Inf. Technol. Intell. Syst. (ISRITI)*, Dec. 2019, pp. 263–268, doi: [10.1109/ISRITI48646.2019.9034562](https://doi.org/10.1109/ISRITI48646.2019.9034562).
- [2] O. Sargar and S. Kinger, "Image captioning methods and metrics," in *Proc. Int. Conf. Emerg. Smart Comput. Informat. (ESCI)*, Mar. 2021, pp. 522–526, doi: [10.1109/ESCI50559.2021.9396839](https://doi.org/10.1109/ESCI50559.2021.9396839).
- [3] Y. Chu, X. Yue, L. Yu, M. Sergei, and Z. Wang, "Automatic image captioning based on ResNet50 and LSTM with soft attention," *Wireless Commun. Mobile Comput.*, vol. 2020, Oct. 2020, Art. no. 8909458, doi: [10.1155/2020/8909458](https://doi.org/10.1155/2020/8909458).
- [4] Z. Yuan, X. Li, and Q. Wang, "Exploring multi-level attention and semantic relationship for remote sensing image captioning," *IEEE Access*, vol. 8, pp. 2608–2620, 2020, doi: [10.1109/ACCESS.2019.2962195](https://doi.org/10.1109/ACCESS.2019.2962195).
- [5] S. Wang, L. Lan, X. Zhang, G. Dong, and Z. Luo, "Cascade semantic fusion for image captioning," *IEEE Access*, vol. 7, pp. 66680–66688, 2019, doi: [10.1109/ACCESS.2019.2917979](https://doi.org/10.1109/ACCESS.2019.2917979).
- [6] X. Xiao, L. Wang, K. Ding, S. Xiang, and C. Pan, "Dense semantic embedding network for image captioning," *Pattern Recognit.*, vol. 90, pp. 285–296, Jun. 2019, doi: [10.1016/j.patcog.2019.01.028](https://doi.org/10.1016/j.patcog.2019.01.028).
- [7] H. F. Pardede, E. Suryawati, V. Zilvan, A. Ramdan, R. B. S. Kusumo, A. Heryana, R. S. Yuwana, D. Krisnandi, A. Subekti, F. Fauziah, and V. P. Rahadi, "Plant diseases detection with low resolution data using nested skip connections," *J. Big Data*, vol. 7, no. 1, p. 57, Dec. 2020, doi: [10.1186/s40537-020-00332-7](https://doi.org/10.1186/s40537-020-00332-7).
- [8] Z. Lin, J. Jia, W. Gao, and F. Huang, "Fine-grained visual categorization of butterfly specimens at sub-species level via a convolutional neural network with skip-connections," *Neurocomputing*, vol. 384, pp. 295–313, Apr. 2020, doi: [10.1016/j.neucom.2019.11.033](https://doi.org/10.1016/j.neucom.2019.11.033).
- [9] M. Umer, S. Sadiq, M. Ahmad, S. Ullah, G. S. Choi, and A. Mehmood, "A novel stacked CNN for malarial parasite detection in thin blood smear images," *IEEE Access*, vol. 8, pp. 93782–93792, 2020, doi: [10.1109/ACCESS.2020.2994810](https://doi.org/10.1109/ACCESS.2020.2994810).
- [10] M. Hodosh, P. Young, and J. Hockenmaier, "Framing image description as a ranking task: Data, models and evaluation metrics," *J. Artif. Intell. Res.*, vol. 47, no. 1, pp. 853–899, 2013.
- [11] T. Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft COCO (common objects in context): Common objects in context," in *Proc. CVPR*, 2014, pp. 740–755.
- [12] I. Laina, R. Christian, and N. Nassir, "Towards unsupervised image captioning with shared multimodal embeddings," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 7414–7424.
- [13] Y. Feng, L. Ma, W. Liu, and J. Luo, "Unsupervised image captioning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4120–4129, doi: [10.1109/CVPR.2019.00425](https://doi.org/10.1109/CVPR.2019.00425).
- [14] W. Zhao, X. Wu, and X. Zhang, "Memcap: Memorizing style knowledge for image captioning," in *Proc. AAAI*, 2020, pp. 12984–12992.
- [15] Y. Qin, J. Du, Y. Zhang, and H. Lu, "Look back and predict forward in image captioning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8359–8367, doi: [10.1109/CVPR.2019.00856](https://doi.org/10.1109/CVPR.2019.00856).
- [16] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 664–676, Apr. 2017, doi: [10.1109/TPAMI.2016.2598339](https://doi.org/10.1109/TPAMI.2016.2598339).
- [17] Z. Yang and Q. Liu, "ATT-BM-SOM: A framework of effectively choosing image information and optimizing syntax for image captioning," *IEEE Access*, vol. 8, pp. 50565–50573, 2020, doi: [10.1109/ACCESS.2020.2980578](https://doi.org/10.1109/ACCESS.2020.2980578).
- [18] J. Wang, W. Wang, L. Wang, Z. Wang, D. D. Feng, and T. Tan, "Learning visual relationship and context-aware attention for image captioning," *Pattern Recognit.*, vol. 98, Feb. 2020, Art. no. 107075, doi: [10.1016/j.patcog.2019.107075](https://doi.org/10.1016/j.patcog.2019.107075).
- [19] S. Chen and Q. Zhao, "Boosted attention: Leveraging human attention for image captioning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 68–84.
- [20] E. Cetinic, "Iconographic image captioning for artworks," in *Pattern Recognition. ICPR International Workshops and Challenges* (Lecture Notes in Computer Science), vol. 12663, A. Del Bimbo et al., Eds. Cham, Switzerland: Springer, 2020, pp. 502–516, doi: [10.1007/978-3-030-68796-0_36](https://doi.org/10.1007/978-3-030-68796-0_36).
- [21] Z. Ren, X. Wang, N. Zhang, X. Lv, and L.-J. Li, "Deep reinforcement learning-based image captioning with embedding reward," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 290–298.
- [22] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 652–663, Apr. 2017, doi: [10.1109/TPAMI.2016.2587640](https://doi.org/10.1109/TPAMI.2016.2587640).
- [23] L. Wu, M. Xu, L. Sang, T. Yao, and T. Mei, "Noise augmented double-stream graph convolutional networks for image captioning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 8, pp. 3118–3127, Aug. 2021, doi: [10.1109/TCSVT.2020.3036860](https://doi.org/10.1109/TCSVT.2020.3036860).
- [24] T. Yao, Y. Pan, Y. Li, and T. Mei, "Exploring visual relationship for image captioning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 684–699.
- [25] T. Yao, Y. Pan, Y. Li, and T. Mei, "Hierarchy parsing for image captioning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2621–2629.
- [26] L. Gao, B. Wang, and W. Wang, "Image captioning with scene-graph based semantic concepts," in *Proc. 10th Int. Conf. Mach. Learn. Comput.*, Feb. 2018, pp. 225–229, doi: [10.1145/3195106.3195114](https://doi.org/10.1145/3195106.3195114).
- [27] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4651–4659.
- [28] L. Guo, J. Liu, X. Zhu, P. Yao, S. Lu, and H. Lu, "Normalized and geometry-aware self-attention network for image captioning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10327–10336.
- [29] L. Guo, J. Liu, J. Tang, J. Li, W. Luo, and H. Lu, "Aligning linguistic words and visual semantic units for image captioning," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 765–773, doi: [10.1145/3343031.3350943](https://doi.org/10.1145/3343031.3350943).
- [30] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang, S. Watanabe, T. Yoshimura, and W. Zhang, "A comparative study on transformer vs RNN in speech applications," in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Dec. 2019, pp. 449–456, doi: [10.1109/ASRU46091.2019.9003750](https://doi.org/10.1109/ASRU46091.2019.9003750).
- [31] P. L. Shah, T. K. Gupta, J. B. Dhakad, and M. R. D'silva, "A review paper on understanding capsule networks," in *Proc. IJEDR*, 2018, pp. 58–65.
- [32] S. Sabour, N. Fross, and G. E. Hinton, "Dynamic routing between capsules," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 3859–3869.
- [33] M. K. Patrick, A. F. Adekoya, A. A. Mighty, and B. Y. Edward, "Capsule networks—A survey," *J. King Saud Univ.-Comput. Inf. Sci.*, to be published.
- [34] J. Yu, J. Li, Z. Yu, and Q. Huang, "Multimodal transformer with multi-view visual representation for image captioning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 12, pp. 4467–4480, Dec. 2020, doi: [10.1109/TCSVT.2019.2947482](https://doi.org/10.1109/TCSVT.2019.2947482).
- [35] D. Liu and G. Liu, "A transformer-based variational autoencoder for sentence generation," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–7, doi: [10.1109/IJCNN.2019.8852155](https://doi.org/10.1109/IJCNN.2019.8852155).
- [36] Higgsfield. Accessed: Sep. 5, 2021. [Online]. Available: <https://github.com/higgsfield/Capsule-Network-Tutorial>
- [37] S. Chen, Q. Jin, P. Wang, and Q. Wu, "Say as you wish: Fine-grained control of image caption generation with abstract scene graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9962–9971.
- [38] J. Wang, H. Wang, and K. Xu, "Swell-and-shrink: Decomposing image captioning by transformation and summarization," *Proc. IJCAI*, 2019, pp. 5226–5232, doi: [10.24963/IJCAI.2019/726](https://doi.org/10.24963/IJCAI.2019/726).
- [39] L. Huang, W. Wang, J. Chen, and X.-Y. Wei, "Attention on attention for image captioning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4634–4643.

...