IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Text to Image Synthesis for Improved Image Captioning

**MD. ZAKIR HOSSAIN[ID]1, (Student Member, IEEE),
FERDOUS SOHEL[ID]1, (Senior Member, IEEE), MOHD FAIRUZ SHIRATUDDIN1,
HAMID LAGA[ID]1, AND MOHAMMED BENNAMOUN[ID]2, (Senior Member, IEEE)**

1Discipline of Information Technology, Murdoch University, Perth, WA 6150, Australia
2Department of Computer Science and Software Engineering, The University of Western Australia, Perth, WA 6009, Australia

Corresponding author: Md. Zakir Hossain (mdzakir.hossain@murdoch.edu.au)

ABSTRACT Generating textual descriptions of images has been an important topic in computer vision and natural language processing. A number of techniques based on deep learning have been proposed on this topic. These techniques use human-annotated images for training and testing the models. These models require a large number of training data to perform at their full potential. Collecting human generated images with associative captions is expensive and time-consuming. In this paper, we propose an image captioning method that uses both real and synthetic data for training and testing the model. We use a Generative Adversarial Network (GAN) based text to image generator to generate synthetic images. We use an attention-based image captioning method trained on both real and synthetic images to generate the captions. We demonstrate the results of our models using both qualitative and quantitative analysis on popularly used evaluation metrics. We show that our experimental results achieve two fold benefits of our proposed work: i) it demonstrates the effectiveness of image captioning for synthetic images, and ii) it further improves the quality of the generated captions for real images, understandably because we use additional images for training.

INDEX TERMS Image captioning, synthetic images, attention, generative adversarial network.

I. INTRODUCTION

Image captioning is the task of providing a natural language description of the content in an image. It lies at the intersection of computer vision and Natural Language Processing (NLP) [1]. Automatic image captioning is useful to many applications, such as developing image search engines with complex natural language queries and helping the visually impaired people to understand their surroundings. Hence, image captioning has been an active research area. The advent of new convolutional neural networks and object detection architectures have contributed enormously to improving image captioning. Moreover, sophisticated sequential models, such as attention-based recurrent neural networks, have also been presented for accurate image caption generation.

Inspired by neural machine translation, most modern deep learning-based image captioning methods use an encoder-decoder framework. In this framework, an encoder is used to encode an intermediate representation of the informa-

tion contained within the image. A decoder is used to decode this information into a descriptive text sequence. Thus this framework is composed of two principal modules: a Convolutional Neural Network (CNN) [2], [3] as an encoder for image feature extraction and a Long Short-Term Memory (LSTM) model [4] as a language decoder for caption generation.

Different CNNs such as AlexNet [5], VGGNet [6], ResNet [7], and DenseNet [8] have their own strengths and weaknesses. It is generally accepted that the deeper the network is, the more relevant are the learned features [7]. However, if the depth of the network exceeds a threshold, one may obtain the opposite effect, i.e., a decline in performance. There are two main reasons behind this fact: (i) The vanishing-gradient problem: when the input or the gradient passes through many layers, it can vanish or gets "washed out" by the time it reaches the end of the network, and (ii) the degradation problem. This problem has been addressed in the literature by using residual learning mechanisms such as ResNet [8]. However, the element-wise addition used in the identity mapping in ResNet is computationally expensive during training. In contrast, with DenseNet, each layer has connections with

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

every other layer in the network in a feed-forward manner. The network reuses the feature-maps and uses concatenation for various operations instead of addition. Therefore, it can reduce the number of parameters and it can be memory efficient. Moreover, each layer of DenseNet receives feature maps from all previous layers. Thus, it gets diversified features and tends to have rich patterns. In this paper, we use DenseNet as an encoder to extract image features.

However, encoder-decoder based methods focus only on the factual description of an image. They lose the information of the relevant objects in the scene. Visual attention mechanisms can selectively focus on the relevant parts of the image for a period of time, similar to the human visual system. Simultaneously, they can discard irrelevant information. Several methods [9], [10] use attention-based techniques and can describe the relevant parts of the image successfully. All of these methods use the three most common datasets: Microsoft COCO (MSCOCO) [11], Flickr30k [12], and Flickr8k [13]. The images of all these datasets are human-annotated. However, these deep learning-based methods require a large amount of labeled data in order for them to perform at their very best. Moreover, the manual generation of (additional) data is expensive and time-consuming [14].

Nowadays a lot contents including images are generated automatically, e.g., for news, illustration, artwork, promotion, as well as for human computer interaction and augmented reality. Such synthetic data can be effectively used in machine learning techniques, where there is a scarcity of labelled data.

Application such as sceneflow [15], classification [16], semantic segmentation [17], and 3D reconstruction [17] have all benefited from the use of synthetic data.

To the best of our knowledge, there is no method available in image captioning which use synthetic images. Existing image caption generators are only trained on labelled real images. It is important to develop caption generators for synthetic images as well. In this work, we extend the training of caption generators by using both real and synthetic images. Getting new synthetic images with appropriate caption-labels is a challenge. To generate new synthetic but labelled images we resort to the ground truth captions available with current datasets. For example, each image in MSCOCO dataset usually has five captions. We use these captions to generate five synthetic images. We subsequently label these synthetic images with the respective captions.

Generative Adversarial Networks (GANs) have been popularly used in generating synthetic images. In this work, we use an attention-based GAN, which can generate synthetic images from an input text. This gives us a synthetic image dataset with ground truth captions. We further use these synthetic images together with the real images to train and test an image captioning module. Finally, we demonstrated that synthetic images can significantly improve the quality of the generated captions.

Overall, we investigate and analyze image captioning for real images as well as machine-generated synthetic images. Thus, this paper has the following key contributions:

- We use a GAN-based text-to-image synthesis method to generate synthetic images from text.
- We use both real and synthetic images for training and testing our model.
- Finally, we demonstrate that synthetic data can significantly improve the performance of caption generators.

We organize the rest of the paper as follows: In Section II, we discuss the related work. The architecture and methodology of the proposed technique are described in Section III. Experimental results are discussed in Section IV. Section V concludes the paper.

II. RELATED WORK

With the advancements in deep neural network models, automatic image captioning has become a promising research area. Hossain *et al.* [18] present a comprehensive survey of the topic. They group the methods into several categories namely, template-based image captioning, retrieval-based image captioning, and novel caption generation. Template-based methods [19] use fixed templates with a number of blank slots to generate captions. In these methods, different objects, attributes, and actions are detected first, and then the blank spaces in the templates are filled. However, templates are predefined and cannot generate variable-length captions.

Captions can also be retrieved from visual space and multi-modal space [20]. In retrieval-based methods, captions are retrieved from a set of existing captions [21]. These methods produce generalized syntactically correct captions. However, they have limitations in producing image-specific syntactically correct captions [22].

Novel captions can be generated from both visual space and multimodal space [23], [24]. A typical method of this category analyzes the visual content of the image first and then generates the image captions using a language model. These methods can generate image captions that are semantically more accurate than the aforementioned approaches [22]. Most methods of this category use an encoder-decoder architecture to generate image captions [23]. In these methods, a vanilla CNN is used as the encoder to extract the image representations and an LSTM is used as a decoder to generate captions using these representations. However, these methods have problems in identifying prominent objects of the image.

Attention-based methods [24], [25] can represent the prominent objects in captions because they selectively focus on the relevant objects of an image. Therefore, we use an attention-based method to generate a description of an image.

These deep learning-based image captioning methods popularly use three common publicly available datasets i.e., MSCOCO [11], Flickr30k [12], and Flickr8k [13] for training and testing the networks. These datasets were collected and annotated by humans. However, deep learning-based methods have some issues to work with these data.
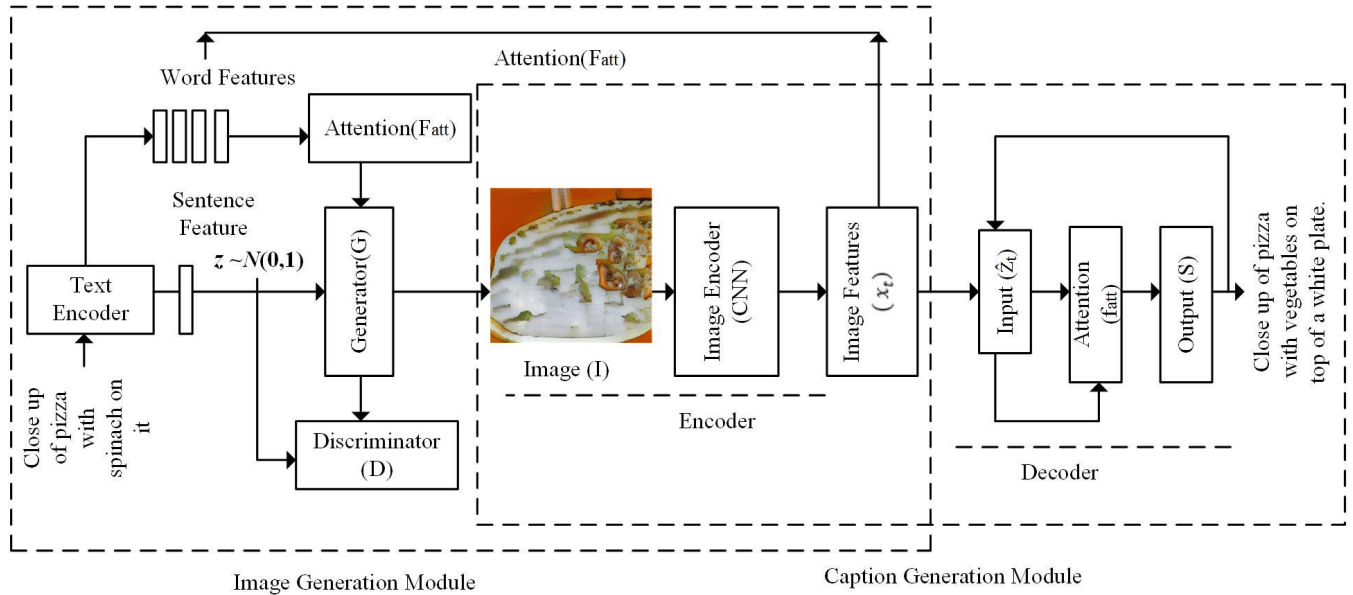
FIGURE 1. The architecture of our proposed method: a GAN-based model is used to generate synthetic images from text. The model applies attention to focus on the relevant word vectors to generate different regions of the image. Then an attention-based image captioning model is used to generate captions for that image. Image (I) can refer to any image (either real or synthetic), whichever is being used for training.

- These methods require a large and diverse set of data to learn the visual representations.
- Existing models overfit the common objects that co-occur in a common context. For example, if a model is trained for a scene which contains a bed and bedroom but it is tested on unseen contexts e.g., bed and forest. The model will struggle to generalize to these scenes.
- The Manual labelling of large volume of data is expensive, biased, and time-consuming.

Synthetic data can be an attractive alternative to address these issues. A number of methods [15], [26] have been proposed to generate synthetic images for different computer vision tasks such as semantic segmentation, object classification, and 3D reconstruction. In recent years, GAN-based methods have shown significant advances in image synthesis. They can generate more accurate, more semantically consistent results than traditional methods. GANs can produce textured details and realistic content of an image. They are useful for many applications, such as texture synthesis, super-resolution, and image inpainting.

Qiao *et al.* [27] proposed a text-to-image generation method in "MirrorGAN: Learning Text-to-Image Generation by Redescription". The ultimate goal of this method is to generate high-quality and visually realistic images. In contrast, the ultimate goal of our method is to generate semantically meaningful and superior image captions than a baseline method. Therefore, MirrorGAN can be regarded as the inverse problem of image captioning. The methods [28] and [29] also used GAN in image captioning. Dai *et al.* [28] used a Conditional Generative Adversarial Networks (CGAN) for evaluating the generated captions with the

aim to improve the naturalness and diversity of the captions. In this work, CGAN is used as an evaluation metric. Similarly, Chen *et al.* [29] used the same CGAN with reinforcement learning to deal with the inconsistent evaluation problem among different existing evaluation metrics.

All existing methods use one encoder to extract image representation. One encoder is not capable to extract diverse and semantic information of images [30]. Jiang *et al.* introduced an image captioning method where they used multiple encoder to extract image features. They proposed a Recurrent Fusion Network (RFNet) with multiple encoders for image captioning. They demonstrated that multiple CNNs, serve as the encoders, can provide diverse and comprehensive descriptions of the input image.

In the image captioning literature, it has been recognized that reasoning visual relationships, i.e., interactions or relative positions between object is crucial to a richer semantic understanding of an image [31], [32]. Yao *et al.* [33] proposed a Graph Convolutional Networks plus Long Short-Term Memory (GCN-LSTM) architecture based image captioning. This architecture is capable to extract both semantic and spatial object relationship of an image.

Attention mechanism is one of the valuable breakthroughs in image captioning. The attention mechanism used in the existing image captioning methods generates an weighted average on encoded vectors at each time step to guide the caption decoding process. Thus the decoder does not have much idea whether the generated attention is really related to the given query. Huang *et al.* [34] proposed an attention on attention (AoA) module for image captioning. This method adds an additional attention on top of the conventional attention

mechanisms to determine the relevance between the generated attention results and the given queries.

The traditional LSTMs tend to focus on the relatively closer vocabulary while ignoring the long-term dependencies [35]. Ke *et al.* proposed a Reflective Decoding Network (RDN) for image captioning [36]. This method jointly applies attention mechanism in both visual and textual domain to capture the long-range dependency between words of a caption. Thus the method can extract each word's relative position to come up with the maximum information in the generated caption.

Cornia *et al.* proposed a novel fully-attentive approach for image captioning [37]. They follow the architecture of Transformer model [35]. This method incorporates a multi-layer architecture to exploit both low-level and high-level visual relationships of an input image. Experimental results show that the method can generate high quality captions for images.

Pan *et al.* proposed an X-Linear attention block for image captioning [38]. This framework applies a bi-linear attention pooling to extract both spatial and channel-wise attention. These attentions can capture rich visual information of an image and perform multi-modal reasoning for generating high quality captions.

Zhao *et al.* [39] proposed a dual learning method for cross-domain image captioning. The method uses dual learning to optimize two tasks: generating captions for images and generating images from generated captions. It uses a pre-trained model, which is trained on one dataset (MS COCO) and tested on another dataset (Flickr30k) for generating image captions.

In this paper, we use an attention-based GAN [40] to generate synthetic images from a given text and then we used these synthetic images together with the real images to train and test a baseline attention-based image captioning method. The motivation of the proposed method is to demonstrate the effectiveness of image captioning for synthetic images and to further improve the quality of the generated captions for real images.

III. MODEL ARCHITECTURE
Synthetic images are used for many deep learning-based applications for training. They are used for modeling various deep learning-based methods. In this paper, we propose a pipeline whose goal is to use both real and synthetic images to train and test an image captioning method. We use an automatic system to generate synthetic images. Generative Adversarial Network (GAN) has the popularity to be used for generating realistic synthetic images. To achieve our goal, we built a pipeline composed of a GAN Module to generate synthetic images and an image captioning module to generate captions.

A. GAN MODULE FOR SYNTHETIC IMAGE GENERATION
The GAN Module learns to generate synthetic images from an input text. In this method, we use AttnGAN [40] to generate synthetic images. AttnGAN has m gen-

erators $(G_0, G_1, \ldots, G_{m-1})$. They take the hidden states $(h_0, h_1, \ldots, h_{m-1})$ as input and then generate images of different scales, from small to large $(\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{m-1})$. Therefore,

$$
\begin{aligned}
h_0 &= F_0(z, F^{ca}(\bar{e})); \\
h_i &= F_i(h_{i-1}, F_i^{attn}(e, h_{i-1})); \\
\hat{x}_i &= G_i(h_i).
\end{aligned} \tag{1}
$$

where, z is a latent variable which is calculated from a standard normal distribution, e is word vector matrix, $\bar{e}$ is the sentence vector. F^{ca}, F_i, F_i^{attn}, and G_i are neural networks. The attention module takes two inputs: the image features from the previous hidden state and the word features. $e \in \mathbb{R}^{D \times T}$ and $h \in \mathbb{R}^{\hat{D} \times N}$ represent the word features and the image features from the previous hidden state, respectively. First, a multi-layer perceptron is used to transfer the word features into a common semantic space. Then based on the previous hidden state features h, a word context vector is computed to generate a region of an image. The context vector can be defined as:

$$
\hat{c}_j = \sum_{i=0}^{T-1} \beta_{j,i} \acute{e}_i, \quad \text{where } \beta_{j,i} = \frac{\exp(\acute{s}_{j,i})}{\sum_{k=0}^{T-1} \exp(\acute{s}_{j,k})}, \tag{2}
$$

In the above equation, $\beta_{j,i}$ represents the weight that the model uses to attend to the ith word when it generates the jth region of the image. In order to generate images at the next state, the image features and the corresponding word features are combined. Both the sentence level and the word level conditions are checked to generate the final synthetic image. The module has multiple stages to generate synthetic images. Initially it generates low-resolution images. Then high-resolution images are obtained by refining the low-resolution images in multiple steps through multiple generators and discriminators. The architecture of this network is similar to a tree structure. Different branches of the tree generate images of different resolutions: at branch i, the generator G_i learns the image distribution p_{G_i} at that scale, while the discriminator D_i estimates the probability of a sample being real. The discriminator D_i takes a real image x_i or a fake sample s_i as input and is trained to classify them as real or fake by minimizing the cross-entropy loss:

$$
\begin{aligned}
&\mathcal{L}_{D_i} \\
&= \underbrace{-\mathbb{E}_{x_i} \sim p_{data_i}[log D_i(x_i)] - \mathbb{E}_{x_i} \sim p_{G_i}[log(1 - D_i(s_i))]}_{\text{unconditional loss}} \\
&\quad + \underbrace{-\mathbb{E}_{x_i \sim p_{data_i}}[log D_i(x_i, c)] - \mathbb{E}_{x_i \sim p_{G_i}}[log(1 - D_i(s_i, c))]}_{\text{conditional loss}},
\end{aligned}
\tag{3}
$$

where x_i is an image from the true image distribution p_{data_i} at the ith scale, s_i is from the model distribution p_{G_i} at the same scale. StackGAN-v2 trains a text encoder [41] following the approach of Reed *et al.* [42]. The encoder is used to extract visually-discriminative text embeddings of the given description. Sentences that share semantic and

TABLE 1. Comparison of our different models with their generated captions on real images. The real images sample and their ground-truth captions are collected from the MS COCO dataset. 'R' means the image is from the original dataset, 'S(1)' means the synthetic images generated using the ground-truth caption 1, and 'S(all)' means the synthetic images generated from all the ground-truth captions. Images are best viewed in color.

Input Image	Output Captions
	Ground-Truth Captions: Soccer player wearing red and black shirt kicking at ball. **Generated Captions:** *(Train-R;Test-R (Baseline method)):* *A man playing a soccer ball on a field.* *(Train-S(1);Test-R):* *A man standing around soccer ball.* *(Train-R+S(1);Test-R):* *A man kicking a soccer ball on a soccer field.* *(Train-R+S(all);Test-R):* *A man in a soccer uniform playing soccer on a field.*
	Ground-Truth Captions: Woman talking on cell phone while wearing sun glasses.. **Generated Captions:** *(Train-R;Test-R (Baseline method)):* *A woman holding a cell phone in her hand.* *(Train-S(1);Test-R):* *A person talking on her cell phone.* *(Train-R+S(1);Test-R):* *A woman talking on a cell phone in the sun.* *(Train-R+S(all);Test-R):* *A woman wearing sunglasses talking on a cell phone.*
	Ground-Truth Captions: Bowl of broccoli on cutting board. **Generated Captions:** *(Train-R;Test-R (Baseline method)):* *A bowl of food with broccoli.* *(Train-S(1);Test-R):* *A lot of broccoli sitting in bowl.* *(Train-R+S(1);Test-R):* *A plate of food with broccoli on a table.* *(Train-R+S(all);Test-R):* *A white plate of food topped with broccoli on a board.*

syntactic properties are mapped to corresponding vector representations. The multiple discriminators and generators are trained to jointly approximate multi-scale image distributions $p_{data_0}, p_{data_1}, \ldots, p_{data_{m-1}}$ by minimizing the following loss function:

$$\mathcal{L}_G = \sum_{i=1}^{m} \mathcal{L}_{G_i} \mathcal{L}_{G_i} = \underbrace{-\mathbb{E}_{s_i} \sim p_{G_i}[log D_i(s_i)]}_{\text{unconditional loss}} + \underbrace{-\mathbb{E}_{s_i} \sim p_{G_i}[log D_i(s_i, c)]}_{\text{conditional loss}} \quad (4)$$

where L_{G_i} is the loss function for approximating the image distribution at the ith scale. The unconditional loss is used to determine whether the image is real or fake. In contrast, the conditional loss is used to determine if the image and the condition match.

B. IMAGE CAPTIONING MODULE

The goal of the Image Captioning Module is to generate a natural language description of an image. The model needs to understand the scene described in the image. It also needs to recognize the objects and their relationships taking part in image. Finally, the model composes a natural language sentence describing the whole picture. Given the complexity of such a task, it is still a challenging and open problem in the

TABLE 2. Comparison of our different models with their generated captions on synthetic images. The sample synthetic images are generated from the given text using an attention-based GAN model. 'R' means the image from the original dataset, 'S' means the synthetic images generated from the given text, 'S(1)' means the synthetic images generated using the ground-truth caption 1, and 'S(all)' means the synthetic images generated from all ground-truth captions. Images are best viewed in color.

Synthetic Image	Output Captions
	Text to Generate Image : Pizza covered in veggies on white plate sitting on table. **Generated Captions:** *(Train-R;Test-S):* *A pizza sitting on top of a white plate.* *(Train-S(1);Test-S):* *A pizza sitting on top of a wooden table.* *(Train-R+S(1);Test-S):* *Whole pizza with slices sits on pan on the table.* *(Train-R+S(all);Test-S):* *Cheese pizza with vegetables on top of a while plate on table.*
	Ground-Truth Captions: Close up view of banana sitting on top of a table. **Generated Captions:** *(Train-R;Test-S):* *A bunch of bananas sitting on a table .* *(Train-S(1);Test-S):* *A bunch of banana on a table .* *(Train-R+S(1);Test-S):* *A bunch of banana that are on a table.* *(Train-R+S(all);Test-S):* *A bunch of banana sitting on a top of a wooden table.*
	Ground-Truth Captions: Room with bed headboard two tables and comforter. **Generated Captions:** *(Train-R;Test-S):* *A living room with a bed and a table.* *(Train-S(1);Test-S):* *A hotel room with a bed and lamp.* *(Train-R+S(1);Test-S):* *A bedroom with a blanket and pillows on it.* *(Train-R+S(all);Test-S):* *A bedroom with a white comforter with pillows and a table.*

fields of NLP and computer vision. In our pipeline, we implement Image Captioning Module in a similar way as the one proposed in [43], meaning that we also use an attention-based captioning method based on FC models. Traditional convolutional networks with L layers have L connections. However, DenseNet has $L(L + 1)/2$ direct connections. As a result, the feature-maps of all preceding layers are used as inputs to the current layer, and its own feature-maps are used as inputs into all subsequent layers. The transformation function for DenseNet is:

$$I_l = H_l([I_0, I_1, \ldots, I_{l-1}]), \qquad (5)$$

where $[I_0, I_1, \ldots, I_{l-1}]$ refers to the concatenation of the feature-maps generated in layers $0, 1, \ldots, l - 1$ and $H_l(.)$ is a composite function.

The attention-based network can recompute its attention for the relevant parts of the image according to the perceived importance from LSTM. This recomputed image feature is a dynamic representation of the relevant parts of the image and is called a context vector ($\hat{z}_t$). Such a vector is computed from the annotation vector a_i defined in Equation (6) and the attention weight (α_{ti}). The attention weight is obtained from the alignment score (e_{ti}). The score defines how well each annotation vector matches with the previous hidden state output (h_{t-1}) of the LSTM decoder. Such an alignment score

is computed by applying an attention function (f_{att}):

$$e_{ti} = f_{\text{att}}(a_i, h_{t-1}), \qquad (6)$$

Next, the attention weight is obtained by normalizing e_{ti} using a Softmax function:

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})}, \qquad (7)$$

Then we compute the context vector ($\hat{z}_t$) using Equations (6) and (7) as follows:

$$\hat{z}_t = \phi(\{a_i\}, \{\alpha_i\}), \qquad (8)$$

We use soft attention [24] in our experiments, where (α_i) is first computed for each image region (x_i) and then the weighted average for (x_i) is calculated to use it as an input of LSTM. Hence the context vector $\hat{z}_t$ for soft attention can be written as:

$$E_{p(x_t|a)}[\hat{z}_t] = \sum_{i=1}^{L} \alpha_i, a_i, \qquad (9)$$

Finally, the LSTM is trained to compute the output word (s_t) probability condition on the context vector ($\hat{z}_t$) and the previously generated word s_{t-1} at time t. It is defined as:

$$P(s_0, s_1, \ldots, s_m) = \prod_{i=0}^{m} P(s_i | \hat{z}, s_0, s_1 \ldots, s_m), \qquad (10)$$

IV. EXPERIMENTS

In this section, we present the results of our experiments involving the proposed pipeline. Our pipeline has two main modules: (i) Text to Image synthesis and (ii) Image caption generation.

A. DATASET AND EXPERIMENTAL SETUP

1) DATASET

We use the large and popularly used MSCOCO dataset. This dataset consists of $82,783$ training and $40,504$ validation images. In our experiments we consider them as real images. In addition to these images, we also used synthetic images for our experiments. Image captioning datasets (e.g., MSCOCO that we used) have separate benchmark sets of images for training and testing. Each image has multiple ground truth captions. We used these captions to generate labelled synthetic images. We explicitly maintained the train and test split as marked in the dataset, i.e., if a synthetic image was generated from training image's ground-truth caption, that synthetic image was used in training only. On the other hand, if a synthetic image was generated from test image's ground-truth caption, that synthetic image was used in testing only.

2) IMPLEMENTATION DETAILS

For text to image generation we follow the implementation details of AttnGAN [40]. Two neural networks: (i) text encoder and (ii) image encoder are used here. A bi-directional LSTM model [50] is used to extract the semantic vectors from the given text descriptions. Thus each word gets the

context of two hidden states, one for the forward direction and one for the backward direction. These two hidden state vectors are concatenated to compute the overall context. The feature matrix for all words are computed by $e \in \mathbb{R}^{D \times T}$, where e_i represents the feature vector of ith word. D and T indicates the dimension of the vector and the total number of words, respectively. A CNN, Inception-v3 model [51] is used to extract the image feature vectors. Two types of Features namely, (i) local features of different image regions and (ii) global features of the image are extracted from the intermediate layer and the last average pooling layer, respectively. The size of the local feature matrix is $e \in \mathbb{R}^{768 \times 289}$. Here, 768 denotes the dimension of the local feature vector and 289 represents the number of sub-regions in the image. On the other hand, the size of the global feature vector is $f \in \mathbb{R}^{2048}$. Finally, a perceptron layer is used to map the image features to the semantic space of the text features.

For the image in the captioning module, we use DenseNet121 [8] with fully connected layers to extract image features. DenseNet121 is pre-trained on ImageNet dataset. We apply the fc7 feature map to compute the attention features. The dimension of our feature map is 1×1024. The size of the hidden layer in the prediction module is 1024. We apply dropout, a learning rate of 0.001 and use a linear layer to obtain a 512-dimensional word embedding. We also apply Adam optimizer with a mini-batch size 16 to train the model. Text to image generation module is implemented in Pytorch and image captioning module is implemented in Tensorflow. We used an existing PyTorch code and customise it for the image generation module, while the image captioning modules were mostly built by us on TensorFlow. The models are trained for 100 epochs on the training dataset, which took 28 days on TITAN Xp GPUs. Note, it includes the computationally heavy synthetic image generation part. Once the synthetic images are generated, it is computationally reasonable to train the model. Testing per image took approximately 30 ms.

3) COMPARED MODELS

We demonstrate our results using qualitative analysis reported in Tables 1 and 2. In both cases, we compare the different models between them and with one baseline model. In addition, we quantitatively compare our models with other state-of-the-art image captioning models such as DeepVS [21], m-RNN [20], Google NIC [23], LRCN [46], hard-ATT [24], soft-ATT [24], and ConvCap [48] The results are shown in Table 3.

B. ANALYSIS OF RESULT

We discuss and analyze both qualitative and quantitative results of the generated captions.

1) QUALITATIVE ANALYSIS

We used both real and synthetic images for the training and testing of our different models. Next, we have generated

TABLE 3. Performance of our models in comparison with other state-of-the-art techniques. Bold indicates the best results and a dash(-) indicates that results are unavailable.

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGH-L	CIDEr-D
DeepVS [21]	62.5	45.0	32.1	23.0	19.5	-	66.0
m-RNN [20]	67.0	49.0	35.0	25.0	-	-	-
NIC [23]	66.6	46.1	32.9	24.6	-	-	-
g-LSTM [44]	67	49.1	35.8	26.4	23.9	-	-
Bi-LSTM-M [45]	68.7	50.9	36.4	25.8	22.9	-	73.9
LRCN [46]	69.7	51.9	38.0	27.8	22.9	50.8	83.7
Hard-ATT [24]	71.8	50.4	35.7	25.0	23.0	-	-
Soft-ATT [24]	70.7	49.2	34.4	24.3	23.9	-	-
ATT-FCN [47]	70.9	53.7	40.2	30.4	23.9	-	-
ConvCap [48]	69.3	51.8	37.4	26.8	23.8	51.1	85.5
COMIC [49]	70.6	53.4	39.5	29.2	23.7	51.7	88.1
Ours(Train-R;Test-R; Baseline method)	68.0	47.4	32.5	22.9	23.7	48.9	85.4
Ours(Train-S(1);Test-S(1))	62.7	44.4	31.1	22.0	20.9	46.3	73.2
Ours(Train-S(1);Test-R)	63.4	45.0	31.5	22.5	21.7	46.8	74.4
Ours(Train-R;Test-S(1))	66.5	46.1	34.4	23.8	23.1	47.6	76.8
Ours(Train-R+S(1);Test-S(1))	68.2	47.5	35.1	24.3	23.9	49.4	86.1
Ours(Train-R+S(1);Test-R)	71.1	53.5	40.3	30.0	24.6	52.3	91.2
Ours(Train-R+S(all);Test-S(all))	71.9	52.9	43.2	31.5	25.3	53.6	101.4
Ours(Train-R+S(all);Test-R)	**73.6**	**54.7**	**44.2**	**33.6**	**25.7**	**54.8**	**105.6**

captions for these images with these models. Then we have analyzed and compared the generated captions with a baseline method and between our different models. The generated captions in Table 1 are only on real images. However, the models "Train-R;Test-R" and "Train-S(1);Test-R" are trained on real images and synthetic images (the synthetic images are generated from each caption #1, for its corresponding real images), respectively. Next, these synthetic images together with the real image are used to train the model "Train-R + S(1);Test-R". Finally, all the synthetic images (the synthetic images are generated from each five captions of the corresponding real images) together with the real images are used to train the "Train-R + S(all);Test-R" model. Here, the model "Train-R;Test-R" is considered to be baseline method. It can be seen from Table 1 that we get longer and semantically more accurate captions when we use both real and synthetic images for training. In the first example of this table, the baseline method does not generate anything about the "jersey" of the soccer player. However, the model "Train-R + S(all);Test-R" picks this information as "uniform" successfully. Although the model "Train-R + S(1)" does not include anything about the soccer player's cloths, it picks the word "kick" which is present in the ground-truth caption. Following the example one, the "Train-R + S(all);Test-R" model successfully includes "sun glass" and "board" in the generated captions of the second and third examples, respectively. However, these words are missing in the baseline method's generated captions. Similarly, for the second and third examples, the "Train-R + S(1)" model generates captions that are closer to the ground-truth captions and these captions are semantically more accurate than the ones from the baseline method. It is also seen that the model "Train-S(1);Test-R" which is solely trained on synthetic images generates semantically weaker captions than the other models.

We illustrated the generated captions of synthetic images in Table 2. Since the synthetic images are very different from the real images, we do not compare the generated captions of the synthetic images with the real ones. In Table 2, we analyze and compare the generated captions of the synthetic images between our different models along with the corresponding text used to generate the synthetic images. The models "Train-R + S(1)" and "Train-R + S(all)" generate reasonably better captions than other models and they are closer to the input text as well. The model "Train-R + S(all); Test-S" includes few words such as "vegetables", "top", and "white" in its generated captions of example one. It can be seen that the generated captions are longer and semantically richer than those of other models. Similarly, "top" in the second example and "white comforter", "pillows", and "table" in the third example are appropriate pick by this model. It is also seen in all three examples that the generated captions by the model "Train-R + S(1);Test-S" are semantically more accurate than those of the models "Train-R;Test-S" and "Train-S;Test-S".

2) QUANTITATIVE ANALYSIS

Table 3 shows the results of the generated captions with our different models on BLEU-1, BLEU-2, BLEU-3, and BLEU-4 evaluation metrics. In order to demonstrate our results, we use the soft attention method proposed by Xu *et al.* [24]. However, we use DenseNet instead of VGGNet to extract visual features from images. Xu et al. reported 70.7, 49.2, 34.4, 24.3, and 23.90 scores for BLEU-1, 2, 3, 4, and METEOR respectively, for soft attention in their paper. However, we use the code of Yunjei available in GitHub and the scores we got are 67.7, 46.1, 32.3, and 22.4. We achieved slightly better results on DenseNet as reported in Table 3 and we considered it as our baseline method. In terms of scores of the evaluation metrics, the mod-

els which use both real and synthetic images for training achieve superior results than other models. BLEU metrics work by counting the matching n-grams in the generated captions to the n-grams of the ground-truth captions. Therefore, It can be seen from Table 3 that the generated captions with some of our models can match better than the baseline method and some other state-of-art methods. For example, the model "Train-R + S(all);Test-R" achieves 73.6, 54.7, 44.2, 33.6, 25.7, 54.8, and 105 in BLEU-1, 2, 3, 4, METEOR, ROUGE-L, and CIDEr, respectively and outperforms all the other methods. On the other hand, the models which use only synthetic images for training achieve poor results. For example, the model "Train-R;Test-S(1)" achieves 66.5, 46.1, 34.4, 23.8, 23.1, 47.6, and 76.8 in BLEU-1, 2, 3, 4, METEOR, ROUGE-L, and CIDEr respectively, which are inferior to the corresponding scores of the base line method and other state-of-the-art methods.

V. CONCLUSION

Image Captioning is vital for several reasons. Automatic image captioning can be useful for assisting visually impaired people, intelligent human computer interactions, and developing image search engines. Social media platforms such as Facebook and Twitter can directly generate descriptions from the image, where we are (beach, cafe), what we wear and importantly what are we doing there. Nowadays, machine generated synthetic images are becoming available more and more, e.g., for news, illustration, artwork, promotion, as well as for human computer interaction and augmented reality. There are challenges e.g., DeepFake to distinguish between real and fake ideas. In this paper, we have investigated the effectiveness of synthetic images in image captioning. For this task, we built a pipeline to first generate synthetic images from text using an attention based generative adversarial network. This gives us a synthetic image dataset with ground truth captions. Then we used these synthetic images together with the real images to train and test an image captioning model. We have demonstrated that the models, which use both real and synthetic images for training achieve superior performances compared to the baseline method and other state-of-the-art methods. In this paper, we used synthetic images generated from text only. However, image synthesis from real images can be a future work. Furthermore, the use of synthetic caption for improved image captioning can also be a potential extension of this work.

REFERENCES

[1] L. White, R. Togneri, W. Liu, and M. Bennamoun, *Neural Representations of Natural Language*, vol. 783. Singapore: Springer, 2018.

[2] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[3] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, "A guide to convolutional neural networks for computer vision," *Synth. Lectures Comput. Vis.*, vol. 8, no. 1, pp. 1–207, Feb. 2018.

[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[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, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[8] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.

[9] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, H. Laga, and M. Bennamoun, "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, C. Zhang, T. Zhou, and Y. Quan, "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, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "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, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "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, and J. Hockenmaier, "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, P. Wohlhart, and K. Konolige, "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, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4040–4048.

[16] J. Borrego, A. Dehban, R. Figueiredo, P. Moreno, A. Bernardino, and J. Santos-Victor, "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, J. Materzynska, D. Vazquez, and A. M. Lopez, "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, M. F. Shiratuddin, and H. Laga, "A comprehensive survey of deep learning for image captioning," *ACM Comput. Surv.*, vol. 51, no. 6, p. 118, Feb. 2019.

[19] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth, "Every picture tells a story: Generating sentences from images," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 15–29.

[20] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (M-RNN)," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–17.

[21] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3128–3137.

[22] R. Bernardi, R. Cakici, D. Elliott, A. Erdem, E. Erdem, N. Ikizler-Cinbis, F. Keller, A. Muscat, and B. Plank, "Automatic description generation from images: A survey of models, datasets, and evaluation measures," *J. Artif. Intell. Res.*, vol. 55, pp. 409–442, Feb. 2016.

[23] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3156–3164.

[24] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.

[25] S. He, H. R. Tavakoli, A. Borji, and N. Pugeault, "Human attention in image captioning: Dataset and analysis," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 8529–8538.

[26] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "FlowNet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2758–2766.

[27] T. Qiao, J. Zhang, D. Xu, and D. Tao, "MirrorGAN: Learning text-to-image generation by redescription," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1505–1514.

[28] B. Dai, S. Fidler, R. Urtasun, and D. Lin, "Towards diverse and natural image descriptions via a conditional GAN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2970–2979.

[29] C. Chen, S. Mu, W. Xiao, Z. Ye, L. Wu, and Q. Ju, "Improving image captioning with conditional generative adversarial nets," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8142–8150.

[30] W. Jiang, L. Ma, Y.-G. Jiang, W. Liu, and T. Zhang, "Recurrent fusion network for image captioning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 499–515.

[31] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang, "Scene graph generation from objects, phrases and region captions," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 1261–1270.

[32] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei, "Visual relationship detection with language priors," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 852–869.

[33] 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.

[34] L. Huang, W. Wang, J. Chen, and X.-Y. Wei, "Attention on attention for image captioning," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 4634–4643.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[36] L. Ke, W. Pei, R. Li, X. Shen, and Y.-W. Tai, "Reflective decoding network for image captioning," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 8888–8897.

[37] M. Cornia, M. Stefanini, L. Baraldi, and R. Cucchiara, "Meshed-memory transformer for image captioning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 10578–10587.

[38] Y. Pan, T. Yao, Y. Li, and T. Mei, "X-linear attention networks for image captioning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 10971–10980.

[39] W. Zhao, W. Xu, M. Yang, J. Ye, Z. Zhao, Y. Feng, and Y. Qiao, "Dual learning for cross-domain image captioning," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 29–38.

[40] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1316–1324.

[41] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "StackGAN++: Realistic image synthesis with stacked generative adversarial networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1947–1962, Aug. 2019.

[42] S. Reed, Z. Akata, H. Lee, and B. Schiele, "Learning deep representations of fine-grained visual descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 49–58.

[43] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, H. Laga, and M. Bennamoun, "Attention-based image captioning using DenseNet features," in *Proc. Int. Conf. Neural Inf. Process. (ICONIP)*, 2019, pp. 109–117.

[44] X. Jia, E. Gavves, B. Fernando, and T. Tuytelaars, "Guiding the long-short term memory model for image caption generation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2407–2415.

[45] C. Wang, H. Yang, and C. Meinel, "Image captioning with deep bidirectional LSTMs and multi-task learning," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 14, no. 2, p. 40, 2018.

[46] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, T. Darrell, and K. Saenko, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2625–2634.

[47] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4651–4659.

[48] J. Aneja, A. Deshpande, and A. G. Schwing, "Convolutional image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5561–5570.

[49] J. H. Tan, C. S. Chan, and J. H. Chuah, "COMIC: Toward a compact image captioning model with attention," *IEEE Trans. Multimedia*, vol. 21, no. 10, pp. 2686–2696, Oct. 2019.

[50] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[51] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2818–2826.

MD. ZAKIR HOSSAIN (Student Member, IEEE) received the bachelor's and master's degrees in computer science and engineering from Jahangirnagar University, Bangladesh. He is currently pursuing the Ph.D. degree with Murdoch University, Australia. His research interests include computer vision, machine learning, and natural language understanding.

FERDOUS SOHEL (Senior Member, IEEE) received the Ph.D. degree from Monash University, Australia. He is currently an Associate Professor in information technology with Murdoch University, Australia. Prior to joining Murdoch University, in 2015, he was a Research Fellow with the School of Computer Science and Software Engineering, The University of Western Australia, from January 2008 to June 2015. His research interests include computer vision, image processing, machine learning, pattern recognition, sound event detection, remote sensing, digital agritech, retinal imaging, cyber forensics, and video coding. He served as an international program committee member of many conferences. He is also a member of the Australian Computer Society. He was a recipient of the Best Ph.D. Thesis Medal from Monash University. He has received several best paper awards. He was a Co-Presenter of a tutorial at CVPR2015. He was the Technical Program Chair of DICTA2019, an Organizing Secretary of APCC2017, and the Tutorial Chair of PSIVT2019. He is also an Associate Editor of IEEE Transactions on Multimedia, from 2020 to 2021, and IEEE Signal Processing Letters.

MOHD FAIRUZ SHIRATUDDIN received the B.Eng. degree in electrical and electronics from Northumbria University, U.K., the M.Sc. degree in information technology (virtual reality) from University Utara Malaysia, and the M.S. degree in architecture (construction management) and the Ph.D. degree in environmental design and planning from Virginia Tech, USA. He is currently a Senior Lecturer with Murdoch University, Australia. In his early career, he was trained in the U.K. and Malaysia as an Electrical and Electronics Engineer mainly dealing with computers, high-speed Internet communication technologies, and their applications. He then decided to pursue a career in academia. His main areas of research and teaching are in eXtended Reality (XR = virtual/mixed/augmented reality), natural user interfaces, games design, and the development and technologies for practical and real-world purposes.

HAMID LAGA received the Ph.D. degree in computer science from the Tokyo Institute of Technology, in 2006. He is currently an Associate Professor with Murdoch University, Australia, and an Adjunct Associate Professor with the Phenomics and Bioinformatics Research Centre (PBRC), University of South Australia. His research interests include machine learning, computer vision, computer graphics, and pattern recognition, with a special focus on the 3D reconstruction, modeling and analysis of static and deformable 3-D objects, image analysis, and big data in agriculture and health. He was the recipient of the Best Paper Award at SGP2017, DICTA2012, and SMI2006.

MOHAMMED BENNAMOUN (Senior Member, IEEE) is currently a Winthrop Professor with the Department of Computer Science and Software Engineering, UWA, and a Researcher in computer vision, machine/deep learning, robotics, and signal/speech processing. He has published four books (available on Amazon), one edited book, one Encyclopedia article (by invitation), 14 book chapters, more than 120 journal articles, more than 250 conference publications, and 16 invited and keynote publications. His H-index is 48 and his number of citations is close to 11 000 (Google Scholar). He was awarded more than 65 competitive research grants (approximately more than 17 million in funding) from the Australian Research Council, numerous other governments and UWA, and industry research grants. He has delivered conference tutorials at major conferences, including the IEEE Computer Vision and Pattern Recognition (CVPR 2016), the Interspeech 2014, the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), and the European Conference on Computer Vision (ECCV). He was invited to give a tutorial at an International Summer School on Deep Learning (DeepLearn 2017). He widely collaborated with researchers from within Australia (e.g., CSIRO) and internationally (e.g., Germany, France, Finland, and the USA). He served for two terms (three years each term) with the Australian Research Council (ARC) College of Experts and the ARC ERA 2018 (Excellence in Research for Australia).

• • •