IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 28, 2021, accepted November 12, 2021, date of publication November 15, 2021,
date of current version November 23, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3128140

A Systematic Survey of Remote Sensing Image Captioning

BEIGENG ZHAO
College of Public Security Information Technology and Intelligence, Criminal Investigation Police University of China, Shenyang 110035, China
e-mail: zhaobeigeng@qq.com

This work was supported in part by the Scientific Research Fund of the Criminal Investigation Police University of China under Grant D2020032, and in part by the Fundamental Research Funds for the Central Universities under Grant D2019023.

ABSTRACT Image captioning is a cross-disciplinary task to automatically generate textural descriptions for a given image using computer vision and natural language processing techniques. Remote sensing image captioning refers to the application of this task to remote sensing images taken from high altitude by satellites, aircraft or drones. This interesting and valuable topic has only emerged in recent years and attracted considerable research attention. There has been extensive related work in the literature, with considerable results and an independent body of research, and various issues must be addressed in future work. However, to the best of our knowledge, there has been no review study in this area that can provide researchers with systematic reference information, which is the motivation of this study. To achieve this goal, 30 relevant articles were conditionally filtered and obtained for the review study. We analyzed and summarized the existing work from various perspectives, including technical solutions, data, evaluation metrics, and the experimental results of state-of-the-art methods. Based on this summary, the trends, pros and cons of the existing studies, issues to be addressed and valuable research directions in future work are discussed. The results of this paper can provide valuable reference information for researchers in related fields.

INDEX TERMS Image captioning, remote sensing, deep learning, natural language processing.

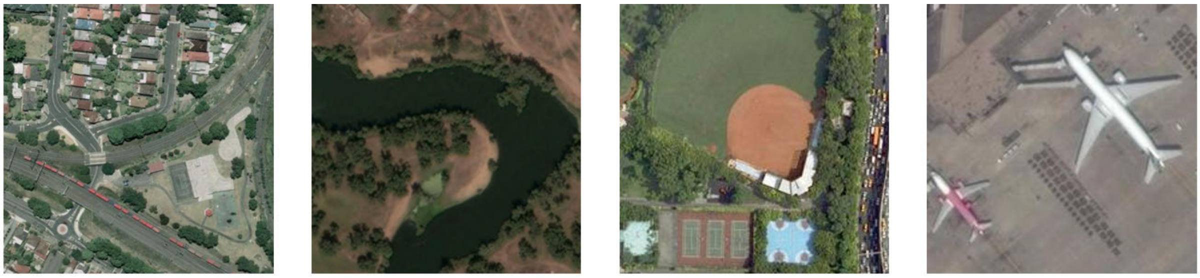
I. INTRODUCTION

Image captioning [89]–[92] is a cross-disciplinary topic covering computer vision, natural language processing and deep learning, with the goal of recognizing the content of a given image and generating a descriptive text through computer technology. Remote sensing images (RSIs) [93], [94] refer to images containing geographic information captured by satellites, aircraft, drones, etc. from high altitude overhead. Remote sensing image captioning (RSIC) is a combination of the above two concepts, i.e., generating textual descriptions for RSIs, describing the scenes and ground objects in the image, as well as their attributes and relationships. Figure 1 shows some examples of RSIC. This research direction has very high potential for application [2], such as in generating real-time text or voice descriptions for photos taken by unmanned aerial vehicles (UAVs) in war, reconnaissance, traffic command and rescue scenarios.

Many researchers have worked on making machines better understand RSIs through studies such as scene classification and object detection [83]–[87], [93], [94]. Compared to these studies, RSIC focuses on providing higher-level semantics and human-readable descriptions. In addition, while many methods in natural image captioning (NIC) studies [71], [72], [75], [76], [89]–[92] can be applied to RSIC research, some unique challenges need to be addressed in this field. For example, natural images have a clear viewing direction, i.e., sky at the top and earth at the bottom, while RSIs taken from ‘‘God’s perspective’’ do not have a fixed viewing direction [3]. Another example is that the scales of objects in RSIs vary greatly, resulting in the same objects having completely different sizes and appearances in different images [22].

Since the first RSIC study [1] was proposed, in recent years, an increasing number of scholars have devoted themselves to contributing technical solutions, large-scale datasets and ideas with potential application value to this new research field. The related data, methods, and evaluation metrics have gradually formed an independent system different from other fields. However, to the best of our knowledge, there are no review articles focusing on RSIC in the literature, which can provide researchers with a systematic analysis of the research status, trends, challenges and future work in this field.

The associate editor coordinating the review of this manuscript and approving it for publication was Wei-Yen Hsu.



A residential area with houses arranged neatly while some roads and railways go through.

many green plants and barelands are in two sides of a curved green river .

A baseball field surrounded by many green plants is next to a crossroads.

A red and white small plane and a silver-white plane are parked at their gate .

FIGURE 1. **Examples of remote sensing image captioning.**

To fill this gap, in this article, we present a systematic review of the RSIC work in the literature. Our contributions are mainly as follows. First, we collected articles related to RSIC on Web of Science and Google Scholar and classified them into different categories based on technical solutions and then discussed the characteristics, theoretical foundations, and pros and cons in detail. Second, we discussed the benchmark RSIC datasets and commonly used evaluation metrics and analyzed and compared the performance of the state-of-the-art methods to summarize the research trends. In addition, we briefly discussed the outline and the limitations of the existing work and the future research directions in this field. To the best of our knowledge, this is the first systematic survey of RSIC and can be expected to provide researchers with a strategic and detailed reference in this area.

II. BACKGROUND

Image captioning is an interesting research direction to auto-generate descriptive sentences for a given image to describe the objects that appears in the image, their attributes and relationships, etc. Conventional image captioning [89]–[92], which is also called NIC, focuses on generating descriptions for natural images, i.e. photographic images of everyday life scenes.

According to the method to generate sentences, there are mainly three categories of NIC methods [89]: template-based methods, retrieval-based methods, and sequence generation-based methods. The template-based methods require artificially designing sentence templates with fixed structures containing blanks and filling in the blanks with recognized objects, attributes and other information through image recognition technology. A retrieval-based method requires maintaining a large-scale database that contains images and the corresponding descriptions. The image in the database that is most similar to the input image is retrieved, and its corresponding descriptions are output as the captions. Although template- and retrieval-based approaches can generate syntactically and grammatically correct sentences and are suitable for some scenarios, they cannot generate flexible and variable descriptions. This issue can be solved by sequence-generation-based methods. These kinds of approaches treat sentences as sequences of words and train models to learn not only the correspondence between image features and words, but also the sequential relationships between adjacent words. Then, the trained models can generate flexible and variable word sequences to describe the input images. The drawback of sequence generation-based methods is that the generated sentences may have grammatical or syntactic errors.

Although research for NIC can be traced back over a relatively long time, research focusing on RSIC has only started in recent years. To the best of our knowledge, the first study for RSIC was proposed by Qu *et al.* [1] in 2016. Qu *et al.* borrowed the state-of-the-art models from NIC, extracted features from RSIs using convolutional neural networks (CNNs) [31] and generated captions using recurrent neural networks (RNNs) [67]. In this work, the first two benchmark RSIC datasets, namely Sydney-captions and UCM-captions were also annotated. It is worth noting that at the beginning of the first RSIC study, deep learning methods had been widely used in computer vision and natural language processing. Therefore, unlike the situation in the field of NIC, RSIC researchers have made extensive use of deep learning methods such as CNNs, RNNs from the very beginning. An early RSIC study by Lu *et al.* [3] compared non-deep learning methods with deep learning methods and showed that deep learning methods were far superior. This fact was carried forward by subsequent studies; since then, researchers have only focused on deep learning-based methods.

Inspired by the early studies, many researchers have devoted themselves to the field of RSIC. Unlike natural images whose contents are mainly everyday scenes taken by regular cameras, RSIs that are taken by satellites, aircrafts or UAVs from high altitude have unique characteristics. For example, RSIs taken from "God's view" do not have a clear direction of observation with the sky above and the

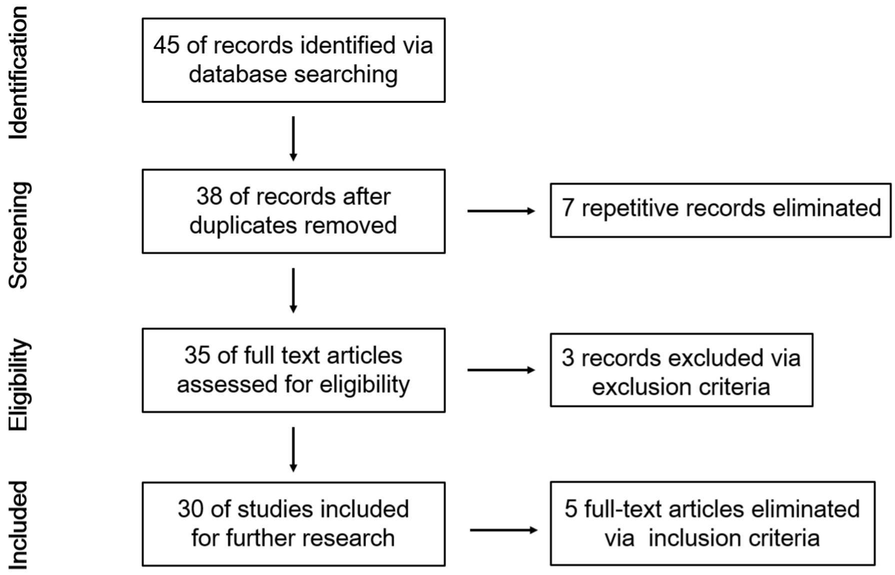

Identification
Screening
Eligibility
Included
45 of records identified via database searching
38 of records after duplicates removed
7 repetitive records eliminated
35 of full text articles assessed for eligibility
3 records excluded via exclusion criteria
30 of studies included for further research
5 full-text articles eliminated via inclusion criteria

FIGURE 2. **Process of extracting relevant papers.**

earth below [3], as ordinary photographs do. The scale of the same type of objects in different RSIs can vary [22]. In response to these unique characteristics, many excellent RSIC studies have emerged in recent years and gradually formed an independent body of research. These studies have achieved impressive results, but many issues must be addressed in future work. However, to the best of our knowledge, no review articles on RSIC can provide researchers with a systematic analysis of this field, which motivated us for this study.

Many excellent review studies have focused on image captioning and RSI processing in the existing literature. In the studies of Bernardi *et al.* [91] and Bai *et al.* [90], different categories of models and representation spaces used to implement image captioning tasks were reviewed in detail. A review study of the evolution of image captioning solutions was given by Kumar *et al.* [92] in a chronological manner. Based on these studies, Hossain *et al.* [89] focused on reviewing deep learning-based image captioning solutions. In the field of RSI processing, review studies of RSI classification were conducted by Li *et al.* [95] and Song *et al.* [96]. Cheng *et al.* [97] focused on a review study of object detection methods for RSIs. Abdollahi *et al.* [94] presented a review study of road extraction methods based on deep learning and remote sensing techniques. Ma *et al.* [93] conducted a meta-analysis study on applications of deep learning-based techniques in the field of remote sensing.

In this paper, we draw on the valuable methods and information from the abovementioned review studies. Also, unlike these studies, we focus solely on the RSIC domain, providing a meta-analysis and summary of the unique models, data, issues and solutions within it. Compared with review studies of NIC [89]–[91], we focus on captioning solutions for RSIs rather than other types of images. In contrast to other RSI-related reviews [93]–[97], our research revolves around image captioning rather than other tasks. More importantly, most of the articles screened and reviewed in our work do not appear in the abovementioned review studies. Therefore, our work can be expected to provide unique and valuable review information for researchers in RSIC and related fields.

III. METHODOLOGY

To obtain high-quality papers related to RSIC for the review study, we followed the process specified by the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) statement [98], as shown in Figure 2. Exclusion and inclusion criteria were developed to filter the record objects and full-text articles. The exclusion criteria are as follows:

- Not peer-reviewed.
- The full text is not available from the publisher.

The inclusion criteria are as follows:

- The article is written in English.
- The images used in the study must be RSIs and no other types of images.
- RSIC must be the primary research goal, not a way to enhance the performance of other tasks, such as RSI retrieval.

In the identification phase, the databases of Web of Science and Google Scholar were utilized to search for and initially access records in the literature. Specifically, we searched all articles to date (search date: October 20, 2021) using the expressions ‘‘remote sensing image captioning’’, ‘‘remote sensing image description’’, ‘‘remote sensing caption generation’’ and ‘‘remote sensing image description generation’’.

In the screening phases, duplicate records were removed and further filtered by exclusion criteria. In the eligibility stage, the full-text papers related to the above obtained records were further filtered by the inclusion condition, and 30 articles were ultimately included for further study.

Then, the extracted articles were classified into different categories according to the technical solutions. We analyzed the theoretical foundations and pros and cons of the studies in each group in Section IV. Then, the datasets and automatic evaluation metrics in the literature are summarized and analyzed in Section V. Based on this information, we compared and analyzed the experimental results of state-of-the-art methods in this field in Section VI and discussed the existing research and future work in Section VII.

IV. TECHNICAL SOLUTIONS FOR RSIC

In this section, methods in the literature for RSIC are grouped into seven categories, i.e., encoder-decoder architecture, image feature extraction, attention mechanism, language model (LM), training strategy, active attention and auxiliary component, according to the technical solutions. We discuss each category of these methods in detail in the following subsections.

Figure 3 illustrates the overall summary of all technical solutions. The majority of RSIC studies follow an encoder-decoder architecture. This architecture contains an encoder for extracting image features and a decoder for translating image features into textual descriptions. Technical solutions for image feature extraction are concerned with how to efficiently extract valuable information from RSIs. The work of LMs is to generate each descriptive word based on the image features and contextual information obtained from other modules. When generating each word, an attention mechanism can output an attention mask telling the model to focus on a specific region in the image. For example, when generating the word "plane", a well-trained attention mechanism can generate the mask associated with the plane in the image. During training, the training strategy optimizes the parameters of the RSIC models based on the differences between the generated sentences and the annotated sentences. Active attention technology uses additional information such as sound or topic information to guide the model to generate sentences of interest. Technical solutions of auxiliary components refer to enhancing the overall performance of RSIC systems by designing auxiliary modules that can be embedded into any encoder-decoder architecture, such as the persistent memory mechanism [21], which can enhance the performance of LMs, and the graph convolutional network (GCN)-based module [11], which can capture relations of objects and attributes in RSIs.

Each technical solution will be discussed in detail separately in Section IV-A to IV-G.

A. ENCODER–DECODER ARCHITECTURE

Qu *et al.* [1] pioneered the task of RSIC. These researchers proposed an encoder-decoder model, as shown in Figure 4, based on Vinyals *et al.*'s method [71]. The encoder extracts features from the input RSI, and the decoder translates the extracted image features into textual descriptions. The structure of a CNN [31], which is suitable for extracting image features and successful in large-scale image classification tasks, is used as the backbone for the implementation of the encoder. The decoder is based on an RNN [67], i.e., a neural network that can efficiently learn the sequence relationships in text data.

I is denoted as an RSI, and S is denoted as its corresponding annotated sentence in the training set. The goal of training is to maximize the probability of generating a correct sentence given an RSI:

$$\theta^* = \arg\max_{\theta} \sum_{(I,S)} \log p(S|I;\theta) \tag{1}$$

where θ are the parameters of the RISC model. Since S represents a sentence containing a variable number of words, the above equation can be expressed according to the chain rule as:

$$\log p(S|I) = \sum_{t=0}^{N} \log p(w_t|I, w_0, \ldots, w_{t-1}) \tag{2}$$

where w_t represents the t-th word in S and N is the length of the sentence. Here, we omit the parameter θ. In [1], $p(w_t|I, w_0, \ldots, w_{t-1})$ is modeled with an RNN, where the next output h_{t+1} of each time step is calculated using the previous output h_t and the new input x_t:

$$h_{t+1} = f(h_t, x_t) \tag{3}$$

where the exact form of f can be a specific RNN variant, such as a long short-term memory (LSTM), a gated recurrent unit or a vanilla RNN, and x_t is calculated using the image features extracted by the CNN encoder and the previously generated word:

$$x_{-1} = CNN(I) \tag{4}$$

$$x_t = W_e w_t \tag{5}$$

where w_t is the one-hot vector of the word generated at time step t and is embedded into a space with the same dimension as the image representation $CNN(I)$ via W_e. Regarding the concrete implementation of f in Equation (3), the experimental results in [1] show that LSTM achieves significantly superior performance in the RSIC task compared to the vanilla RNN. LSTM is a variant of the RNN proposed by Hochreiter *et al.* [68] to solve the problem of vanishing and exploding gradients in RNN model training and has achieved great success in the task of machine translation and sequence generation [40], [69], [70].

The core of LSTM is the memory cell encoding the knowledge in each time step. Three different gates are designed to control the behavior of the memory cell:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1}) \tag{6}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1}) \tag{7}$$

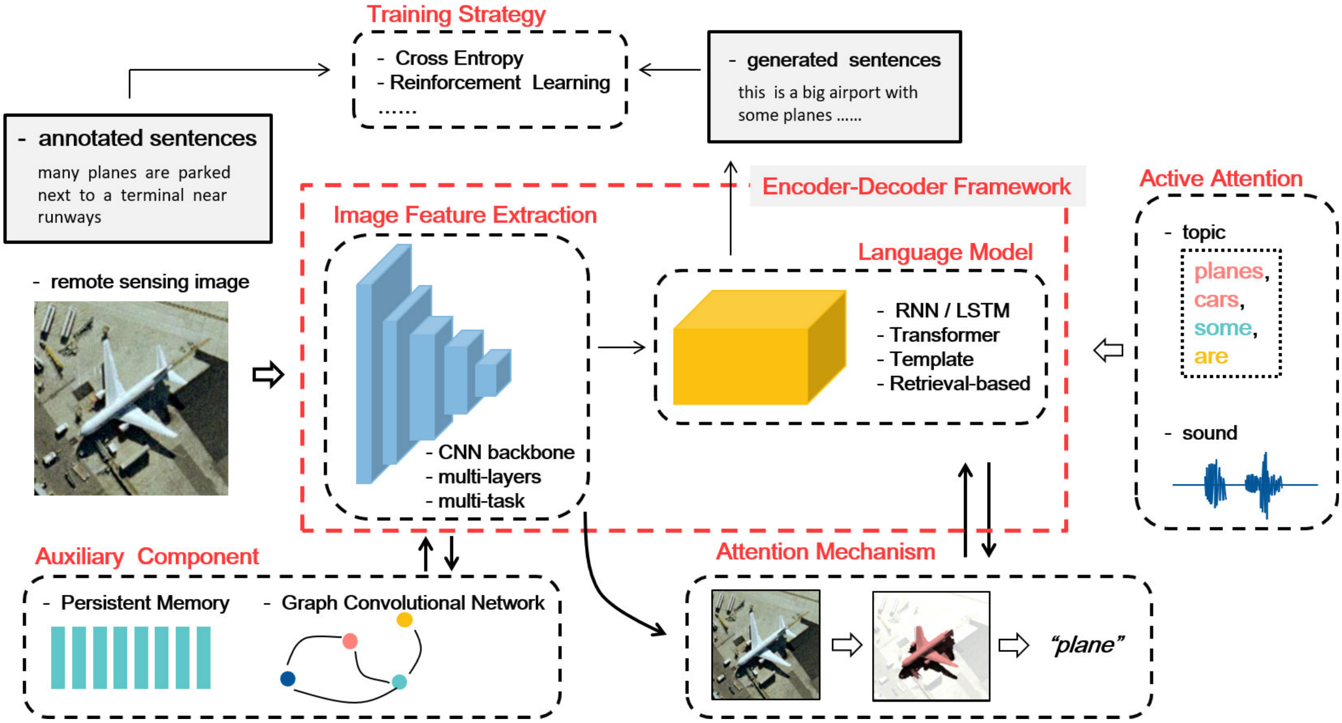


FIGURE 3. Overall summary diagram of technology solutions for RSIC.

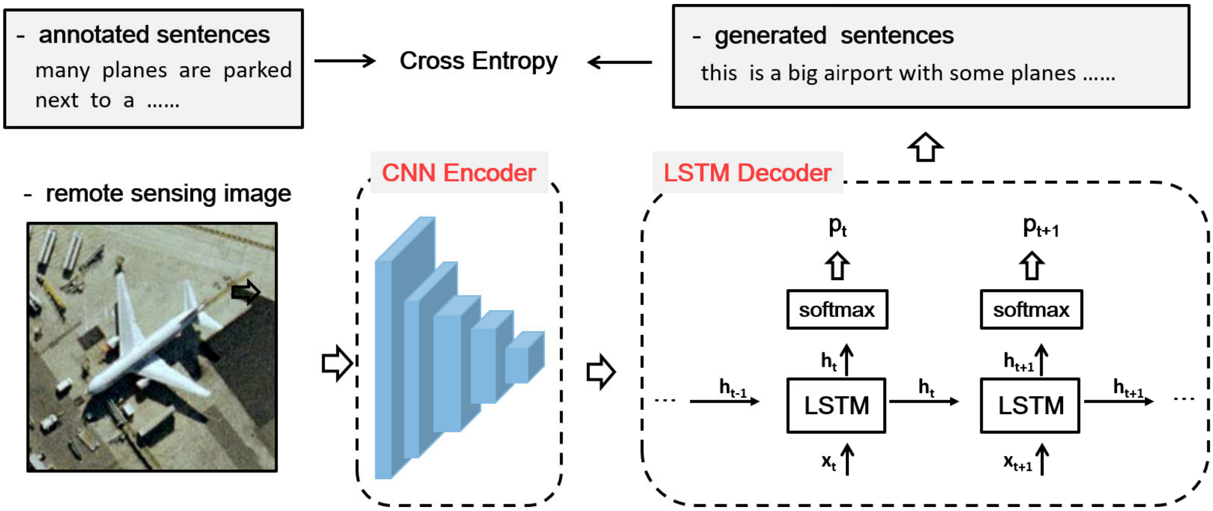


FIGURE 4. Vanilla encoder–decoder framework for RSIC.

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1}) \tag{8}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hi}h_{t-1}) \tag{9}$$

$$h_t = o_t \odot c_t \tag{10}$$

$$p_{t+1} = \mathrm{softmax}(h_t) \tag{11}$$

where i_t, f_t, o_t denote the input, forget and output gates, respectively, $\odot$ represents the product operation, and the various W matrices are trainable parameters. Figure 4 shows the LSTM decoder in an unrolled form. All LSTMs in the figure share the same parameters, and the output h_{t-1} of the LSTM at time $t-1$ is fed to the LSTM at time t.

During training, the overall loss value is obtained by summing the negative log-likelihood of the correct word in each time step:

$$L(I, S) = -\sum_{t=1}^{N} \log p_t(w_t). \tag{12}$$

The goal of training is to fine-tune the parameters of the CNN encoder and the LSTM decoder to minimize the loss value in the above equation. To train and test the proposed RSIC model, Qu *et al.* annotated two new datasets using the RSIs in the University of California Merced Land-Use (UCM) [50] and Sydney [49] datasets. The newly built datasets, named UCM-captions and Sydney-captions, are the first two image captioning datasets for remote sensing, consisting of not only RSIs but also five textual descriptions for each image.

Inspired by Qu *et al.*'s work, many researchers have devoted themselves to the study of RSIC. In the vast majority of RSIC studies, CNN-based encoders and RNN (usually LSTM)-based decoders have been employed as the backbone of the overall architecture. Around this kind of ''encoder-decoder architecture'', many technical solutions have been proposed. We group these technical solutions into different categories and discuss them in the subsequent IV-B to IV-G sections. In addition, several approaches [2], [6], [18] that do not employ the encoder-decoder framework are grouped into the LM category discussed in Section IV-D along with other LM-related studies.

B. IMAGE FEATURE EXTRACTION

In this subsection, we discuss the image feature extraction solutions in the literature of RSIC from three aspects: selecting suitable CNN backbones, extracting features from CNN layers with different depths, and obtaining valuable information from RSIs via multitask approaches. The statistics of these three aspects in each RSIC study are shown in Table 1.

1) CNN BACKBONES

Extracting high-quality and valuable information from RSIs is a key part of the RSIC process. There are two categories of methods to extract information from RSIs, namely, handcrafted methods and deep learning-based methods. Handcrafted methods extract human-specified local features from images and transform them into image representations by encoding techniques such as bag of words (BOW) [34] and Fisher vector (FV) [35]. Deep learning-based methods utilize large-scale datasets to train CNN models. The trained models can automatically extract valuable information from unseen images. In recent years, many variants of CNN structures, such as AlexNet, VGG, and GoogLeNet, have been proposed for better extracting image features. These CNN structures are often used as the backbone of encoders in RSIC studies.

As shown in Table 1, Qu *et al.* [1] tested the performance of four different CNN backbones, AlexNet, VGG-16, VGG-19 and GoogLeNet, in extracting image features in the proposed encoder-decoder based RSIC framework. According to their experimental results, VGG and GoogLeNet outperformed AlexNet on the newly constructed UCM-captions and Sydney-captions datasets, with the combination of VGG-19 and LSTM achieving the best overall performance.

Later, Lu *et al.* [3] introduced the attention mechanism into the RSIC task and constructed a larger scaled dataset called the remote sensing image captioning dataset (RSICD). In addition to the CNN methods of AlexNet, VGG and GoogLeNet, handcrafted features, including BOW, FV, vector of locally aggregated descriptors (VLADs) [36] and scale-invariant feature transform (SIFT) [37], are tested for extracting RSI features. Experimental results show that the features extracted via all CNN methods significantly outperform the handcrafted features. In addition, almost all CNN methods obtain similar results in the encoder-decoder-based RSIC framework with the attention mechanism proposed in [3].

In subsequent RSIC studies, some works compared the performance of different CNN backbones in the experiments, including AlexNet, VGG, GoogLeNet, CaffeNet, residual neural network (ResNet), Inception and DenseNet. However, the research focus of all these works is to propose improved models rather than comparing different CNNs. In addition, in some studies, VGG and ResNet are selected as the backbone for fair comparisons with other works.

In summary, CNN-based deep learning methods are significantly better than handcrafted feature-based methods, and selecting the most appropriate CNN backbone can improve the overall performance of the RSIC models. On the other hand, since almost all RSIC studies focus on proposing better designed frameworks or models rather than comparing different CNN backbones, recent studies tend to choose commonly used backbones such as VGG and ResNet for a fair comparison with other works.

2) MULTILEVEL FEATURES

When an image is fed into a CNN extractor, layers of different depths extract features containing different information. Figure 5 illustrates the information extracted from the layers with different depths of a CNN encoder (taking VGG-16 as an example). The features extracted from the shallower layers contain more spatial and detailed information of the input image. The features extracted via the deeper layers contain more global and high-level semantic information. In the first encoder-decoder framework proposed by Qu *et al.* [1], only the information extracted from the deeper fully connected (FC) layers is utilized, while the information from the shallower convolutional layers is not used. In contrast, only the spatial information extracted from the shallower convolutional layers is utilized in the later proposed attention mechanism-based framework [3].

Zhang *et al.* [7] first combined the spatial information extracted from shallower convolutional layers and the high-level semantic information extracted from deeper FC or softmax layers. The combined multilevel features are used to train a novel attention model called the attribute attention mechanism (AAM). Compared with the conventional attention mechanism [3], [72], which only utilizes the spatial information extracted from shallower convolutional layers, the AAM can provide a better attention mask for the decoder to generate more appropriate captions.

TABLE 1. Image feature extraction methods of each RSIC study. "conv" and "FC" represent features obtained from the shallower convolutional layers and the deeper fully connected layers, respectively, and "conv+FC" represents both.

RSIC study	backbone	layers	multi-task
Qu et al. 2016 [1]	AlexNet [41], VGG-16/19 [33], GoogLeNet [42]	FC	-
Shi et al. 2017 [2]	VGG-f based FCN [32]	-	detect elements of three different levels
Lu et al. 2017 [3]	BOW, FV, VLAD, SIFT, AlexNet, VGG-16/19, GoogLeNet	conv	-
Zhang et al. 2017 [4]	CaffeNet [46]	FC	-
Wang et al. 2018 [5]	VGG-16	FC	output important region proposals
Wang et al. 2019 [6]	ResNet-50 [38]	FC	-
Zhang et al. 2019 [7]	VGG-16	conv+FC	-
Lu et al. 2019 [8]	VGG-16	FC	-
Zhang et al. 2019 [9]	VGG-16	conv	-
Zhang et al. 2019 [10]	VGG-16, Inception-V2 [43], ResNet-152	FC	-
Yuan et al. 2019 [11]	VGG-16	conv+FC	extract nouns & adjectives as attributes
Zhang et al.2019 [12]	VGG-16	conv	extract class labels
Kumar et al. 2019 [13]	VGG-16, Inception-V3 [44], ResNet-50/152	FC	extract class labels
Chavhan et al. 2020 [14]	AlexNet	FC	-
Shen et al. 2020 [15]	ResNet-101	conv	-
Shen et al. 2020 [16]	VGG-16	conv	fine-tune CNN extractor with an additional VAE branch
Li et al. 2020 [17]	AlexNet, VGG-16, ResNet-18, GoogLeNet	conv	-
Wang et al. 2020 [18]	ResNet-101	conv+FC	-
Huang et al. 2020 [19]	VGG-16, ResNet-18	conv	spatial/channel-wise denoising
Hoxha et al. 2020 [20]	Inception-V3	FC	-
Fu et al. 2020 [21]	ResNet-101	conv	-
Li et al. 2020 [22]	ResNet-101	conv	-
Wu et al. 2020 [23]	VGG-19	conv	-
Sumbul et al. 2020 [24]	VGG-16, GoogLeNet, InceptionV3, ResNet-152, DenseNet [45]	FC	-
Ma et al. 2020 [25]	ResNet-50, VGG-16 based SSD-512 [47] framework	conv	predict target-location mask
Wang et al. 2020 [26]	AlexNet, VGG-16, ResNet-18, GoogLeNet	FC	extract word label
Hoxha et al. 2020 [27]	ResNet-50	FC	-
Wang et al. 2020 [28]	ResNet-101, Faster RCNN [48]	FC	extract object/patch features
Yang et al. 2020 [29]	ResNet-101	FC	-
Zhao et al. 2021 [30]	ResNet-50	conv	output important region proposals

This combined utilization of multilayer features was adopted by later RSIC studies. In [18], Wang *et al.* used a ResNet101-based CNN encoder to extract multilevel features from RSIs. Specifically, the convolutional features with dimensions of 2048 × 49 before the pool5 layer are utilized as spatial information. The features output from the pool5 layer with dimensions of 2048 × 1 are used as high-level semantic features. Huang *et al.* [19] utilized multilevel features from the 3rd, 4th and 5th max pooling layers of the CNN encoder in the proposed denoising-based RSIC framework.

In summary, the RSI features extracted from the shallower convolutional layers contain more spatial and detailed information. The features extracted from deeper FC layers contain more global and high-level semantic information. Utilizing both types of features in combination can improve the overall RSIC performance.

3) OBTAIN INFORMATION VIA MULTITASK

In many RSIC studies, the task of CNN encoders is to receive RSIs as input and output the image representations extracted by the convolution of FC layers to other modules. This single-task-based image feature extraction approach has limitations in addressing the problem of diversity and scale variability of ground objects in RSIs. In some studies, researchers designed multitask-based methods to extract various types of information from RSIs or to better train the parameters of CNN encoders.

In [2], Shi *et al.* proposed an object detection and template-based RSIC method. In the object detection phase, three different tasks are designed to obtain different levels of information, i.e., key-instance, environment-element and landscape. The information obtained via multitask learning is input to the language template for generating textural descriptions for RSIs. Wang *et al.* [5] proposed a method named the

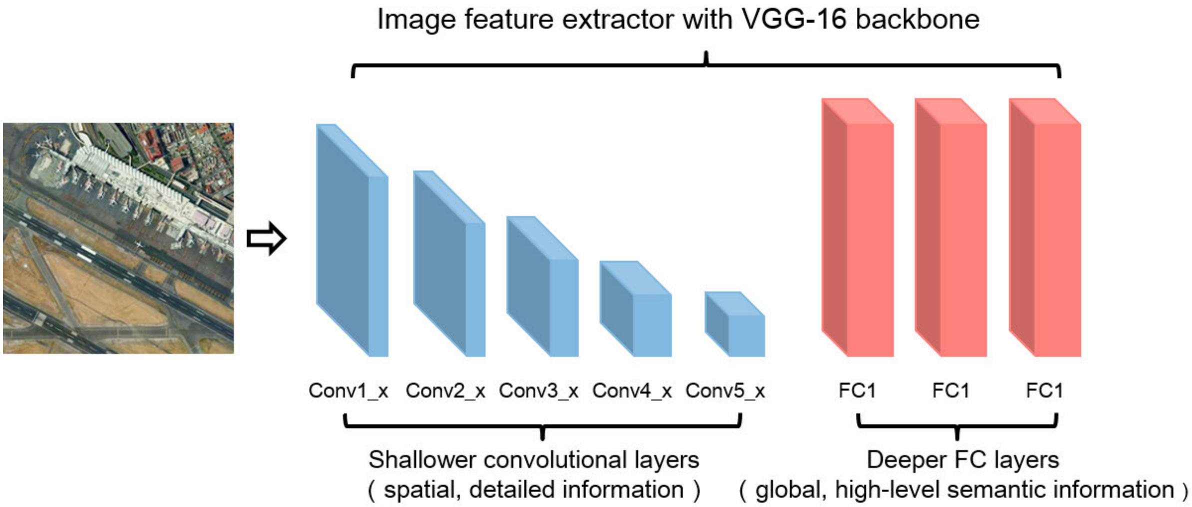


Image feature extractor with VGG-16 backbone
Conv1_x
Conv2_x
Conv3_x
Conv4_x
Conv5_x
FC1
FC1
FC1
Shallower convolutional layers
(spatial, detailed information)
Deeper FC layers
(global, high-level semantic information)

FIGURE 5. Different types of image features are extracted from layers of different depths of the CNN extractor (VGG-16 is used as an example).

intensive positioning network. An intensive positioning layer is designed to extract coordinates, confidence scores and features of valuable regions in RSIs. Zhao *et al.* [30] obtained region proposals of RSIs via additional tasks and utilized them to optimize the proposed RSIC model.

In some work, the classification information of the images is obtained and fused with CNN features to improve the overall performance of the model. Zhang *et al.* [12] designed an additional multiclassification task to detect objects in RSIs and generate corresponding labels in the form of word vectors. The word vectors are fused with the feature vectors extracted by the CNN encoder to train a better attention model named the label attention mechanism. Kumar *et al.* [13] collected RSIs using UAVs and annotated a new RSIC dataset. A region-driven method is proposed to obtain the classification information via an additional image classification task. The performance of the model is improved by combining the region-driven information with the features extracted by the CNN encoder. Wang *et al.* [26] designed a multi-label classification module for extracting word information from RSIs to optimize RSIC models.

Ma *et al.* [25] extracted features of different levels in an RSI through two tasks. In the conventional feature extraction task, RSIs are fed into a ResNet-50 network whose FC layer is removed. The output feature vectors, which are spatially adjacent, are utilized as scene-level features. Another object detection task is used to extract the target-level features. A VGG-16-based SSD-512 framework [47] is trained on the DIOR dataset [56] for this auxiliary task. The obtained target-level features are in the form of lists of vectors. Then, both the scene and target-level features are fed into an LSTM LM to further generate image captions.

Wang *et al.* [28] extracted three different types of image features, namely, object, patch and global features, via a multitask method. The object features are obtained by an object detection model based on a Faster region-based convolutional neural network (RCNN) [48], whose output is a list of regions of interest (ROIs). The patch features are the features of the expanded region around each ROI. The global features are obtained from the FC layer of the CNN encoder. These three different features are then fused and fed into the attention model to generate the attention mask.

Shen *et al.* [16] designed an additional image reconstruction task to train a better CNN encoder. The parameters of the CNN encoder are simultaneously fine-tuned on RSIC datasets through the image captioning task and on a large-scale RSI classification dataset NWPU-RESISC45 [60] via the image reconstruction task. The network before the last convolutional layer of the CNN encoder is connected to a variational autoencoder (VAE) branch for the image reconstruction task. Since the scale of the NWPU-RESISC45 dataset is much larger than that of the existing RSIC datasets, the CNN encoder trained by the proposed multitask can solve the overfitting problems caused by small datasets in the RSIC task.

Huang *et al.* [19] optimized the RSI features extracted by the CNN encoder by denoising tasks. The feature vectors extracted by the convolutional layers of the CNN encoder have $H \times W \times C$ dimensions, where dimension C represents different channels and dimension $H \times W$ represents different spatial locations of the RSI. Two different denoising tasks, namely, spatial-wise and channel-wise, are designed. Spatial denoising aims to denoise features at different spatial locations in each channel, while channel-wise denoising addresses features in different channels at each location. Technically, the convolutional feature vectors are fed into multilayer perceptrons consisting of FC layers and activation functions for denoising.

In summary, obtaining and utilizing additional information such as words, labels, and attributes from multiple tasks can

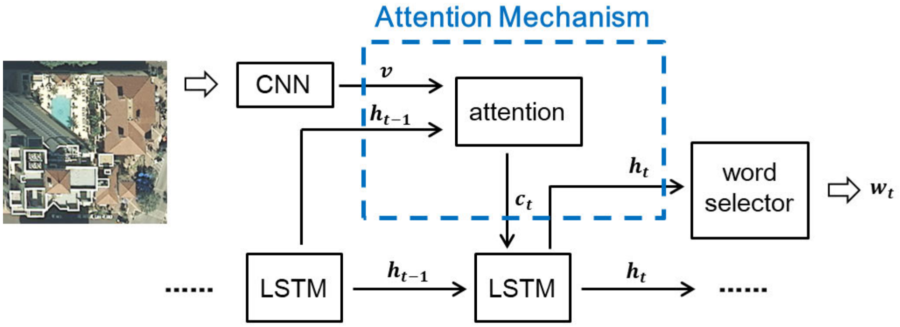


FIGURE 6. Diagram of the attention mechanism.

improve the overall performance of RSIC models. In addition, better-quality RSI features can be obtained through additional tasks, such as VAE training and denoising.

C. ATTENTION MECHANISMS

Lu *et al.* [3] first introduced the attention mechanism [72] into the task of RSIC. As shown in Figure 6, the attention mechanism acts as a middleware between the CNN encoder and the LSTM decoder. At the t-th time step, the attention module calculates a weight distribution α for each spatial image region according to the features v obtained from the CNN encoder and the previous output h_{t-1} of the LSTM decoder:

$$\alpha = \sigma(g(v, h_{t-1})) \tag{13}$$

where g represents a multilayer perceptron network containing trainable parameters, and σ denotes a softmax layer. The context vector c_t is calculated by summing the product of each spatial feature v_i and the corresponding attention weight α_i:

$$c_t = \sum_{i=1}^{L} \alpha_i v_i \tag{14}$$

The LSTM decoder can then generate the next hidden state h_t according to c and the previous hidden state h_{t-1}:

$$h_t = LSTM(c, h_{t-1}) \tag{15}$$

The word selector module can then generate the word w_t for time step t according to h_t. Intuitively, the attention mechanism is consistent with human habits of describing pictures. As we pronounce each word, we focus our attention on a specific region on the image. The experimental results demonstrate a significant performance improvement in the encoder-decoder-based RSIC framework after the introduction of the attention mechanism.

A more efficient attention mechanism for RSIC named attribute attention mechanism (AAM) was proposed by Zhang *et al.* [7]. As shown in Figure 7, compared to [3], who only used the spatial features (i.e., v in Eq.(13)) extracted from shallower convolutional layers, the AAM additionally utilizes high-level features v_{attr} obtained from the FC layer or softmax layer of the CNN encoder to calculate the attention weight distribution α:

$$\alpha = \sigma(g([v_{conv}; v_{attr}], h_{t-1})) \tag{16}$$

where [;] denotes the concatenation operator and v_{conv} and v_{attr} represent the spatial features obtained from the shallower convolutional layers and the attribute features obtained from the deeper FC or softmax layer. The theory behind the AAM is that features extracted from the shallower convolutional layers contain mainly detailed features of the image, while features obtained from deeper layers contain more global semantic features. The combined use of these two types of features can effectively handle the “various scales of objects” problem in RSIs and obtain more accurate attention weights.

In addition to image features, the label attention mechanism (LAM) method [12] utilizes the classification labels of the RSI to calculate attention weights:

$$\alpha = \sigma(g(w_{lab}, v_{conv}, h_{t-1})) \tag{17}$$

where w_{lab} is an embedded word label (e.g., “plane”) for the input RSI, and v_{conv} represents the image features obtained from the convolutional layers of the CNN encoder. A classifier is additionally trained to obtain the word label w_{lab}. Experimental results show that the LAM method outperforms the AAM [7] and the vanilla attention mechanism [3] in the task of RSIC. The rationale behind the experimental results is that accurate label information can help calculate better attention weights. The structure of the LAM is illustrated in Figure 8.

Wu *et al.* [23] proposed a method called the scene attention mechanism (SAM) with a residual structure. As shown in Figure 9, unlike the conventional attention mechanism [3], SAM utilizes the hidden state of the current rather than the previous time step for calculating the attention maps. In addition, the output of SAM is not directly used to train the LSTM decoder, which is different from other attention-based methods. At each time step, the attention weights are calculated as follows:

$$\alpha = \sigma(g(v, [v; h_t])) \tag{18}$$

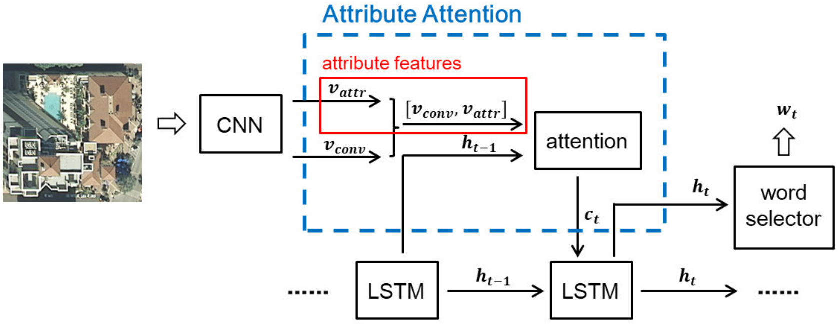


FIGURE 7. Diagram of the attribute attention mechanism.

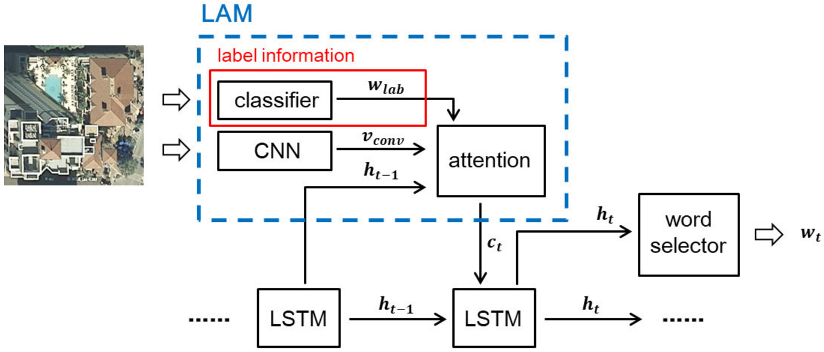


FIGURE 8. Diagram of the label attention mechanism.

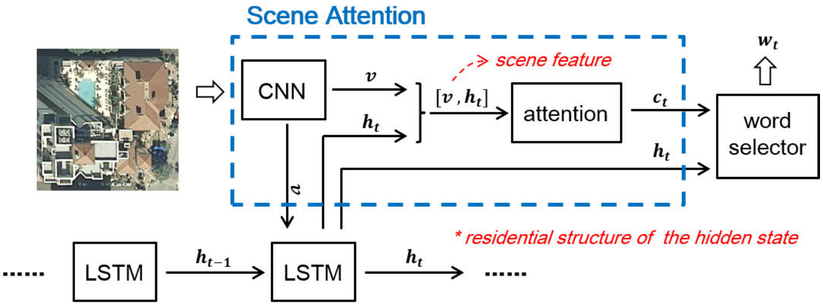


FIGURE 9. Diagram of the scene attention mechanism.

where v is the mean of the image features extracted by the CNN encoder, and the concatenation of v and h_t, i.e., $[v; h_t]$ is utilized as the "scene features". The residual structure of the hidden states helps enhance the stability of SAM to obtain better attention weights.

Li *et al.* [22] proposed a multilevel attention mechanism, which has a more complex structure (as shown in Figure 10) and a better performance. The final context vector z_t fed into the LSTM decoder at each time step is obtained by weighting the visual and text context vectors v_t and s_t:

$$z_t = \alpha_v v_t + \alpha_s s_t \tag{19}$$

where α_v and α_s denote the weights for the visual and text context vectors, respectively, and $\alpha_v + \alpha_s = 1$. When generating the next word, a larger α_v makes the model attend more to the spatial image features, while a larger α_s makes the model focus more on previously generated words. For example, when generating the word "plane", more attention should be given to the corresponding region in the image, while when generating the word "by", more attention is given to the previous generated word "near". The visual context vector v_t is obtained through the conventional attention mechanism [3], [72] ("attention1" in Figure 10). The text context vector s is calculated by the previously generated words:

$$\alpha_i = \frac{exp(g_1(\sum_{m=0}^{i} g_2(c_m, h_m)))}{\sum_{k=1}^{i} exp(g_1(\sum_{m=0}^{k} g_2(c_m, h_m)))} \tag{20}$$

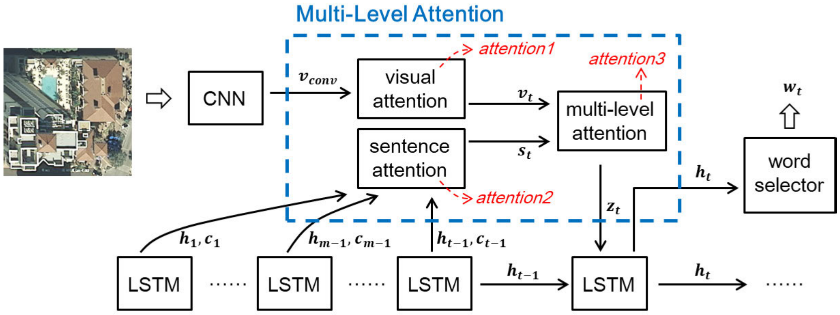


FIGURE 10. Diagram of the multilevel attention mechanism.

$$s = \sum_{i=1}^{t} \alpha_i c_i \tag{21}$$

where α_i denotes the attention weight ("attention2" in Figure 10) for the memory cell c_i at time step i, g_1 and g_2 are multilayer perceptron networks, and c_m and h_m are the content of cell memory and hidden state of the LSTM decoder at time step m. The proposed multilevel attention structure not only utilizes both image and text features (i.e., v_t and s_t) but also takes into account whether generating the next word requires more focus on the spatial image or the previous words.

In summary, there are two main ways to optimize the attention mechanism. One approach is to input richer valuable information into the attention model. The earliest attention model [3] (Equation 13) only accepts image features and the previous hidden state as the inputs. Later approaches additionally feed image features of different layers [7] (Equation 16), label information [12] (Equation 17), scene information [23] (Equation 18), and combined image and text information [22] (Equations 19-21) into attention models. The second approach is to design attention models with more ingenious structures, such as the residual structure in Figure 9 and the multi-layer and multi-connected structure in Figure 10. Both of them aim to make the attention mechanism generate a more accurate region map for each word.

D. LANGUAGE MODEL

In the task of RSIC, the function of the LM is to transform the information obtained through other modules into readable sentences. In the first RSIC study [1], RNN-like LMs, i.e., LMs based on RNN or its variant LSTM, were employed. In each time step, an RNN-like LM can predict the next word in the sentence based on the image features and the previously generated words. In early RSIC studies [1], [3], the performance of RNN and LSTM as the LM for the task of RSIC were compared in experiments. Experimental results show that LSTM is far superior to the RNN. In most of the subsequent RSIC studies, LSTM was widely used as the backbone of LMs.

Shen *et al.* [15], [16] designed RSIC frameworks with better performance using a transformer instead of LSTM as the LM. Unlike the recursive structure of LSTM, the structure of the transformer is a stack of many identical layers, as shown in Figure 11. Each layer consists of three sublayers. The first sublayer contains masked multihead attention, which keeps the model from seeing the future information and generates the current word based only on the previous information. The second sublayer is the core of the model, which provides multihead attention without the masked mechanism, correlating RSI features with textual information. The third sublayer is a feed-forward FC network. The residual connections around each sublayer and the stack structure allow the transformer decoder to capture various types of relationships in the sequence data. In addition, thanks to the absence of a time-dependent recursive structure, each sublayer in a transformer model can be trained in parallel, which greatly reduces the training time compared to LSTM.

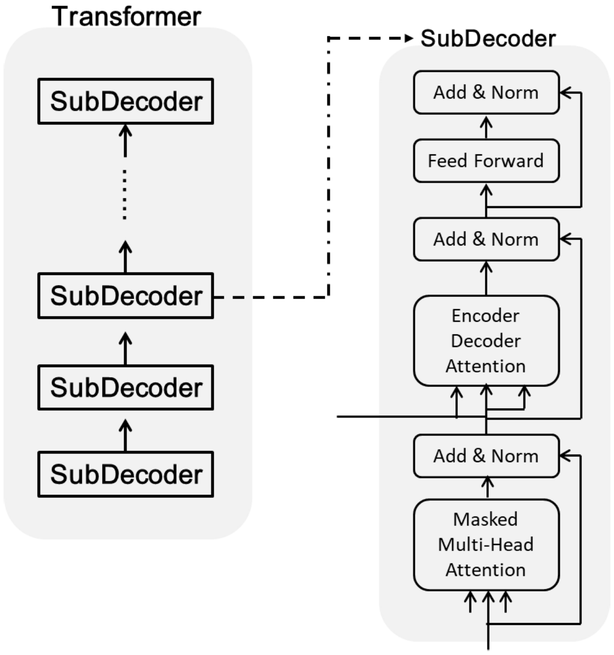


FIGURE 11. Diagram of the transformer-based language model.

In an early study [2] when the RSICD dataset [3] was not published, Shi *et al.* trained a template-based LM on an unpublished large-scale satellite dataset. Templates are prestructured sentences containing blanks to be filled. The blanks in the templates are generally scenes, ground objects, attributes and relationships among objects in RSIs. In [2], semantic information in RSIs is obtained via fully convolutional network (FCN)-based [32] object detection tasks and then transformed into words to fill in the blanks in the templates. The advantage of a template-based LM is that the structure of the sentence can be well defined and modified for specific tasks. The limitation of template-based LMs is that it cannot generate flexible and multivariate descriptions for RISs and cannot obtain high ratings by automatic evaluation metrics of RSIC. Therefore, after the three benchmark datasets [1], [3] were widely known, there was no more research focusing on template-based LMs.

In [6], a retrieval-based LM was designed. The CNN representation of each RSI and the corresponding multiple ground-truth captions are mapped into a common semantic space, in which the distance of the image and text representation can be calculated. A Mahalanobis matrix is trained to retrieve the sentences with the closest distance with an RSI in the space. Compared with LSTM- and transformer-based LMs, the captions obtained by the retrieval-based LM come directly from the ground-truth annotations in the training set, so there are no grammatical and syntactic errors. However, sentences generated by this method lack flexibility and variety and even contain descriptions that do not match the input image. Therefore, the model proposed in [6] achieved significantly lower scores on the benchmark datasets than the methods adopting LSTM- or transformer-based LMs.

Wang *et al.* [18] designed a recurrent memory network-based RSIC framework along with a novel CNN-based LM. Specifically, three different types of memory cells, namely, image, topic and temporary cells, are employed to store the information of the image, topic and previously generated words. In each step, a 1-D convolution is operated along with the dimension of the concatenation of the three types of memory cells. The convolutional result is fed into a softmax layer to generate the next word, and the corresponding temporary memory cell is updated using the newly generated word. This convolution operation is repeated every step until the end symbol is generated. Compared with the RNN, the convolution operation in each step goes along the temporary cells storing all the previously generated words, which solves the information dilution problem.

In summary, although template- and retrieval-based LMs have some advantages in terms of task-specific and grammatical correctness, these two types of methods cannot generate flexible and variable sentences and cannot achieve high performance on benchmark datasets. Therefore, these two types of LMs have attracted little attention in recent studies. LSTM-based LMs are utilized in most studies. This kind of LM can dig deeper into the dependencies between text sequences and generate flexible and versatile or even new sentence structures corresponding to the input RSIs. Transformer-based LMs that adopt a stack structure without recursive connections outperform the conventional LSTM model in terms of both performance and training time consumption. In future RSIC research, more optimized transformer-based LMs are expected to be proposed to outperform RNN-like models across the board. In addition, well-designed CNN-based LMs are expected to be proposed and utilized in specific scenarios and tasks of RSIC.

E. ACTIVE ATTENTION

The concept of ''active attention'', which was first proposed in the work of the sound active attention framework [8], refers to the generation of words guided by extra information other than images during the process of RSIC. The opposite of ''active attention'' is ''passive attention'', which refers to the generation of text descriptions that focus only on the image itself. In [8], sound information is utilized as an additional input to guide the generation of RSI captions. The image features are extracted by VGG16, and the extra sound information is encoded by the mel-frequency cepstral coefficients algorithm. The processed image and sound representations are sent into the model to generate RSI descriptions word by word. In this way, the caption generating process is guided by sound. That is, inputting different sounds with the same image into the trained model will obtain different descriptions. For example, when the sound ''trees'' are inputted, a description focusing on trees will be generated. While inputting the sound ''factory'' will obtain captions concentrating on factories. Since different observers usually describe the same RSI with different attention and people tend to speak out their concerns in voice in practical applications, this interesting sound-based research is expected to inspire many valuable real-time applications.

Another active attention-related RSIC method is proposed in the retrieval topic recurrent memory network (RTRMN) framework [18]. Topic information is used as an additional input to guide the generation of captions. During training, two kinds of topic information, namely, semantic and statistical topics, are extracted from the corresponding ground-truth annotations of the RSIs. The semantic topics are nouns, adjectives and verbs, while the statistical topics are words with high term frequency inverse document frequency (TF-IDF) [79] scores. In the test time, the topic information sent into the model can be either obtained from the annotated sentences of the most similar images in the training set via a retrieval-based method or manually controlled to guide the generation of the descriptions.

In summary, the training and inference of active attention-based models require additional data, such as sound or topic information. Along this direction, more interesting and valuable RSIC studies for practical applications are expected to be proposed.

F. AUXILIARY COMPONENT

Yuan *et al.* [11] designed an attribute relation encoding module that can be integrated into any encoder-decoder RSIC framework. The module receives an RSI as input and generates the attribute words (i.e., nouns and adjectives) via a ResNet-18-based multilabel classification network. The generated attributes are then fed into a GCN [73]-based relation learning module. Attribute words and their relationships are represented as the nodes and edges in the graph. An adjacency matrix with conditional probability is trained to learn the co-occurrence between different attribute words. The relationship information is then fed into the LM to improve the quality of the generated captions.

Fu *et al.* [21] proposed an external storage structure called the persistent memory mechanism (PMM), which can be integrated into any RNN-like RSIC decoder. The core part of PMM is a storage memory implemented in the form of matrices. In each time step, the proposed PMM module performs search and update operations on its storage memory according to the previous output of the RNN-like decoder and the input RSI. The search result is input into the LM to generate the next word. The update operation makes the storage memory provide valid information for later generation. The experimental results show that the proposed PMM module can be integrated into any RNN-based LM and improve the overall performance.

Sumbul *et al.* [24] proposed a framework called summarization-driven RSIC. This framework contains a conventional encoder-decoder module, a summarization module and a fusion module. The latter two modules are auxiliary components that can be embedded into any encoder-decoder-based RSIC framework. The encoder generates multiple single sentences for the input images based on the beam search algorithm. The summarization module is a pointer generation network [81] pretrained on the Gigaword dataset [55]. This module can merge multiple sentences into a single summarization sentence. The merged summarization sentence retrains the semantics of multiple single sentences and removes redundant information. The fusion module is implemented based on an LSTM network, which calculates the probability distribution for all words in the vocabulary based on the input image, the multiple single captions and the summarization sentence in each time step. The word with the highest probability is selected as the next word in the final output sentence. Experimental results show that the proposed external modules can not only improve the performance of traditional encoder-decoder frameworks but can also generate valuable new words for describing RSIs, i.e., words that are contained in the Gigawords corpus but not in the RSIC training sets.

In summary, auxiliary components are proposed to provide a flexible refinement for the traditional encoder-decoder and attention mechanism-based RSIC frameworks. The contributions of this type of research can be easily borrowed and utilized by other studies.

G. TRAINING STRATEGY

The training strategy refers to the strategy used for adjusting the parameters of RSIC models according to the difference between the generated captions and the ground-truth annotated sentences during training. Cross entropy (CE) is employed as the training strategy in most RSIC studies, where the generation of each word is treated as a multiclassification task. The probability distribution of each word in the vocabulary is calculated in every time step, and the object of the training is to minimize the loss value $\mathcal{L}$:

$$\mathcal{L} = -\sum_{i=1}^{N} y^{(i)} * log\hat{y}^{(i)} \tag{22}$$

where $y^{(i)}$ and $\hat{y}^{(i)}$ represent the generated and ground-truth probability of the i-th word in the vocabulary, and N is the total number of words in the vocabulary. Intuitively, CE forces the model to generate the target word with a probability of 1 at each time step.

Li *et al.* [17] pointed out that the importance of the target word is overemphasized by CE, resulting in some valuable words, such as synonyms, being excluded as non-target words. Such an optimization method would lead to overfitting, especially in small datasets. In [17], a new loss function called truncation cross entropy (TCE) was proposed for RSIC. A truncation threshold is designed to reserve a margin for the non-target words to prevent overoptimization of the target words. When the probability of the output word exceeds the threshold, CE optimization is not used. In each time step, the loss value is calculated by the following equation:

$$\mathcal{L}_{TCE} = \begin{cases} -y_t^{(s_t)} * log\hat{y}_t^{(s_t)}, & \text{if } y_t^{(s_t)} < 1-\gamma \\ -log(1-\gamma), & \text{otherwise} \end{cases} \tag{23}$$

where γ denotes the value of the truncation threshold and $y_t^{(s_t)}$ and $\hat{y}_t^{(s_t)}$ represent the generated and ground-truth probability of the target word s_t at time step t, respectively. Experiments on benchmark RSIC datasets demonstrate the efficiency of the proposed TCE optimization scheme.

In the CE-based training strategy, loss values are calculated based on the difference between the generated and ground-truth sentences. For models with attention mechanisms, CE-based optimization cannot help align the attention regions and the corresponding texts. To solve this problem, Zhang *et al.* [9] proposed a new loss function called visual aligning loss for directly optimizing the attention mechanisms during each training epoch. Visual aligning loss is calculated based on the similarity between the attended image region and the corresponding visual word. Concretely, a vocabulary containing visual words is constructed by excluding nonvisual words in the RSIC datasets. Image features extracted by CNNs and embeddings of visual words in the vocabulary are mapped into a common vector space via two separate multilayer perceptrons. In the common space, the similarity of an image feature vector v and the embedding

x of a visual word can be calculated via cosine similarity:

$$sim_{eqnarray}(v, x) = cos(v, x) \tag{24}$$

When a sentence is generated, the visual aligning loss $\mathcal{L}_{eqnarray}$ for optimizing the attention mechanisms can be calculated by the similarity of each visual word x_i in the sentence and the corresponding feature v_i of the attended region:

$$\mathcal{L}_{eqnarray} = \begin{cases} 1 - \frac{1}{N}\sum_{i=1}^{N} sim_{eqnarray}(v_i, x_i), & \text{if } N > 0 \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

where N is the number of all visual words that appeared in the generated sentence. The overall loss $\mathcal{L}$ in each training epoch is obtained by weighting the newly proposed visual aligning loss $\mathcal{L}_{eqnarray}$ and the CE loss $\mathcal{L}_{CE}$ between the generated and ground-truth captions:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda\mathcal{L}_{eqnarray} \tag{26}$$

where λ is the trade-off parameter. Experimental results on Sydney-captions and UCM-captions datasets shows the effectiveness of the proposed visual aligning loss.

Although CE is effective and easy to implement, this teacher-forcing training strategy has an exposure bias issue. That is, the optimization goals during training are not consistent with the evaluation metrics in test time. To address this issue, Shen *et al.* [15] and Yang *et al.* [29] utilized the reinforcement algorithm-based self-critical sequence training [76] method to train RSIC models. In reinforcement learning terminology, the RSIC model is viewed as an "agent", and the RSI features and words are viewed as the "environment". During the interaction between the "agent" and "environment", the "policy" p_θ defined by the parameters θ of the generator makes an "action" that predicts the next word. When a complete sentence is generated, the agent receives a reward r, which is a score of an evaluation metric such as the consensus-based image description evaluation (CIDEr) calculated using the generated sentence and the ground-truth sentence. The goal of training is to minimize the negative expected reward:

$$L(\theta) = -\mathbb{E}_{w^s \sim p_\theta}[r(w^s)] \tag{27}$$

where $w^s = (w_1^s, \ldots, w_T^s)$ and w_t^s is the word sampled from the generator at time step t. A reinforcement algorithm with a baseline [76] is utilized to compute the gradient of loss for r. The reward obtained by the current model under the inference algorithm used at test time is utilized as the baseline for "self-critical". This approach optimizes the model directly with the test-time evaluation metrics, thus solving the exposure bias problem of CE.

Chavhan *et al.* [14] proposed an improved training strategy based on actor-critic reinforcement learning [75] for RSIC. In addition to the LSTM-based critic in [75], a second encoder-decoder critic is added. The newly added critic translated the generated captions back into an RSI and reward the actor by calculating the similarity of the back-translated RSI with the input RSI. Experimental results show that the proposed actor- and dual-critics-based reinforcement learning approach achieved better performance compared with the state-of-the-art RSIC methods and the actor-critic-based reinforcement learning approach in [75].

In general, most RSIC studies utilize a CE-based training strategy, which is effective and easy to implement. However, CE-based methods are prone to overfitting and exposure bias problems. There are two routes in the literature to solve the above problems. One is to propose a modified loss function based on CE, and the other is to adopt reinforcement learning-based approaches.

V. DATASETS AND EVALUATION METRICS

A. DATASETS FOR RISC

Datasets play a critical role in RSIC studies. To spawn better models and ideas, a good RSIC dataset should contain a large amount and a reasonable distribution of RSIs covering a rich variety of remote sensing scenes and ground objects, as well as well-annotated sentences corresponding to each image. In this section, we discuss in detail the characteristics and pros and cons of the existing RSIC datasets in the literature.

1) UCM-CAPTIONS

Qu *et al.* [1] annotated the UCM-captions dataset on the basis of the UCM Land Use Dataset [50]. The images in the dataset were manually extracted from the large images in the USGS National Map Urban Areas Imagery collection for various urban areas around the country. There are 21 different types of scenes included in the dataset, specifically containing: agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium residential, mobile home park, overpass, parkinglot, river, runway, sparse residential, storage tanks, and tennis court. The whole dataset contains a total of 2100 images and 100 images for each scene. Some representative images are shown in Figure 12.

Each image is annotated with five different sentences. The final dataset contained 10,500 annotated sentences with a vocabulary of 368 words. Compared to Sydney-captions, another RSIC dataset created in [1], the annotated sentences in UCM-captions are relatively simple and somewhat monotonous in sentence patterns. Figure 13 shows an example image with the five annotated sentences from this dataset.

2) SYDNEY-CAPTIONS

The Sydney-captions dataset, which was annotated on the basis of the Sydney dataset [49], is another RSIC dataset published in [1] along with UCM-captions. The dataset contains a total of 613 land-use images cropped from a very large satellite image of Sydney, Australia, acquired from Google Earth. Seven different scenes are included, i.e., residential, airport, meadow, rivers, ocean, industrial, runway. Figure 14 shows some representative images in the Sydney-captions dataset.

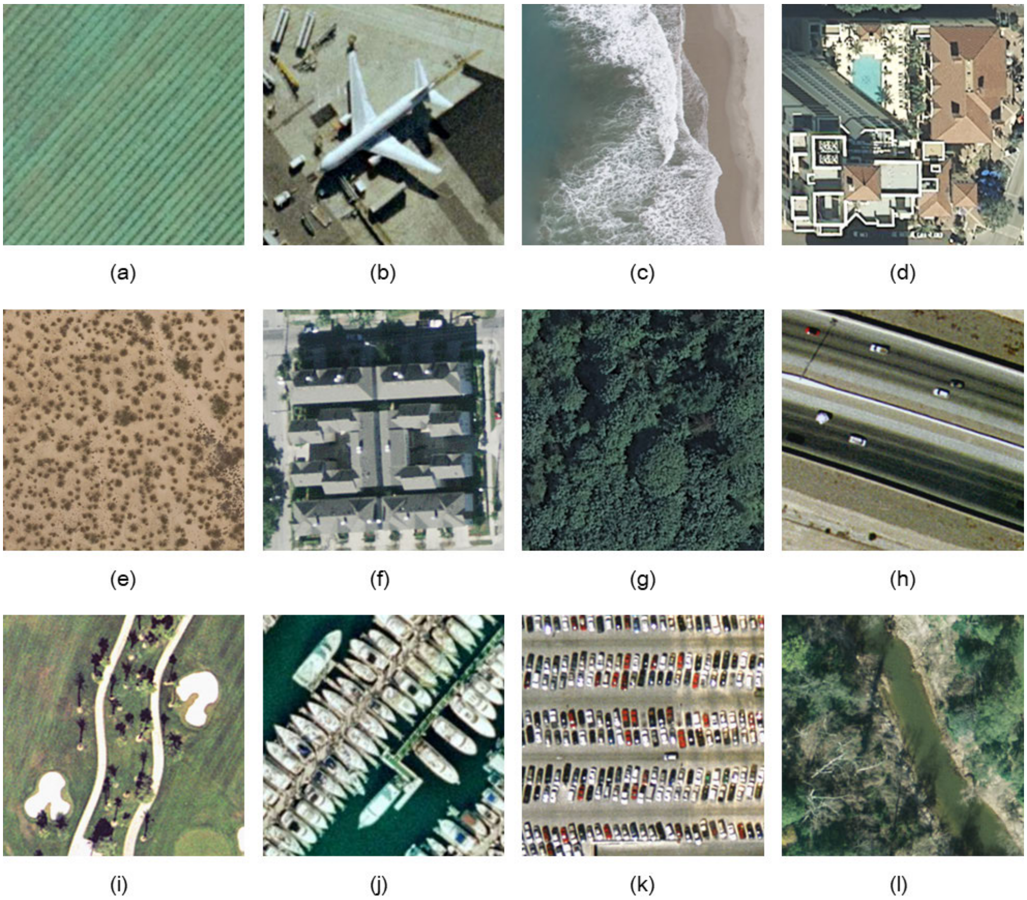
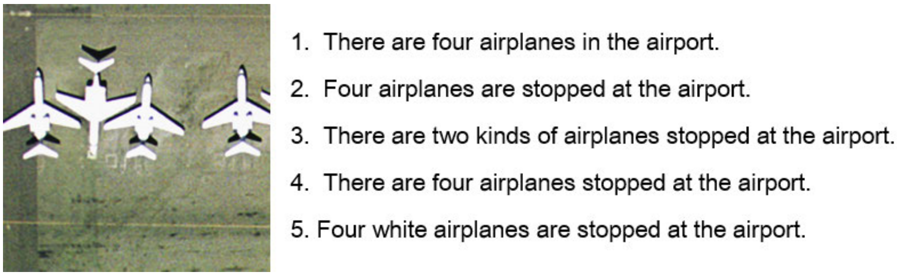


FIGURE 12. Sample images in the UCM-captions dataset: (a) farmland, (b) airport, (c) beach, (d) buildings, (e) chaparral, (f) dense residential, (g) forest, (h) freeway, (i) golf course (j) harbor, (k) parking lot, and (l) river.



1. There are four airplanes in the airport.
2. Four airplanes are stopped at the airport.
3. There are two kinds of airplanes stopped at the airport.
4. There are four airplanes stopped at the airport.
5. Four white airplanes are stopped at the airport.

FIGURE 13. An example image with annotated sentences in the UCM-captions dataset.

Each image was annotated with five different sentences, and the number of words in the vocabulary was 237. Figure 15 shows an example image along with the annotated five sentences. As shown in the figure, the description sentences in this dataset have a larger average length and a more appropriate and richer vocabulary compared with the UCM-captions dataset.

However, there are two obvious drawbacks of this dataset. First, the distribution of images in each category of a scene is severely unbalanced. As shown in Table 2, the scene with the highest number of images, residential, contains 242 images, while the lowest, airport, contains only 22 images. Second, the size of the dataset is small, containing only 613 images. Training RSIC models on such a small and unbalanced dataset is likely to lead to overfitting problems.

3) RSICD

The RSICD dataset was published in 2017 by Lu *et al.* [3] and is the largest publicly available RSIC dataset to date. A total of 10,921 RSIs covering 30 scenes collected from GoogleEarth, BaiduMaps, MapABC, and Tianditu were included in this dataset. Images belonging to different scenes have good differentiation in terms of content and features. Some representative images in RSICD are shown in Figure 16. The numbers of images in different scenes are relatively evenly distributed, and each

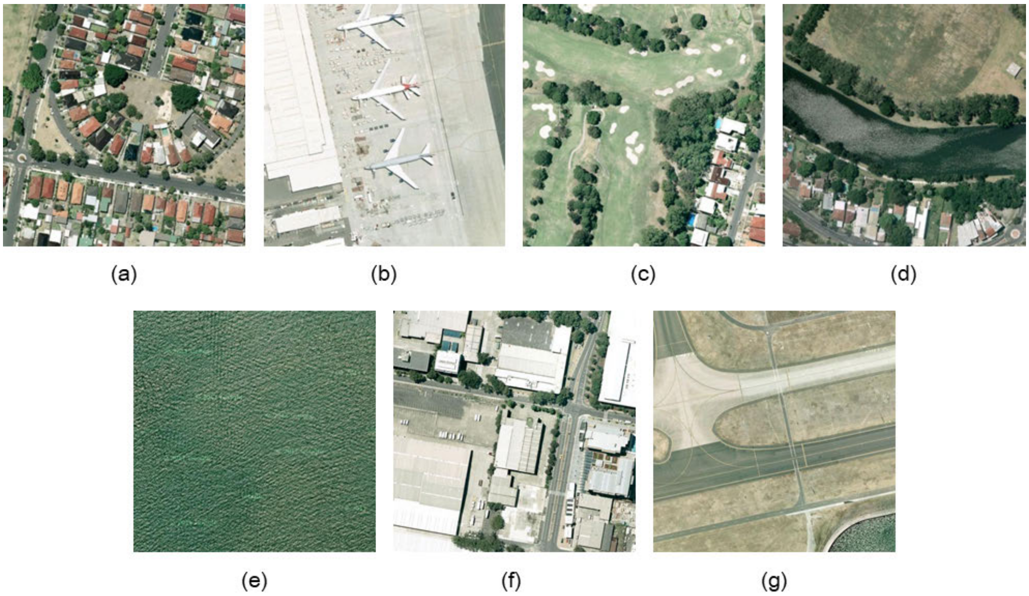


FIGURE 14. Sample images in the Sydney-Captions dataset: (a) residential, (b) airport, (c) meadow, (d) river, (e) ocean, (f) industrial, and (g) runway.

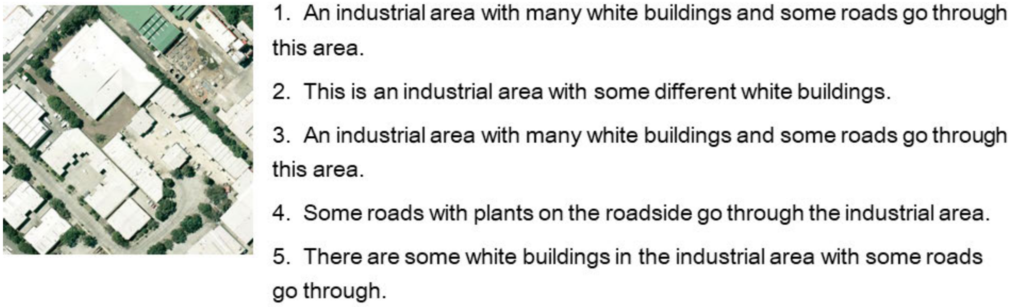

1. An industrial area with many white buildings and some roads go through this area.
2. This is an industrial area with some different white buildings.
3. An industrial area with many white buildings and some roads go through this area.
4. Some roads with plants on the roadside go through the industrial area.
5. There are some white buildings in the industrial area with some roads go through.

FIGURE 15. An example image with annotated sentences in the Sydney-Captions dataset.

TABLE 2. Number of images of each type of scene in the Sydney-Captions dataset.

Scene	Number
Residential	242
Airport	22
Meadow	50
River	45
Ocean	92
Industrial	96
Runway	66

category of scene contains more than 200 images. The specific number of images for each type of scene is shown in Table 3.

During annotation, in addition to the instructions referring to the work of constructing NIC datasets [51]–[53], a set of RSI-specific instructions was proposed to guarantee the quality of the annotated sentences. These instructions include the following: not to use directional words (unlike photos of natural images that have a fixed direction of observation, RSIs with a "God's view" does not have a fixed direction), not to use vague concepts such as large, tall (the size of objects in RSI is variable), and so on. The annotated dataset contains 10,921 images and 24,333 different sentences with a vocabulary of 3323 words.

A drawback of the RSICD is that not every image was annotated with the same number of sentences. Only 724 images were annotated with 5 different sentences, while the remainder of the images were annotated with 1-4 sentences. Lu *et al.* expanded the number of annotated sentences to 54605 (5 for each image) by randomly copying the existing sentences when there were not five. Figure 17 shows an image that was annotated with five different sentences, and Figure 18 shows an image that was annotated with only one sentence The number of annotated sentences was expanded to five by duplicating the existing sentence. Table 4 shows the number of images annotated with different numbers of sentences. In addition, there are some errors in the annotated sentences. For example, the word "separated" is misspelled as "seperated" in the first sentence in Figure 17. Although

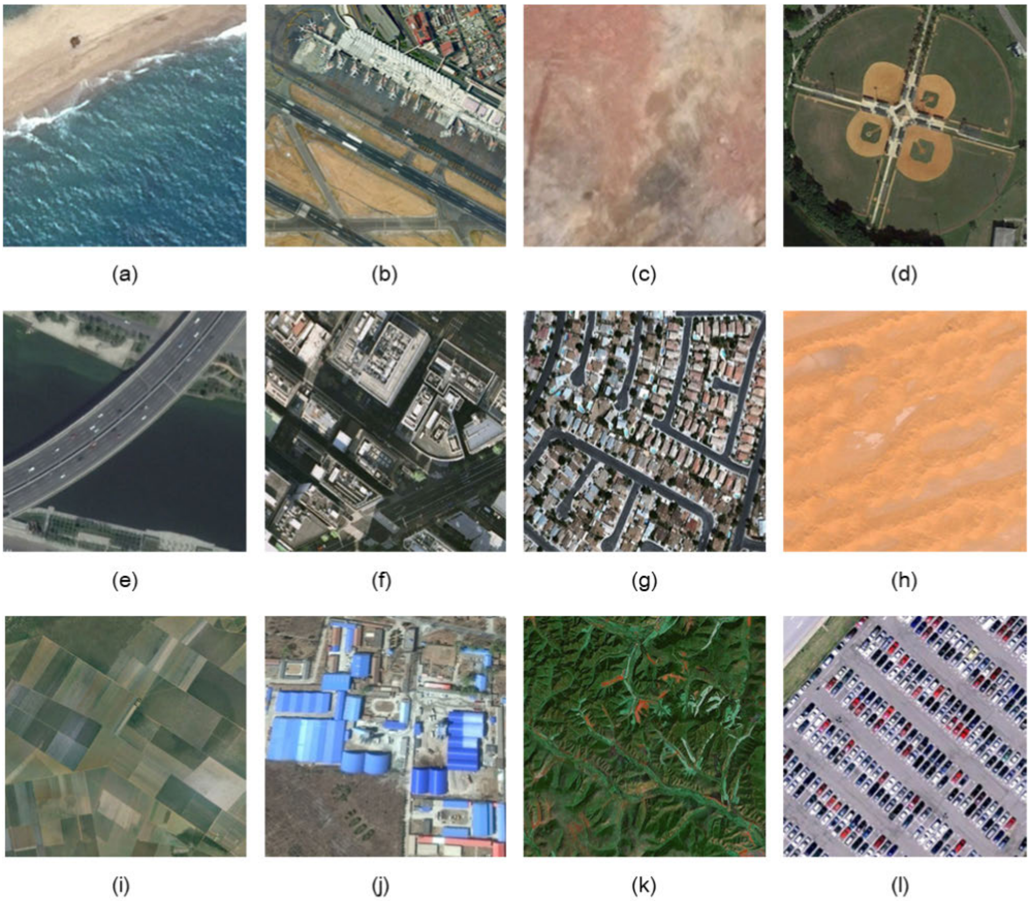


FIGURE 16. Sample images in the RSICD dataset: (a) beach, (b) airport, (c) bareland, (d) baseball field, (e) bridge, (f) commercial area, (g) dense residential, (h) desert, (i) farmland, (j) industrial, (k) mountain, and (l) parking.

TABLE 3. Number of images of each type of scene in RSICD dataset.

Scene	Number	Scene	Number	Scene	Number
Airport	420	Farmland	370	Playground	1031
Bare Land	310	Forest	250	Pond	420
Baseball Field	276	Industrial	390	Viaduct	420
Beach	400	Meadow	280	Port	389
Bridge	459	Medium Residential	290	Railway Station	260
Center	260	Mountain	340	Resort	290
Church	240	Park	350	River	410
Commercial	350	School	300	Sparse Residential	300
Dense Residential	410	Square	330	Storage Tanks	396
Desert	300	Parking	390	Stadium	290

TABLE 4. Number of images with different numbers of sentences annotated.

Number of Images	Sentences per Image
724	5
1495	4
2182	3
1667	2
4853	1

humans can easily correct such spelling errors when reading, there are two completely different words to the computer.

4) OTHERS

In [2], Shi *et al.* trained a target detection and template-based RSIC model using images acquired from Google Earth and GaoFen-2 satellites without textual annotations. The dataset contains 330 RSIs with a very large size of 1000-8000 pixels and 200 images of 480 × 640 pixels. Although the well-designed multilevel object detection task and the template-based language generator in [2] are able to generate detailed and appropriate text descriptions, this dataset cannot be used to incubate RSIC methods other than template-based models due to the absence of textual annotations as ground truth.

An RSIC dataset consisting of images captured by UAVs was annotated by Kumar *et al.* [13]. The images in this dataset involved 12 scenes, i.e., barren lands, farmlands, forests, gardens, highways, playgrounds, residential areas, roads, runways, solar panels, water bodies, and temples.

The annotated dataset contains 1285 UAV images of 400 × 400 pixels, each annotated with 5 ground-truth sentences, with a vocabulary of 940 words. In addition, Zhang *et al.* [4] annotated 2000 sentences for the RSIs in UCM [50] and trained an encoder-decoder-based RSIC model. However, these two datasets are not publicly available.

B. EVALUATION METRICS FOR RSIC

In RSIC studies, many evaluation metrics are utilized to measure the quality of the descriptive sentences generated by the models. In the following subsections, we discuss the evaluation metrics commonly used in RSIC research.

1) BLEU

The biLingual evaluation understudy (BLEU) [61] was originally designed to evaluate the performance of machine translation models and is now also widely used in sequence generation tasks, including image captioning. The core idea of BLEU is to calculate the co-occurrences of the consecutive words (i.e., n-grams) between the candidate sentence and the reference sentence. To address the problem that shorter sentences are more likely to achieve higher scores, the brevity penalty (BP) is introduced to penalize candidate sentences that are shorter than the reference sentence:

$$BP = \begin{cases} 1, & if\ c > r \\ e^{(1-r/c)}, & if\ c \leq r \end{cases} \tag{28}$$

where r and c represent the number of words in the reference and candidate sentences, respectively. The BLEU score can then be calculated by the following equation:

$$BLEU = BP \cdot exp(\sum_{n=1}^{N} w_n log p_n) \tag{29}$$

where N represents the number of consecutive words, p_n is the modified n-gram precision, and w_n represents the weight for each modified n-gram precision. The result of the calculation of $N = n$ in the above equation is usually expressed as BLEU-n, where in practice n is usually taken as one of 1-4. Because of its effectiveness and easy implementation, BLEU is widely used for evaluating models in RSIC studies.

2) METEOR

METEOR [62] stands for metric for evaluation of translation with explicit ordering. This sequence generation evaluation criterion was designed to address some of the problems found in the more popular BLEU metric. Unlike BLEU, which seeks correlation at the corpus level, METEOR produces good correlation with human judgement at the sentence or segment level. The main idea of METEOR is to compute an alignment, i.e., a set of mappings between unigrams, in the generated sentence and the ground-truth sentence. Once the final alignment is computed, the unigram precision P and recall R are calculated as:

$$P = \frac{m}{w_c} \tag{30}$$

$$R = \frac{m}{w_r} \tag{31}$$

where m is the number of unigrams in the candidate sentence that are also found in the reference sentence and w_c and w_r are the number of unigrams in the candidate and reference sentences, respectively. Then, the harmonic mean F_{mean} is calculated to combine the precision and recall:

$$F_{mean} = \frac{10PR}{R + 9P} \tag{32}$$

with recall weighted 9 times more than precision. The above calculations consider only the congruity with respect to single words rather than larger segments that appear in both the reference and the candidate sentence. A penalty p is designed to address this issue:

$$p = 0.5 \left(\frac{c}{u_m}\right)^3 \tag{33}$$

where u_m is the number of unigrams that have been mapped and c is the number of chunks, where a chunk is defined as a set of unigrams that are adjacent in the candidate and reference sentences. The final result M is calculated as:

$$M = F_{mean}(1 - p) \tag{34}$$

3) ROUGE-L

The design of recall-oriented understudy for gisting evaluation (ROUGE [63]) is inspired by BLEU, with the difference that recall rather than precision is utilized as the main criteria. ROUGE is a metric set containing ROUGE-N, ROUGE-L, ROUGE-W, and ROUGE-S. Among them, ROUGE-L, which measures the longest common subsequence (LCS) [82] between the candidate and reference sentences, is widely used in RSIC studies. $LCS(S_r, S_c)$ is denoted as the length of the LCS of the reference sentence S_r and the candidate sentence S_c. The recall R_{lcs} and precision P_{lcs} are calculated as:

$$R_{lcs} = \frac{LCS(S_r, S_c)}{len(S_r)} \tag{35}$$

$$P_{lcs} = \frac{LCS(S_r, S_c)}{len(S_c)} \tag{36}$$

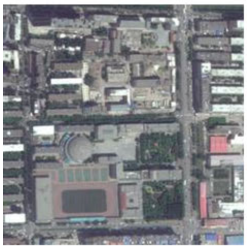
where $len(S)$ is the length of sentence S. The ROUGE-L score R_L can be calculated as:

$$R_L = \frac{(1 + \beta^2) R_{lcs} P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}} \tag{37}$$

In practice, β is often set to a very large number; therefore, only R_{lcs} is taken into account.

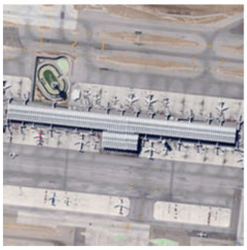
4) CIDEr

Consensus-based image description evaluation (CIDEr [64]) is an evaluation metric specifically designed for image captioning tasks. "Consensus-based" means that n-grams in all multiple ground-truth sentences of an image are taken into account when evaluating a candidate sentence. Moreover, n-grams that appear frequently in all images should be given lower weights since they usually do not contain



1. two exercise yards are seperated by a road with plants and buildings.
2. an outdoor court locate in one of six blocks.
3. many buildings and some green trees are around two playground separately.
4. many buildings and some green trees are in six different blocks and half of them with play grounds.
5. two playgrounds are surrounded by many trees and buildings.

FIGURE 17. An example image in the RSICD dataset annotated with five different sentences.



1. many planes are parked next to a long building in an airport.
2. many planes are parked next to a long building in an airport.
3. many planes are parked next to a long building in an airport.
4. many planes are parked next to a long building in an airport.
5. many planes are parked next to a long building in an airport.

FIGURE 18. An example image in the RSICD dataset annotated with only one sentence. The number of annotated sentences is expanded to five by duplicating the existing sentence.

valuable information. Based on the above considerations, a TF-IDF [66] weighting $g_k(s_{ij})$ for each n-gram w_k is designed as follows:

$$g_k(s_{ij}) = \frac{h_k(s_{ij})}{\sum_{w_l \in \Omega} h_l(s_{ij})} log\left(\frac{|I|}{\sum_{I_p \in I} min(1, \sum_q h_k(s_{pq}))}\right) \tag{38}$$

where $h_k(s_{ij})$ denotes the number of times an n-gram w_k occurs in a reference sentence s_{ij}, Ω is the vocabulary of all n-grams, and I is the set of all images in the dataset. The TF of each n-gram w_k is measured in the first term, while the second term measures the rarity of w_k using the IDF. The introduction of g_k ensures that high-frequency n-grams occurring in the ground-truth sentences are given high weights by the TF, while the IDF reduces the weights of those n-grams that occur frequently across all images. Then, the score for n-grams of length n is obtained by calculating the average cosine similarity between the candidate sentence c_i and the reference sentences s_{ij}:

$$CIDEr_n(c_i, Si) = \frac{1}{m} \sum_j \frac{g^n(c_i) \cdot g^n(s_{ij})}{\parallel g^n(c_i) \parallel \parallel g^n(s_{ij}) \parallel} \tag{39}$$

where $g^n(\cdot)$ represents a TF-IDF vector calculated by Equation (38) corresponding to all n-grams of length n and the $\parallel \cdot \parallel$ operation calculates the magnitude of a vector. The final CIDEr score is the combination of n-grams with various lengths:

$$CIDEr(c_i, S_i) = \sum_{n=1}^{N} w_n CIDEr_n(c_i, S_i) \tag{40}$$

where w_n is the weight for the $CIDEr_n$ score and is usually set to $1/N$. In practice, N is commonly set to 4.

5) SPICE

Semantic propositional image caption evaluation (SPICE [65]) is a graph-based evaluation metric for image captioning tasks. Different from the abovementioned n-gram matching-based metrics, SPICE transforms each caption c into a scene graph:

$$G(c) =< O(c), E(c), K(c) > \tag{41}$$

where $O(c)$ is a set of objects that appear in the image, $E(c)$ is the set of hyperedges representing relations between objects, and $K(c)$ is the set of attributes associated with objects. The function T is designed to transform a scene graph into logical tuples:

$$T(G(c)) \triangleq O(c) \cup E(c) \cup K(c). \tag{42}$$

Each tuple contains 1-3 elements representing objects, attributes and relations. Then, the precision P, recall R and the final SPICE score are calculated as:

$$P(c, S) = \frac{|T(G(c)) \otimes T(G(S))|}{|T(G(c))|} \tag{43}$$

$$R(c, S) = \frac{|T(G(c)) \otimes T(G(S))|}{|T(G(S))|} \tag{44}$$

$$SPICE(c, S) = F_1(c, S) = \frac{2 \cdot P(c, S) \cdot R(c, S)}{P(c, S) + R(c, S)} \tag{45}$$

where the binary matching operator $\otimes$ returns matching tuples in two scene graphs.

6) HUMAN RATING

In early RSIC studies, due to the lack of large-scale annotated datasets, the human-rating method was utilized to evaluate the performance of models. For example, in [2], 10 human raters were asked to give one of four grades, i.e., ''without errors'', ''with minor errors'', ''related to the image'', and ''unrelated to the image'', to each caption generated by the model. Since the goal of the RSIC task is to generate human-readable descriptions for RSIs, the evaluations given by humans may be more valuable in some scenarios than automated evaluation metrics such as BLEU, METEOR, ROUGE, CIDEr, or SPICE. However, human evaluation is prone to subjectivity and cannot be carried out on a large scale because it is time- and labor-intensive. Therefore, after the large-scale benchmark datasets were published, researchers tended to use automated evaluation metrics rather than human ratings to verify the performance of their RSIC models.

VI. COMPARISON OF STATE-OF-THE-ART METHODS

A. EXPERIMENTAL RESULTS

A brief comparison and analysis of the state-of-the-art methods in the RSIC literature is presented in this section. The specific experimental results of each method on the three benchmark datasets are shown in Table 5. The highest scores for each evaluation metric are shown in bold font.

The first RSIC method proposed by Qu *et al.*, named multimodal [1], is based on a vanilla CNN encoder and LSTM decoder architecture. Subsequently, Lu *et al.* [3] introduced the attention mechanism into the RSIC and achieved significantly better performance than the multimodal method. Afterwards, a number of RSIC methods [7], [9], [12], [22], [23], [30] focusing on optimizing the attention mechanism were proposed (see Section IV-C for details). Among them, multi-level AM [22] and structured AM [30] performed best. The experimental results show that the performance of the RSIC model can be effectively improved by introducing and optimizing AM.

In some solutions [19], [25], [26], [28], modules that can extract richer information (e.g., words, object features, region proposals, and classification labels) from RSIs are designed. As shown in Table 5, the valuable information extracted from RSI makes these methods perform better compared to the baseline methods [1], [3].

Compared to the other methods in Table 5 that adopt the generation based methods, the retrieval-based collective semantic metric learning framework (CSMLF) [6] achieved the worst score in the experiment. This demonstrates that although retrieval-based methods are valuable in some aspects, such as no grammatical errors in the output sentences, and can be utilized by downstream tasks such as bidirectional text-image retrieval, their performance measured by automatic evaluation metrics on benchmark datasets is far inferior to that of generation based methods.

Auxiliary components that can be integrated into any encoder-decoder RSIC framework are proposed in the studies of the PMM [21] and GCN [11] and significantly improve the overall performance of vanilla encoder-decoder [1] and attention [3] based methods. In addition, although the sound-a [8] method did not achieve particularly high scores in the experiment, the proposed ''active attention'' concept, i.e., guiding the generation of captions by additional sound information, has valuable practical applications and can be applied to other RSIC studies. The contributions of methods such as the three abovementioned methods [8], [11], [21] are not limited to their experimental results, but provide valuable modules or ideas that can be utilized by all other RSIC studies.

In terms of the training strategy and LM, the TCE [17] method significantly improves the encoder-decoder-based RSIC model by setting a truncation threshold for the calculation of CE loss. Notably, in the experiment of [17], TCE optimization is only applied to the vanilla CNN-LSTM framework, and it is highly expected that TCE can be combined with other RSIC methods to achieve better experimental results. In Actor-Dual-Critics (ADC) [14], Transformer+self-critical sequence training (SCST) [15], and the variational autoencoder and reinforcement learning-based two-stage multitask learning model (VRTMM) [16], reinforcement learning is introduced into RSIC, and a transformer is utilized as the LM instead of LSTM in [15], [16]. These proposals substantially enhance the overall performance of the RSIC models. According to the results of various evaluation metrics, the VRTMM [16] achieves the best performance in the experiment because of its combined use of reinforcement learning, transformer and multitask-based training of the CNN encoder.

Notably, we used the scores from the original papers of each study directly in Table 5. Therefore, the scores of multimodal methods on the RSICD dataset and the scores of VRTMM on UCM-captions and Sydney-captions are not contained in the table. In addition, some methods in the table could be expected to achieve better evaluation scores in a more optimized experimental setting than the original papers. Moreover, some valuable methods [2], [4], [5], [10], [13], [20], [27], [29] are not included in the table due to their lack of experimental results on the three benchmark datasets.

B. ADVANTAGES, DRAWBACKS AND APPLICATION SCENARIOS

Table 6 shows the advantages, drawbacks and application scenarios of different types of RSIC methods.

In the field of image captioning [89]–[92], template-based and retrieval-based approaches offer irreplaceable advantages. Manually defined sentence structures or directly retrieving methods make grammatical or lexical errors rare for the generated descriptions. In some tasks with explicit requirements for sentence format and content, well-designed templates can be used to generate super-compliant descriptions for RSIs. However, these two types of approaches are rarely found in the RSIC literature due to the inability to generate flexible and variable sentences and the lack of ability to learn lexical sequence features from large corpora. The

TABLE 5. Experimental results of the state-of-the-art methods. B1–B4 denote BLEU1–BLEU4 and M, R, C, and S represent METEOR, ROUGE-L, CIDEr, and SPICE respectively.

Dataset	Method	B1	B2	B3	B4	M	R	C	S
UCM-captions	multimodal [1]	0.638	0.536	0.377	0.219	0.206	-	0.451	-
	soft AM [3]	0.746	0.660	0.595	0.539	0.396	0.724	2.629	-
	CSMLF [6]	0.436	0.273	0.186	0.121	0.132	0.393	0.223	0.076
	attribute AM [7]	0.815	0.757	0.694	0.646	0.424	0.763	3.186	-
	sound-a [8]	0.783	0.728	0.676	0.633	0.380	0.686	2.906	0.420
	VAA [9]	0.819	0.751	0.693	0.639	0.438	0.782	3.395	-
	GCN [11]	0.833	0.771	0.715	0.662	0.437	0.776	3.168	-
	LAM [12]	0.819	0.776	0.748	**0.716**	0.484	0.791	3.617	0.502
	ADC [14]	0.853	0.757	0.678	0.612	**0.832**	0.808	**4.865**	-
	Transformer+SCST [15]	0.838	0.790	0.744	0.701	0.446	0.776	3.565	-
	TCE [17]	0.821	0.762	0.714	0.670	0.478	0.757	2.855	-
	RTRMN [18]	0.803	0.732	0.682	0.639	0.426	0.773	3.127	0.453
	DMSFF [19]	0.831	0.760	0.697	0.634	-	0.732	3.296	-
	PMM [21]	0.862	-	-	0.712	0.482	**0.825**	3.654	**0.536**
	multi-level AM [22]	**0.875**	**0.829**	**0.769**	0.705	0.528	0.816	3.079	0.462
	scene AM [23]	0.822	0.765	0.717	0.674	0.440	0.778	3.228	-
	SD-RSIC [24]	0.748	0.664	0.598	0.538	0.390	0.695	2.132	-
	multiscale methods [25]	0.834	0.782	0.741	0.702	0.450	0.792	3.257	-
	Word-Sentence [26]	0.793	0.724	0.667	0.620	0.439	0.713	2.787	-
	Instance Aware [28]	0.823	0.768	0.710	0.659	-	0.756	3.192	-
	structured AM [30]	0.854	0.803	0.757	0.715	0.463	0.814	3.350	-
Sydney-captions	multimodal [1]	0.548	0.398	0.228	0.215	0.208	-	0.379	-
	soft AM [3]	0.728	0.638	0.563	0.501	0.387	0.715	2.118	-
	CSMLF [6]	0.600	0.458	0.387	0.343	0.248	0.502	0.756	0.262
	attribute AM [7]	0.814	0.735	0.659	0.581	0.411	0.719	2.302	-
	sound-a [8]	0.716	0.632	0.547	0.466	0.313	0.604	1.803	0.387
	VAA [9]	0.743	0.665	0.603	0.549	0.393	0.700	2.407	-
	GCN [11]	0.823	**0.755**	0.659	0.600	0.420	0.724	2.311	-
	LAM [12]	0.740	0.655	0.590	0.530	0.369	0.681	2.352	0.404
	Transformer+SCST [15]	0.802	0.710	0.631	0.572	0.418	0.726	**2.539**	-
	TCE [17]	0.794	0.730	**0.672**	**0.619**	0.443	0.713	2.404	-
	DMSFF [19]	**0.830**	0.742	0.665	0.593	-	0.707	3.182	-
	PMM [21]	0.816	-	-	0.556	0.415	0.736	2.365	**0.430**
	multi-level AM [22]	0.806	0.719	0.645	0.582	**0.466**	**0.747**	2.203	0.401
	scene AM [23]	0.786	0.698	0.626	0.561	0.381	0.709	2.505	-
	SD-RSIC [24]	0.761	0.666	0.586	0.517	0.366	0.657	1.690	-
	multiscale methods [25]	0.751	0.680	0.615	0.556	0.367	0.702	2.243	-
	Word-Sentence [26]	0.789	0.709	0.632	0.562	0.418	0.692	2.041	-
	Instance Aware [28]	0.817	0.742	0.657	0.591	-	0.721	2.291	-
	structured AM [30]	0.779	0.702	0.639	0.586	0.395	0.730	2.379	-
RSICD	soft AM [3]	0.646	0.507	0.411	0.340	0.333	0.616	1.819	-
	CSMLF [6]	0.576	0.386	0.283	0.222	0.213	0.446	0.530	0.200
	attribute AM [7]	0.757	0.634	0.538	0.461	0.351	0.646	2.356	-
	sound-a [8]	0.620	0.482	0.390	0.320	0.273	0.514	1.638	0.360
	GCN [11]	0.760	0.642	0.552	0.462	0.354	0.656	2.361	-
	ADC [14]	0.740	0.552	0.463	0.410	0.221	0.713	2.243	-
	Transformer+SCST [15]	0.770	0.655	0.563	0.488	0.370	0.670	2.684	-
	LAM [12]	0.675	0.554	0.469	0.403	0.325	0.582	2.285	0.464
	VRTMM [16]	**0.793**	**0.679**	**0.588**	**0.511**	0.373	0.680	**2.793**	-
	TCE [17]	0.761	0.636	0.547	0.479	0.343	0.669	2.467	-
	RTRMN [18]	0.620	0.462	0.364	0.297	0.283	0.554	1.515	0.332
	PMM [21]	0.736	-	-	0.454	0.373	0.660	2.634	**0.477**
	multi-level AM [22]	0.791	0.678	0.574	0.503	**0.464**	**0.725**	2.631	0.455
	scene AM [23]	0.625	0.463	0.364	0.297	0.253	0.474	0.809	-
	SD-RSIC [24]	0.644	0.474	0.369	0.300	0.249	0.523	0.794	-
	multiscale methods [25]	0.687	0.553	0.460	0.392	0.301	0.566	1.668	-
	Word-Sentence [26]	0.724	0.586	0.493	0.425	0.320	0.626	2.063	-
	Instance Aware [28]	0.770	0.649	0.532	0.471	-	0.651	2.363	-
	structured AM [30]	0.702	0.561	0.465	0.393	0.329	0.571	1.703	-

relevant studies [2], [6] reviewed in this article are detailed in Section IV-D.

Most of the existing RSIC studies (all but [2] and [6]) have adopted the Encoder-Decoder based architecture because they can benefit from deep learning techniques and large-scale datasets. Well-designed encoders and decoders are able to learn and extract features from RSIs and text descriptions to generate new sentences with variable syntax. The drawback of this type of method compared to template- and retrieval-based approaches is the possibility of syntactic or lexical errors occurring in the generated sentences. See Section IV-A for a discussion of the Encoder-Decoder based architecture, Section IV-B for a discussion of the extraction of RSI features by different encoders, and Section IV-D for a discussion of the design of different structured LMs as decoders.

Optimizing AM for RSIC models [3], [7], [12], [22], [23], [30] provides an improved alignment of each word or phrase in the generated sentences with the features in RSIs. Training AM-based models requires well annotated sentences and large-scale datasets to enable the model to learn enough alignment relationships between RSI features

TABLE 6. Comparison of the advantages, drawbacks, and applicable scenarios of the existing RSIC methods.

Methods	Advantages	Drawbacks	Application Scenarios
Template-based	• No syntactic or semantic errors • Well-designed templates provide good reading experiences	• Unable to generate flexible and versatile sentences • Labor-intensive construction and maintenance of templates	• Specific tasks with fixed sentence formats and clear purposes
Retrieval-based	• No syntactic or semantic errors • Can be applied to other tasks such as bi-directional retrieval of RSIs and text	• Unable to generate new sentences • High requirement for sample richness of dataset	• Scenarios where the input RSI content is highly similar to the samples in the retrieval database
Encoder-Decoder Architecture	• Can generate descriptions with variable sentence structures, even new sentences that do not exist in the corpus • Benefit from new deep learning technologies with high potential for enhancement	• May generate sentences with grammatical or syntactic errors • Possible overlearning of lexical sequence relations in the corpus and weakening the relationship between RSI features and words	• Large-scale and well annotated RSIC datasets
Attention Mechanism	• Efficiently learning the alignment between image details and words • Interpretable process of generating each word	• May learn false matches between RSI features and text on small-sized datasets • High quality requirement for text annotation	• Well annotated datasets with enough RSI features and text information to align
Reinforcement Learning	• Optimize directly with the test-time evaluation metrics • Solve the overfitting problems and generate better sentences	• Much higher complexity than supervised learning-based methods • Training is unstable and slow	• Scenarios that require special optimization for particular metrics
Active Attention	• Allows the use of additional information such as sound, topic words, etc. to guide caption generation	• Additional data (sound, topic words, etc.) required	• Interesting and valuable real-life scenarios, such as generating targeted descriptions for RSI by voice and keywords, etc.
Auxiliary Component	• Integratable into other RSIC frameworks • Provide special optimizations or additional functionality	• Additional space and time complexity	• Improving the performance of RSIC models for particular aspects or special tasks

and text; otherwise, it may counterproductively learn invalid matches. Improving the structure of AM modules or feeding them with more valuable information can lead to enhanced model performance, as discussed in Section IV-C.

Compared to the solutions of the traditional CE-based training strategy, although requiring greater computational and training overhead, the reinforcement learning-based RSIC solutions [14], [15], [29] can directly optimize the parameters of the models with the metrics (e.g., CIDEr and others) used at the test time. Proper use of this training strategy allows for the targeted optimization of RSIC models to generate better sentences. See Section IV-G for a detailed discussion.

Although additional data such as voice and topic words, are required for training RSIC models with active attention [8], [18], the trained models can be adjusted by voice, keywords, etc. to guide the generation of captions. This feature supports interesting and valuable applications in real-world scenarios, such as generating targeted descriptions for captured RSIs through speech guidance during military or reconnaissance missions. A detailed discussion can be found in Section IV-E.

The auxiliary component-based approaches [11], [21], [24] aim to achieve specific optimization by designing modules that can be flexibly integrated into generic encoder-decoder-based RSIC models. With additional time or space complexity, such methods can provide special optimizations or additional functionality. See Section IV-F for details.

VII. DISCUSSION AND FUTURE RESEARCH DIRECTIONS

A. RSIC METHODS

Except for a few studies that have adopted template or retrieval-based methods, the majority of RSIC studies have adopted generation methods with an encoder-decoder architecture. This is because encoder-decoder methods can automatically and jointly learn image and text features in large-scale datasets to generate more variable and flexible sentences. The introduction of an attention mechanism greatly enhances the encoder-decoder-based RSIC approaches, as it allows the models to focus on key regions in the image when generating specific words. Some studies in the literature have focused on improving the structure of the attention mechanisms by utilizing more information from RSIs and context and have achieved better results.

From the aspect of image feature extraction, researchers have focused on extracting multilevel features from the layers of CNN encoders with different depths, and obtained richer image information through multitask approaches. These methods are well suited to meet the challenge of variable object sizes in RSIs. Moreover, since comparing different CNNs is not the focus of research, VGG and ResNet are often selected as the backbone of RSI encoders in many studies for a fair comparison with other works.

LSTM has been employed as the LM in most RSIC studies due to its ability to effectively learn sequence relations in a large-scale corpus and to solve the vanishing and exploding gradient problem of traditional RNN models. In some recent studies, transformers have been chosen as an alternative and outperformed LSTM in both training time and performance. This new kind of LM can be expected to achieve better results in more RSIC studies in the future. In addition, CNN-based LMs can be utilized in combination with specific frameworks to achieve good applications. Template- and retrieval-based approaches, although generating less flexible and various captions, can still be expected to play important roles in some scenarios, such as generating formatted descriptions for specific tasks.

The conventional CE-based training strategy treats RSIC model training as a multiclassification task. The probability of the target word at each time step is forced to be 1. Valuable non-target words such as synonyms are excluded, which leads to overfitting problems. In addition, the evaluation metrics at test time are not directly applied during training. Improving the training strategy by modified CE loss calculation and the reinforcement learning-based method effectively solves these problems. In addition, auxiliary components that can be integrated into any encoder-decoder-based model and ideas with potential applications such as active attention for guiding the process of caption generation with extra information are proposed, which contribute positively to the field of RSIC.

Although many excellent studies have been proposed in the literature of RSIC, there are still issues that need to be addressed in the future. Existing supervised learning-based approaches rely too much on the scale of datasets and the quality of annotations. Methods that can leverage external knowledge and unsupervised approaches deserve to be explored. In addition, borrowing and introducing successful methods from the field of NIC can benefit RSIC research. However, some methods proposed in existing RSIC studies can be applied to both RSIC datasets and NIC datasets. The image characteristics and task characteristics of RSIC need further consideration. In future work, it would be more worthwhile to propose exclusively improved RSIC methods for the unique characteristics of RSIC data and tasks.

B. DATASETS

The images in the existing publicly available RSIC datasets are taken by satellites from high altitude, where the content distribution and coverage are as described in Section V-A.

From the perspective of images, all RSIs in the existing publicly available benchmark datasets are taken by satellites from super high altitude, and all contents are artificially constructed or natural surface objects. In recent years, the technology of high-altitude photography by UAVs is becoming increasingly mature, and images taken by UAVs from high altitude, which cover daily scenes such as traffic, rallies, disasters, and area patrol reconnaissance, can be applied to specific RSIC applications such as rescue and investigation. However, no large-scale well-labeled dataset of everyday scenes captured by UAVs is publicly available in the existing RSIC literature. This gap can be expected to be filled in future research.

For text annotation, the ground-truth captions in the existing RSIC datasets are manually annotated for research purposes, and certain instructions have been developed to ensure the quality of annotated sentences. However, many problems remain to be solved. First, there is a mismatch between the content of RSIs and the corresponding vocabulary size in the existing benchmark RSIC datasets, with some sentences being complex and others simple, and with many RSIs not annotated with enough five sentences, as shown in Figure 13, Figure 15 and Table 4. Second, the existing corpus is annotated for deep learning research instead of real applications. Therefore, some important information such as direction is simply ignored in the annotation process, and the annotated sentences tend to be simple, containing only the main objects and attributes. In addition, there are spelling and grammatical errors in the existing annotated sentences.

Many valuable studies can be expected to appear in future work. On one hand, supplementing and improving the RSIs and sentences in the existing datasets have positive implications for RSIC research. On the other hand, taking images of daily scenes (e.g., road traffic and public places) and special scenes (e.g., disaster and rescue) at different heights using satellites and UAVs and purposefully annotating them in combination with specific application scenarios can give rise to many valuable studies. In addition, new annotating methods in conventional IC studies such as stylized captioning [88] can be expected to be borrowed to construct RSIC datasets.

C. EVALUATION METRICS

The performance of the RSIC models in the literature are mainly evaluated using the metrics described in Section V-B, i.e., BLEU, METEOR, ROUGE, CIDEr, and SPICE. All of these metrics compare the captions generated by the models with the ground-truth sentences, with higher scores representing higher quality. BLEU, which is easy to implement and the most widely used metric, measures the ability of the models to generate continuous words in the ground-truth sentences. METEOR focuses on smaller semantic segments and accounts for synonyms. The basic idea of ROUGE is identical to that of BLEU but emphasizes recall instead of accuracy. Intuitively, ROUGE focuses more on the correctness rather than the fluency of the generated captions than BLEU. Unlike these three metrics, which were originally designed to evaluate machine translating methods, CIDEr and SPICE were specifically designed for evaluating image captioning models. In CIDEr, important words are given higher weights. SPICE represents and measures the relationships between phrases through graph structures.

In order to validate the proposed RSIC methods, the scores of multimodel [1] and CMSLF [6] are often compared as baselines. It is worth noting that CMLSF is a retrieval-based method which cannot generate new sentences that do not exist in the retrieval databases and therefore scores low in all automated evaluations. However, these types of methods have their own unique advantages, as detailed in Section IV-D.

Although the existing evaluation metrics can be utilized to validate, verify and compare the performance of RSIC models, they were originally designed for machine translation or NIC tasks, so there is room for enhancement from this perspective. Evaluation metrics that are specifically designed for RSIC tasks are expected to be proposed in future work, e.g., assigning higher weights to focal objects or attributes in the RSIs for particular scenarios or tasks, assigning higher scores to special word sequences in the corpus, proposing more efficient data structures to represent and evaluate the correspondences between content of RSIs and phrases, etc.

D. APPLICABLE SCENARIOS AND USE CASES

Compared to other studies that focus on remote sensing, RSIC is a relatively new topic. The existing studies of methods and datasets are in the preliminary exploration stage, although they have achieved impressive results. In future research, RSIC can be applied in many real-world scenarios and output interesting and valuable information to users.

On one hand, there have been many mature application studies for RSIs [93], such as scene classification, change detection, target detection, land-use and land-cover classification, etc.. Compared with these applications, whose outputs are simply labels or a few words, RSIC can provide users with richer semantic information and excellent reading experience. A reliable and easy-to-implement idea is to build RSIC datasets by annotating descriptive sentences for the RSIs in the existing large-scale datasets of these traditional mature applications, and applying targeted and modified methods from the NIC domain to the newly constructed datasets. Most of the existing RSIC studies are formally conducted along this direction.

On the other hand, in addition to the above research ideas, RSIC studies for specific scenarios are worth exploring. Some examples are road traffic management, security monitoring of public areas, disaster scene rescue, identification of drugs and other plants. In such special scenarios, RSIs taken by satellites or UAVs from high altitude contain super valuable information that ordinary camera photos do not have. It makes sense to perform RSIC studies to generate human-readable and semantically rich captions for these RSIs.

VIII. CONCLUSION

RSIC is an interesting and valuable topic that has emerged in recent years. The existing studies in this field have achieved remarkable results, but many issues must be addressed in future work. In this article, we present a systematic survey of the literature on RSIC. To the best of our knowledge, this is the first review article in this field. Our main findings are as follows.

First, 30 high-quality papers in the field were conditionally filtered and obtained. Second, existing studies are grouped according to technical solutions, and the fundamental theory, pros and cons and trends of the methods in each group are described and presented. Third, the construction of existing RSIC datasets, commonly used evaluation metrics, and experimental results of the state-of-the-art methods are comparatively analyzed and presented. Finally, trends in existing RSIC research, pressing issues to address in future work, and valuable use cases that are worth studying are discussed and presented.

The findings in this article can be expected to provide valuable reference information for relevant researchers. With advances in deep learning, RSI processing, and natural language processing, RSIC research is expected to attract the participation of more researchers.

REFERENCES

[1] B. Qu, X. Li, D. Tao, and X. Lu, "Deep semantic understanding of high resolution remote sensing image," in *Proc. Int. Conf. Comput., Inf. Telecommun. Syst. (CITS)*, Jul. 2016, pp. 1–5.

[2] Z. Shi and Z. Zou, "Can a machine generate humanlike language descriptions for a remote sensing image?" *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3623–3634, Jun. 2017.

[3] X. Lu, B. Wang, X. Zheng, and X. Li, "Exploring models and data for remote sensing image caption generation," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2183–2195, Apr. 2017.

[4] X. Zhang, X. Li, J. An, L. Gao, B. Hou, and C. Li, "Natural language description of remote sensing images based on deep learning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 4798–4801.

[5] S. Wang, J. Chen, and G. Wang, "Intensive positioning network for remote sensing image captioning," in *Proc. Int. Conf. Intell. Sci. Big Data Eng.* Cham, Switzerland: Springer, 2018, pp. 567–576.

[6] B. Wang, X. Lu, X. Zheng, and X. Li, "Semantic descriptions of high-resolution remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 8, pp. 1274–1278, Aug. 2019.

[7] X. Zhang, X. Wang, X. Tang, H. Zhou, and C. Li, "Description generation for remote sensing images using attribute attention mechanism," *Remote Sens.*, vol. 11, no. 6, p. 612, 2019.

[8] X. Lu, B. Wang, and X. Zheng, "Sound active attention framework for remote sensing image captioning," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 1985–2000, Mar. 2020.

[9] Z. Zhang, W. Zhang, W. Diao, M. Yan, X. Gao, and X. Sun, "VAA: Visual aligning attention model for remote sensing image captioning," *IEEE Access*, vol. 7, pp. 137355–137364, 2019.

[10] X. Zhang, Q. Wang, S. Chen, and X. Li, "Multi-scale cropping mechanism for remote sensing image captioning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 10039–10042.

[11] 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.

[12] Z. Zhang, W. Diao, W. Zhang, M. Yan, X. Gao, and X. Sun, "LAM: Remote sensing image captioning with label-attention mechanism," *Remote Sens.*, vol. 11, no. 20, p. 2349, Oct. 2019.

[13] S. C. Kumar, M. Hemalatha, S. B. Narayan, and P. Nandhini, "Region driven remote sensing image captioning," *Proc. Comput. Sci.*, vol. 165, pp. 32–40, Jan. 2019.

[14] R. Chavhan, B. Banerjee, X. Xiang Zhu, and S. Chaudhuri, "A novel actor dual-critic model for remote sensing image captioning," 2020, *arXiv:2010.01999*.

[15] X. Shen, B. Liu, Y. Zhou, and J. Zhao, "Remote sensing image caption generation via transformer and reinforcement learning," *Multimedia Tools Appl.*, vol. 79, nos. 35–36, pp. 26661–26682, Sep. 2020.

[16] X. Shen, B. Liu, Y. Zhou, J. Zhao, and M. Liu, "Remote sensing image captioning via variational autoencoder and reinforcement learning," *Knowl.-Based Syst.*, vol. 203, Sep. 2020, Art. no. 105920.

[17] X. Li, X. Zhang, W. Huang, and Q. Wang, "Truncation cross entropy loss for remote sensing image captioning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5246–5257, Jun. 2021.

[18] B. Wang, X. Zheng, B. Qu, and X. Lu, "Retrieval topic recurrent memory network for remote sensing image captioning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 256–270, 2020.

[19] W. Huang, Q. Wang, and X. Li, "Denoising-based multiscale feature fusion for remote sensing image captioning," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 3, pp. 436–440, Mar. 2021.

[20] G. Hoxha, F. Melgani, and J. Slaghenauffi, "A new CNN-RNN framework for remote sensing image captioning," in *Proc. Medit. Middle-East Geosci. Remote Sens. Symp. (M GARSS)*, Mar. 2020, pp. 1–4.

[21] K. Fu, Y. Li, W. Zhang, H. Yu, and X. Sun, "Boosting memory with a persistent memory mechanism for remote sensing image captioning," *Remote Sens.*, vol. 12, no. 11, p. 1874, Jun. 2020.

[22] Y. Li, S. Fang, L. Jiao, R. Liu, and R. Shang, "A multi-level attention model for remote sensing image captions," *Remote Sens.*, vol. 12, no. 6, p. 939, Mar. 2020.

[23] S. Wu, X. Zhang, X. Wang, C. Li, and L. Jiao, "Scene attention mechanism for remote sensing image caption generation," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–7.

[24] G. Sumbul, S. Nayak, and B. Demir, "SD-RSIC: Summarization-driven deep remote sensing image captioning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6922–6934, Aug. 2021.

[25] X. Ma, R. Zhao, and Z. Shi, "Multiscale methods for optical remote-sensing image captioning," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 11, pp. 2001–2005, Nov. 2021.

[26] Q. Wang, W. Huang, X. Zhang, and X. Li, "Word-sentence framework for remote sensing image captioning," *IEEE Trans. Geosci. Remote Sens.*, early access, Dec. 25, 2020, doi: 10.1109/TGRS.2020.3044054.

[27] G. Hoxha and F. Melgani, "Remote sensing image captioning with SVM-based decoding," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Sep. 2020, pp. 6734–6737.

[28] C. Wang, Z. Jiang, and Y. Yuan, "Instance-aware remote sensing image captioning with cross-hierarchy attention," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Sep. 2020, pp. 980–983.

[29] Q. Yang, G. Wang, X. Zhang, C. Grecos, and P. Ren, "Coastal image captioning," *J. Coastal Res.*, vol. 102, no. S1, pp. 145–150, Dec. 2020.

[30] R. Zhao, Z. Shi, and Z. Zou, "High-resolution remote sensing image captioning based on structured attention," *IEEE Trans. Geosci. Remote Sens.*, early access, Apr. 12, 2021, doi: 10.1109/TGRS.2021.3070383.

[31] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10. Cambridge, MA, USA: MIT Press, 1995, p. 1995.

[32] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[34] Sivic and Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, vol. 3, Oct. 2003, p. 1470.

[35] F. Perronnin, Y. Liu, J. Sanchez, and H. Poirier, "Large-scale image retrieval with compressed Fisher vectors," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3384–3391.

[36] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1704–1716, Sep. 2012.

[37] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[39] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[40] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*.

[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25, Dec. 2012, pp. 1097–1105.

[42] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[43] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[44] 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. (CVPR)*, Jun. 2016, pp. 2818–2826.

[45] 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. 4700–4708.

[46] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, Nov. 2014, pp. 675–678.

[47] W. Liu, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[48] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," 2015, *arXiv:1506.01497*.

[49] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.

[50] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst. (GIS)*, 2010, pp. 270–279.

[51] 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.

[52] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," *Trans. Assoc. Comput. Linguistics*, vol. 2, pp. 67–78, Feb. 2014.

[53] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick, "Microsoft COCO captions: Data collection and evaluation server," 2015, *arXiv:1504.00325*.

[54] Dhaksha Team. (2018). *Drone Manufacture in INDIA—Team Dhaksha*. [Online]. Available: https://www.teamdhaksha.com/

[55] D. Graff and C. Cieri, "English gigaword LDC2003T05," Linguistic Data Consortium, Philadelphia, PA, USA, 2003, doi: 10.35111/0z6y-q265.

[56] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS J. Photogramm. Remote Sens.*, vol. 159, pp. 296–307, Jan. 2020.

[57] Y.-W. Chen, K.-H. Yap, and J. Y. Lee, "Tianditu: China's first official online mapping service," *Media, Culture Soc.*, vol. 35, no. 2, pp. 234–249, Mar. 2013.

[58] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[59] R. J. Lisle, "Google Earth: A new geological resource," *Geol. Today*, vol. 22, no. 1, pp. 29–32, Jan. 2006.

[60] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.

[61] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, ''BLEU: A method for automatic evaluation of machine translation,'' in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2001, pp. 311–318.

[62] M. Denkowski and A. Lavie, ''Meteor universal: Language specific translation evaluation for any target language,'' in *Proc. 9th Workshop Stat. Mach. Transl.*, 2014, pp. 376–380.

[63] C.-Y. Lin, ''ROUGE: A package for automatic evaluation of summaries,'' in *Proc. Text Summarization Branches Out*. Barcelona, Spain: Assoc. Comput. Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://aclanthology.org/W04-1000.pdf

[64] R. Vedantam, C. L. Zitnick, and D. Parikh, ''CIDEr: Consensus-based image description evaluation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4566–4575.

[65] P. Anderson, ''Spice: Semantic propositional image caption evaluation,'' in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 382–398.

[66] S. Robertson, ''Understanding inverse document frequency: On theoretical arguments for IDF,'' *J. Document.*, vol. 60, no. 5, pp. 503–520, Oct. 2004.

[67] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, ''Learning internal representations by error propagation,'' La Jolla Inst. Cogn. Sci., California Univ., San Diego, CA, USA, Tech. Rep. ICS 8506, 1985. [Online]. Available: https://www.utm.mx/~jjf/rna/A1%20Learning%20representations%20by%20back-propagating%20errors.pdf

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

[69] I. Sutskever, O. Vinyals, and Q. V. Le, ''Sequence to sequence learning with neural networks,'' 2014, *arXiv:1409.3215*.

[70] A. Graves, ''Generating sequences with recurrent neural networks,'' 2013, *arXiv:1308.0850*.

[71] 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. (CVPR)*, Jun. 2015, pp. 3156–3164.

[72] 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.

[73] T. N. Kipf and M. Welling, ''Semi-supervised classification with graph convolutional networks,'' 2016, *arXiv:1609.02907*.

[74] J. Lu, C. Xiong, D. Parikh, and R. Socher, ''Knowing when to look: Adaptive attention via a visual sentinel for image captioning,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 375–383.

[75] L. Zhang, F. Sung, F. Liu, T. Xiang, S. Gong, Y. Yang, and T. M. Hospedales, ''Actor-critic sequence training for image captioning,'' 2017, *arXiv:1706.09601*.

[76] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, ''Self-critical sequence training for image captioning,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7008–7024.

[77] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, ''Attention is all you need,'' 2017, *arXiv:1706.03762*.

[78] C. C. Park, B. Kim, and G. Kim, ''Attend to you: Personalized image captioning with context sequence memory networks,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 895–903.

[79] H. Christian, M. P. Agus, and D. Suhartono, ''Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF),'' *ComTech, Comput., Math. Eng. Appl.*, vol. 7, no. 4, pp. 285–294, 2016.

[80] D. J. McQueen, M. R. S. Johannes, J. R. Post, T. J. Stewart, and D. R. S. Lean, ''Bottom-up and top-down impacts on freshwater pelagic community structure,'' *Ecol. Monographs*, vol. 59, no. 3, pp. 289–309, Sep. 1989.

[81] A. See, P. J. Liu, and C. D. Manning, ''Get to the point: Summarization with pointer-generator networks,'' 2017, *arXiv:1704.04368*.

[82] T. H. Cormen, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.

[83] Z. Zou and Z. Shi, ''Ship detection in spaceborne optical image with SVD networks,'' *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 5832–5845, Oct. 2016.

[84] Z. An, Z. Shi, X. Teng, X. Yu, and W. Tang, ''An automated airplane detection system for large panchromatic image with high spatial resolution,'' *Optik*, vol. 125, no. 12, pp. 2768–2775, 2014.

[85] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, ''Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning,'' *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.

[86] V. Risojević and Z. Babić, ''Unsupervised quaternion feature learning for remote sensing image classification,'' *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 4, pp. 1521–1531, Apr. 2016.

[87] B. Demir and L. Bruzzone, ''Hashing-based scalable remote sensing image search and retrieval in large archives,'' *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 2, pp. 892–904, Sep. 2016.

[88] C. Gan, Z. Gan, X. He, J. Gao, and L. Deng, ''StyleNet: Generating attractive visual captions with styles,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3137–3146.

[89] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, and H. Laga, ''A comprehensive survey of deep learning for image captioning,'' *ACM Comput. Surveys*, vol. 51, no. 6, pp. 1–36, Feb. 2019.

[90] S. Bai and S. An, ''A survey on automatic image caption generation,'' *Neurocomputing*, vol. 311, pp. 291–304, Oct. 2018.

[91] R. Bernardi, ''Automatic description generation from images: A survey of models, datasets, and evaluation measures,'' *J. Artif. Intell. Res.*, vol. 55, pp. 409–442, Jan. 2016.

[92] A. Kumar and S. Goel, ''A survey of evolution of image captioning techniques,'' *Int. J. Hybrid Intell. Syst.*, vol. 14, no. 3, pp. 123–139, Mar. 2018.

[93] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, ''Deep learning in remote sensing applications: A meta-analysis and review,'' *ISPRS J. Photogramm. Remote Sens.*, vol. 152, pp. 166–177, Jun. 2019.

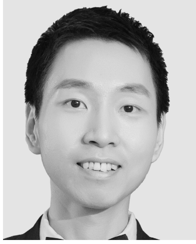
[94] A. Abdollahi, B. Pradhan, N. Shukla, S. Chakraborty, and A. Alamri, ''Deep learning approaches applied to remote sensing datasets for road extraction: A state-of-the-art review,'' *Remote Sens.*, vol. 12, no. 9, p. 1444, May 2020.

[95] Y. Li, H. Zhang, X. Xue, Y. Jiang, and Q. Shen, ''Deep learning for remote sensing image classification: A survey,'' *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 8, no. 6, p. e1264, 2018.

[96] J. Song, S. Gao, Y. Zhu, and C. Ma, ''A survey of remote sensing image classification based on CNNs,'' *Big Earth Data*, vol. 3, no. 3, pp. 232–254, Jul. 2019.

[97] G. Cheng and J. Han, ''A survey on object detection in optical remote sensing images,'' *ISPRS J. Photogramm. Remote Sens.*, vol. 117, pp. 11–28, Jul. 2016.

[98] D. Moher, ''Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement,'' *Ann. Internal Med.*, vol. 151, no. 4, Aug. 2009, Art. no. e1000097.



BEIGENG ZHAO received the B.S. degree in software engineering from Beihang University, China, in 2007, and the M.S. degree in software engineering from the University of Science and Technology of China, in 2013. He is currently a Lecturer with the College of Public Security Information Technology and Intelligence, Criminal Investigation Police University of China. His research interests include computer vision, natural language processing, and deep learning.