Integrating Pre-Trained Transformers in Encoder-Decoder Models for Remote Sensing Image Captioning

João Barata

***Abstract*—The application of deep learning methods in the analysis of remote sensing images has started to attract extensive attention. Noticeable progress has been made in tasks such as global scene classification and object detection. However, less attention has been given to the problem of describing remote sensing images with accurate and concise natural language sentences. Recent proposals have explored attention-based encoder-decoder frameworks, mostly focusing on improving the visual attention mechanism. Despite previous efforts, there are still many possibilities to improve on the quality of the generated captions. One possible approach is to consider recent developments on language models pre-trained on large amounts of data. Considering this, we propose a novel remote sensing image captioning framework that integrates pre-trained language models in a traditional attention-based encoder-decoder architecture. The integration is achieved through a fusion module that consists of the concatenation of the hidden states of an auxiliary language model and a LSTM decoder. Furthermore, in order to use an auxiliary language model pre-trained for summarization tasks, this work proposes a retrieval method to make use of additional information given by similar image captions. Experiments on the well known UCM, Sydney and RSICD datasets show improvements over simpler baselines and previous state-of-the-art models, showcasing the usefulness of pre-trained language models on the task.**

***Index Terms*—Remote sensing imagery, image captioning, image retrieval, image classification, deep learning.**

I. Introduction

DATA collected through remote sensing techniques have been extensively used in several Earth science disciplines, and it has supported numerous military, social and humanitarian applications. The development of remote sensing technology has led to an increase in the availability of high-resolution aerial/satellite imagery [48], which has attracted the attention of artificial intelligence researchers due to the possible application of deep learning methods to the analysis of these data.

Recent advances in computer vision and natural language processing, particularly related to the use of deep convolutional networks for image classification and understanding as well as attention-based and/or recurrent neural networks for text generation, have achieved state-of-the-art results in tasks such as image captioning. In brief, image captioning is concerned with the automatic generation of correct natural language descriptions for image contents. In the context of remote sensing imagery, this is a challenging but vital problem, as it can support practical applications, such as image retrieval with textual queries, or the generation of explanations for scene classification.

Throughout the years, many studies have been conducted in connection to natural (i.e., ground-level) image captioning [10, 37]. Researchers have typically approached the task with one of three different types of methods: templates [17, 12, 18, 51], retrieval from a database of examples [26, 24], and neural encoder-decoder [23, 44, 49, 15, 6] methods. The latter type of methods is currently the most common. Taking inspiration from the previous approaches for addressing general image captioning problems, researchers working with remote sensing images have also proposed to tackle the problem with methods based on deep neural networks. Most of these methods, as in general image captioning tasks, follow an encoder-decoder architecture that combines convolutional, recurrent, and attention components (see Figure 1). An encoder component corresponding to a pre-trained convolutional neural network generates a representation for the image, and a decoder based on a recurrent neural network generates the caption word-by-word, at each step using neural attention to weight parts of the visual input according to relevance for the current prediction.

Inspired by recent progress on language modeling, namely through the development of Transformer models, Kalimuthu et al. [14] proposed a novel architecture for image captioning that aimed to improve the generation of captions and their emendation. The approach involved a pre-trained convolutional neural network (CNN) encoder, a Long Short-Term Memory (LSTM) decoder, a pre-trained auxiliary language model (AuxLM), and a fusion module. The authors specifically focused on the task of emending captions, using a pre-trained BERT model [5] as the AuxLM. However, using a different language model (e.g., an auto-regressive text generation model like GPT-2 [28]), one can address caption generation as well. The authors showed that fusion models outperformed simpler baselines [49], with promising results.

In this paper, we propose a novel remote sensing image

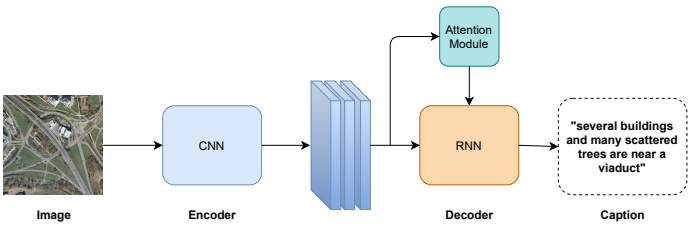


Fig. 1. Illustration of a traditional encoder-decoder architecture for image captioning featuring an attention mechanism over the visual features.

captioning framework that explores the idea of integrating a pre-trained AuxLM in a typical encoder-decoder architecture. We also developed a retrieval method that works in connection with a pre-trained AuxLM for summarization tasks, namely we argue that integrating retrieval and generation is particularly advantageous to the remote sensing image captioning task, since the templated nature of the currently available datasets leads to a large number of captions that are identical in terms of phrase structure and vocabulary. Our method consists of retrieving similar images and their captions, afterwards using them to guide the caption generation. The retrieved captions are encoded with a fixed pre-trained PEGASUS encoder, and we fuse the hidden states of the PEGASUS decoder with the hidden states of a LSTM decoder at each time-step. Our results demonstrate that the fusion architecture surpasses previous approaches in remote sensing image captioning. In addition, we perform several experiments with a different AuxLM, using different attention components, and different training strategies (e.g., we explore the impact of using a supervised contrastive loss [16] whilst fine-tuning our image encoder with remote sensing images).

In summary, our main contributions are as follows:

- We propose a novel remote sensing image captioning approach, combining generation and retrieval-based methods, and using state-of-the-art models for the encoder, the decoder, and the visual attention components.
- We present a fusion architecture for remote sensing image captioning, integrating a pre-trained Transformer with a traditional encoder-decoder model.
- We extensively evaluate the proposed approach, assessing the impact of different model components and model training strategies, using a diverse set of metrics on existing datasets.

The rest of this document is organized as follows: Section 2 presents related work on remote sensing image captioning. Section 3 details the proposed approach. Section 4 describes the datasets, the evaluation metrics, the implementation details, and discusses the obtained results. Finally, Section 5 concludes the paper, summarizing the main findings and discussing possible directions for future work.

II. Related Work

Remote sensing image captioning studies has been heavily inspired by traditional image captioning approaches, following template-based, retrieval-based, or encoder-decoder approaches. Template-based approaches use predefined templates containing blank spots to be filled with the results of an object detection system. Earlier studies, such as those from Shi and Zou [36], explored this approach by proposing the usage of fixed language templates combined with a single fully convolutional network to identify key ground objects. However, despite being fast and robust, this approach is relatively poor in terms of capturing fine-grained semantics.

Retrieval-based methods look for similar images in a background dataset, afterwards using and/or combining the captions from similar images [46]. In spite of the interesting results, this type of method is tied to the coverage of the background collection and to the performance of the retrieval component which, if inaccurate, can produce grammatical incorrect captions. Recent image captioning studies have proposed to combine retrieval and generation methods, conditioning caption generation on image contents, together with retrieved captions [31]. In the remote sensing image captioning domain, recent work by Wang et al. [46, 45] proposed a retrieval-based approach with a topic recurrent memory network that makes use of topic information extracted from the image annotations to guide the caption generation.

Qu et al. [27], conducted the first experiments with an encoder-decoder approach for remote sensing image caption generation based on multimodal neural networks. The authors also presented the first datasets for the task, namely the UCM-captions and Sydney-captions datasets. Noting that these first datasets had a very reduced size and a limited vocabulary, Lu et al. [21] introduced a larger dataset of remote sensing images with textual descriptions, namely the Remote Sensing Image Captioning Dataset (RSICD). These authors also presented an attention-based method for remote sensing image captioning, achieving superior performance over methods with no visual attention and showcasing the importance of these mechanisms when working with remote sensing imagery.

After the work from Lu et al. [21], most published work in the area followed the standard encoder-decoder approach in conjunction with some tailored attention mechanism. For example, Zhang et al. [58] proposed a framework that makes use of attributes extracted from the last fully connected layer of a CNN encoder. An attribute-attention mechanism could leverage global information into the calculation of intermediate vectors, and consequently improve attention accuracy. Ma et al. [22] introduced two multiscale methods, namely, multiscale attention (MSA) and multifeature attention (MFA). The proposed methods alleviated the scale-diversity problem of remote sensing images. More recently, Zhao et al. [60] proposed a structured attention module for high-resolution remote sensing images, which uses segmentation proposals from a joint captioning and pixel-level segmentation framework. The segmentation proposals are encoded into the attention module and ultimately aid the model to focus on highly structured semantic contents present in the images. Another example is the study from Wang et al. [47], which introduced a method with instance-awareness and cross-hierarchy attention. The authors first proposed a multi-level feature extractor to deal with the high complexity of remote sensing images. To accomplish this, in a first phase, a Faster R-CNN [33] is used to locate possible regions containing key objects and their respective neighbourhood. Secondly, the last fully-connected layer of a ResNet-101 [8] encoder is also used as a global feature extractor to deal with less complex images (e.g., desert or ocean scenes). Finally, the extracted features are dynamically re-weighted by a cross-hierarchy attention mechanism, allowing the model to focus on different areas of the image.

Li et al. [20] noted that the previous attention mechanisms were mainly designed for ground-level images, and did not replicate precisely how the human attention mechanism actually works. The reason behind this is that humans focus

TABLE I
OVERVIEW OF PREVIOUS RESULTS FOR REMOTE SENSING IMAGE CAPTIONING WITH NEURAL METHODS.

	UCM				**Sydney**				**RSICD**			
Method	BLEU 1	BLEU 4	METEOR	CIDER	BLEU 1	BLEU 4	METEOR	CIDER	BLEU 1	BLEU 4	METEOR	CIDER
CSMLF [45]	0.4361	0.1210	0.1320	0.2227	0.5998	0.3433	0.2475	0.7555	0.5759	0.2217	0.2128	0.5297
RTRMN (statistical) [46]	0.8028	0.6393	0.4258	—	—	—	—	—	0.6102	0.2859	0.2751	—
Multi-Scale Cropping [57]	0.5940	0.4290	—	—	0.6150	0.4000	—	—	—	—	—	—
Soft-Attention [21]	0.7454	0.5250	0.3886	2.6124	0.7322	0.5820	0.3942	2.4993	0.6753	0.3617	0.3255	1.9643
Hard Attention [21]	0.8157	0.6182	0.4263	2.9947	0.7591	0.5258	0.3898	2.1819	0.6669	0.3407	0.3201	1.7925
FC Attention + LSTM [58]	0.8135	0.6352	0.4173	2.9958	0.8076	0.5544	0.4099	2.2033	0.7459	0.4574	0.3395	2.3664
SM Attention + LSTM [58]	0.8154	0.6458	0.4240	3.1864	0.8143	0.5806	0.4111	2.3021	0.7571	0.4612	0.3513	2.3563
Structured attention [60]	0.8538	0.7149	0.4632	3.3489	0.7795	0.5861	0.3954	2.3791	0.7016	0.3934	0.3291	1.7031
Cross-hierarchy attention [47]	0.823	0.659	—	3.192	0.817	0.591	—	2.291	0.770	0.471	—	2.363
SD-RSIC ResNet50 [38]	0.7430	0.5150	0.3580	—	0.7160	0.3980	0.3200	—	0.6490	0.2950	0.2490	—
SD-RSIC DenseNet169 [38]	0.7470	0.5180	0.3750	—	0.7300	0.4670	0.3410	—	0.6430	0.2850	0.2440	—
SAT (LAM-TL) [59]	0.8208	0.7229	0.4880	**3.7088**	0.7425	0.5369	0.3700	2.3563	0.6790	0.4148	0.3298	2.6672
Adaptive (LAM-TL) [59]	0.8570	**0.7430**	0.5100	3.758	0.7365	0.5348	0.3693	2.3513	0.6756	0.4077	0.3261	2.6285
ML Attention + Semantic [54]	0.8330	0.6623	0.4371	3.1684	0.8233	0.6003	0.4202	2.3110	0.7597	0.4623	0.3543	2.3614
Denoising-based fusion, [11]	0.8306	0.6345	—	3.2956	**0.8324**	0.5851	—	**3.8198**	—	—	—	—
VRTMM+SCST [35]	—	—	—	—	—	—	—	—	0.7934	0.5113	0.3726	2.7930
Multi-level attention [20]	**0.8864**	0.7271	**0.5222**	3.3074	0.7900	**0.6052**	**0.4741**	2.1811	**0.8058**	0.5163	**0.4718**	2.7716
Continuous Representations [30, 29]	0.8510	0.6666	0.4229	3.4239	—	—	—	—	0.7846	**0.5190**	0.3714	**2.7777**

not only on the image when they are describing it, but also on the description itself. The authors then proposed a multi-level attention model to try to simulate this mechanism. Additionally, the authors proposed an updated version of the publicly available datasets supporting research in the area (i.e., UCM, Sydney, and RSICD), reducing grammatical errors and inappropriate captions, and thus minimizing the predisposition to learn wrong descriptions. The proposed method was composed by 3 different attention structures that respectively represent attention to the image, to the description, and to the combination of vision and semantics. This method currently holds the state-of-the-art in several metrics on the UCM-*captions* and Sydney-*captions* datasets.

In addition to the introduction of novel attention mechanisms, there have also been some other interesting encoder-decoder approaches. Sumbul et al. [38] proposed applying a summarization technique to the ground-truth captions to remove possible redundancies, to later integrate with standard captions by an adaptive weighting strategy. Li et al. [19] came up with a novel truncation cross-entropy loss to prevent the networks from overfitting. Ramos and Martins [30, 29] presented a novel encoder-decoder architecture for remote sensing image captioning, based on continuous output representations. The authors argue that these representations can better capture the global semantic similarity between captions and images, enabling the use of loss functions that can go beyond token-level comparisons, and allow for comparisons at the sequence-level. The authors also point out some problems with the modified versions of the currently available datasets. This approach holds the current state-of-the-art results on most of the metrics on RSICD.

Table 1 summarizes the results obtained by different neural methods on the task, using the BLEU-1, BLEU-4, METEOR and CIDEr metrics.

III. METHODOLOGY

The structure of the proposed encoder-decoder method is shown in Figure 3. The image encoder corresponds to an EfficientNetV2-M model, and the decoder consists of an attention component, an LSTM, and an integrated Transformer model aiding in caption generation. A smaller module to retrieve similar images and their respective captions is also present.

A. Image Encoder

In a traditional encoder-decoder architecture for image captioning, the image encoder is responsible for mapping the image inputs into real-valued vector representations. In remote sensing image captioning, the image encoder is often a CNN model pre-trained on the ImageNet dataset [4], this dataset only contains ground-level images with very different characteristics from remote sensing imagery [20]. Furthermore, the ImageNet task entails categorizing the image's main object, whereas in remote sensing image captioning, many objects of interest must often be considered inside a single image. Inspired by previous state-of-the-art approaches in the area [20, 30], we propose to fine-tune a pre-trained state-of-the-art CNN model using remote sensing imagery.

Ramos and Martins [30, 29] had already reported on good results with an EfficientNet model and, in this work, we opted to use a more recent EfficientNetV2-M(21K) [41] model as our encoder, due to the state-of-the-art accuracy achieved on ImageNet and other transfer learning tasks [40], while being smaller and faster than previous alternatives (e.g., ResNet, DenseNet, InceptionNet, and the original EfficientNet). The EfficientNetV2 is an optimized version of the original EfficientNet, where several improvements were made to the original architecture. One of the major changes consisted on the usage of Fused-MBConvs [39], which replaces the original 1x1 convolution and 3x3 depth-wise convolution on the MBConv [40] with a traditional 3x3 convolution. Additionally, a novel progressive learning method was proposed to address the issue of large image sizes during training, by progressively changing image size and regularization during training, jointly optimizing training speed and accuracy. The specific model used in our experiments has around 55 million parameters and achieved a top-1 accuracy of 85.1% on the object classification task in ImageNet.

We concentrated our efforts on a contrastive fine-tuning task to make our encoder more suitable for remote sensing data. Specifically, the model is fine-tuned using a supervised contrastive loss [16], with the objective of maximizing the correct

prediction of classes for the images by the encoder, whilst also maximizing the agreement between image representations of the same class, and diverging the representations from different classes.

The contrastive loss function is inspired by work on self-supervised learning [3], in which visual representations for downstream tasks can be obtained by pushing similar images pairs to also have similar representations and dissimilar image pairs to stay far apart. Considering that we draw $2N$ augmentation pairs from a sample of N <image, class> pairs, the following equation corresponds to the loss:

$$\mathcal{L}_{out}^{sup} = \sum_{i \in I} \mathcal{L}_{out,i}^{sup} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau)}, \quad (1)$$

where τ represents a scalar temperature parameter [16, 3], $P(i)$ is the set of indices concerning positives instances (i.e., same class images as the anchor) distinct from the anchor instance i in the augmented samples batch, and $A(i)$ is the set of all indices in the multiviewed batch $2N$ also distinct from i. Also, $\boldsymbol{z}_i$ is the representation of the anchor (i.e., source sample), $\boldsymbol{z}_p$ is the representation of another positive sample, and the denominator represents the computation over the remaining terms (i.e., both positives and negatives), where the negatives refer to the images with a different class from the anchor i.

During the fine-tuning of the encoder component, we apply image augmentation twice on an input batch of data to get two different views of the batch, as in Khosla et al. [16]. Specifically, the augmentation methods consisted of geometric transformations such as vertical and horizontal flips, image rotations (90º, 180º and 270º), and color perturbations such as randomized histogram matching [52] against another image, or randomly changing the contrast, colour gamma and brightness of an image. Each of the previous methods had a 50 percent chance of being applied.

We also assessed the performance of the encoder on image classification. For that, we followed the same approach taken by Khosla et al. [16], training a linear classifier on top of the frozen representations using a cross-entropy loss, where each image is allocated to one of the scene classes associated to the images (e.g., 31 classes in RSICD).

In our image captioning model the encoder weights are fixed after the fine-tuning with in-domain data, while the complete captioning model is training.

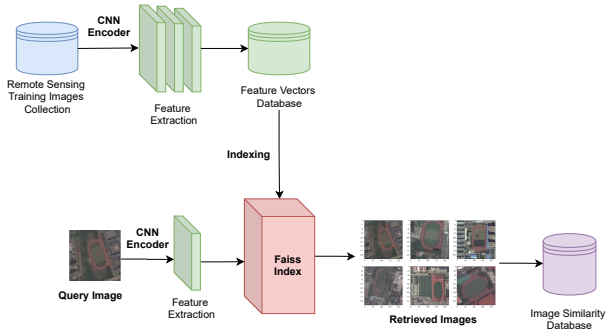


Fig. 2. Illustration for the image retrieval component.

B. Caption Retrieval

To retrieve similar images and their respective captions, we implemented an image retrieval component using our fine-tuned encoder as a feature extractor, and the Facebook AI Similarity Search (FAISS) library [13] to index the extracted features. The process is illustrated in Figure 2. Our image encoder's last convolutional layer retrieves the feature maps, which are then down sampled with an average pooling layer and flattened into a 1-D vector with a dimensionality equal to the number of output channels (i.e., 1280 for EfficientNetV2-M). In a second phase, we save the feature vectors in a FAISS index, without compression or quantization. Finally, we run a search on the index, comparing the distance between the indexed vectors and the query vector using the Euclidean (L2) distance. Because we only indexed the training feature vectors, when the query vector refers to a training image, we retrieve the second most similar image. Every search is pre-computed and the most similar images are saved in a collection for further access during the training (and testing) of the captioning model.

C. Text Decoder

As in a typical encoder-decoder framework for image captioning, the decoder's objective is to generate an accurate caption considering the provided image information (i.e., the visual feature vectors). In our specific architecture, we use not only the input image representation, but also the captions from the most similar image in the training data. The complete process involves three main parts: 1) attention-based LSTM decoding, 2) gathering of auxiliary information through a pre-trained Transformer, and 3) fusion of both methods.

1) Attention-Based LSTM Decoding:
Because generating a description for a remote sensing image can be seen as a sequential classification problem, we propose using a LSTM [9] as our language generation decoder, influenced by earlier image captioning work. Similarly to a vanilla RNN, the LSTM processes data sequentially. The difference lays on the operations within the LSTM cells, which allow the model to better deal with long sequences.

First, we initialize the initial hidden state $\mathbf{h}_0$ and cell state $\mathbf{c}_0$ for the decoder's LSTM, based on the encoded image $\mathbf{V}$. The image representations are linearly projected to the LSTM dimensionality using two separate layers. Then, the LSTM takes the concatenation of the representation for a start token $< \text{start} >$ and the attention-weighted image encoding $\mathbf{v}$ as the input $\mathbf{x}_0$. Two types of attention were tested with the decoder, namely standard soft attention and dual-attention in conjunction with pyramid feature maps [53].

Soft attention maximizes the marginal likelihood over all possible attention locations, meaning that every patch (i.e., every group of pixels) in the image is given some weight. We follow the approach proposed by Xu et al. [49], also using a form of regularization to prevent the decoder to focus too much on a certain area of the image, instead better distributing attention across all the areas. Dual Attention calculates attention weights not only in a spatial perspective (i.e., according to

width and height), but also takes into account the dependencies between feature maps, calculating a channel weight vector. Furthermore, we pair Dual Attention with Pyramid Attention as proposed by Yu et al. [53], by concatenating three different levels of feature maps generated with different pooling kernels (1×1, 2×2, 4×4). More formally, the spatial attention is computed as follows, where $\oplus$ represents the broadcast adding operator:

$$\mathbf{V} = [\mathbf{I_1}; \mathbf{I_2}; \mathbf{I_3}], \tag{2a}$$

$$\mathbf{v}_t^{(s)} = \frac{1}{L} \sum_{i=1}^{L} \alpha_i \mathbf{V}_i, \tag{2b}$$

$$\mathbf{a} = \tanh\left((\mathbf{V}\mathbf{W}_s + \mathbf{b}_s) \oplus \mathbf{h}_{t-1}\mathbf{W}_{hs}\right), \tag{2c}$$

$$\alpha = \mathrm{softmax}\left(\mathbf{a}\mathbf{W}_a + \mathbf{b}_a\right), \tag{2d}$$

In the previous expression, $\mathbf{V}$ represents the concatenation of the different feature maps, and $\mathbf{h}_{t-1}$ represents the previous LSTM hidden state. The spatial context vector $\mathbf{v}_t^{(s)}$ is computed by weighting each 2D image patch $\mathbf{V}_i$ (i.e. group of pixels) with a spatial attention vector $\alpha = [\alpha_1, \ldots, \alpha_L] \in \mathbb{R}^L$, where L represents the height and width dimensions multiplied ($L = w \times h$), the parameters $\mathbf{W}_s$, $\mathbf{W}_{hs}$, $\mathbf{W}_a$ are learnable weight matrices, and $\mathbf{b}_a$, $\mathbf{b}_s$ are bias vectors. This spatial attention vector α results from passing the spatial score vector $\mathbf{a}$ through a softmax function. Note that this score determines how important is a certain area of an image to the semantics of the sentence. However, as Yu et al. [53] observed, complementing this by learning channel-wise dependencies can be beneficial, due to the restricted setting of spatial attention. This is particularly true in complex remote sensing images. The channel context vector computation can be represented as follows, where $\odot$ represents an element-wise product:

$$\mathbf{v}_t^{(c)} = \beta \odot \frac{1}{L} \sum_{i=1}^{L} \mathbf{V}_i, \tag{3a}$$

$$\mathbf{c} = \tanh\left((\mathbf{W}_c\mathbf{V} + \mathbf{b}_c) \oplus \mathbf{W}_{hc}\mathbf{h}_{t-1}\right), \tag{3b}$$

$$\beta = \mathrm{sigmoid}\left(\mathbf{W}_b\mathbf{c} + \mathbf{b}_b\right). \tag{3c}$$

Notice that the channel attention score $\mathbf{c}$ and the channel weight vector β are computed similarly to the spatial attention, having as the only difference the fact that instead of a softmax function, the authors propose to use a sigmoid function, enabling multiple channel dimensions to be emphasized. The channel context vector $\mathbf{v}_t^{(c)}$ is then calculated by re-weighting the whole feature map $\mathbf{V}$ with the channel weight vector β. The final dual attention vector $\mathbf{v}_t$ results from the summation of both the spatial $\mathbf{v}_t^{(s)}$ and channel $\mathbf{v}_t^{(c)}$ context vector.

$$\mathbf{v}_t = \mathbf{v}_t^{(s)} + \mathbf{v}_t^{(c)}. \tag{4}$$

Model training uses a teacher forcing strategy in which the word embedding that is concatenated to the input is actually obtained from the token in the ground-truth caption. This process can be formulated as follows:

$$\mathbf{x}_t = [\mathbf{r_t}; \mathbf{v_t}], \tag{5a}$$

$$\mathbf{h}_t^{LSTM} = \mathrm{LSTM}\left([\mathbf{x}_t; \mathbf{h}_{t-1}]\right). \tag{5b}$$

The parameter $\mathbf{r_t}$ corresponds to the ground-truth word embedding, and $\mathbf{h}_{t-1}$ corresponds to the previous hidden state. Note that in this stage we do not yet perform word generation, so there is no need for a fully connected layer transforming the LSTM hidden state into a probability distribution over the token vocabulary.

2) Auxiliary Transformer Language Model:

The second component of our decoder consists of an auxiliary language model (AuxLM), inspired by the work of Kalimuthu et al. [14]. We specifically take the hidden states from a pre-trained Transformer model and fuse them with the LSTM states. We primarily employed the PEGASUS encoder-decoder model [55] in our implementation, while also performing experiments with the GPT-2 model [28].

The architecture of PEGASUS is similar to that of a typical Transformer encoder-decoder [42], but the authors considered model training with Gap Sentence Generation (GSG), i.e. a novel self-supervised pre-training task for abstractive text summarization. GSG involves masking the most important sentences from a document, which are determined using the ROUGE metric, and then training the model to predict the missing sentences. Additionally, some tokens from the remaining phrases are masked at random, similarly to Masked Language Modeling in the BERT pre-training strategy [5]. Both these pre-training tasks are combined during the training of the PEGASUS model, which achieves state-of-the-art results in a variety of summarization datasets.

In our decoder, we use PEGASUS for conditional generation with a language modeling head, extracting the last decoder hidden state $\mathbf{h}_t^{LM}$. The encoder is initialized using the captions from the most similar image in the training dataset (see the section on caption retrieval). Next, each token embedding in the input sequence is multiplied by three different weight matrices, generating three sets of vectors, the queries $\mathbf{Q}$, the keys $\mathbf{K}$ and the values $\mathbf{V}$. These vectors are passed through a self-attention layer, allowing for the encoder to attend over all positions of the input sequence. The keys and values from the final encoder output (i.e., the encoding and decoding components are composed of N self-attention layers) are fed to the self-attention layer in the decoder, allowing it to attend over all positions in the input sequence. Additionally, the decoder is auto-regressive, in the sense that it attends over past positions in the output sequence (i.e., by masking future positions).

Finally, the last hidden state generated by the model, $\mathbf{h}_t^{LM}$, is collected for further concatenation with $\mathbf{h}_t^{LSTM}$.

3) Fusion-Based Caption Generation:

The fusion module combines the hidden states $\mathbf{h}^{LM}$ and $\mathbf{h}^{LSTM}$, using the combination to inform the caption generation. Our implementation is based on work from Kalimuthu et al. [14], which described three types of model fusion: Simple (SF), Cold (CF), and Hierarchical (HF) fusion. We did not use the HF strategy because it did not increase overall performance, despite additional complexity.

The SF technique is the simpler fusion mechanism, consisting of a concatenation between the hidden states, followed by a linear projection with a ReLU activation function.

$$\mathbf{h}_t^F = \mathrm{ReLU}\left(\mathbf{W}\left[\mathbf{h}_t^{LSTM}; \mathbf{h}_t^{LM}\right] + \mathbf{b}\right). \tag{6}$$

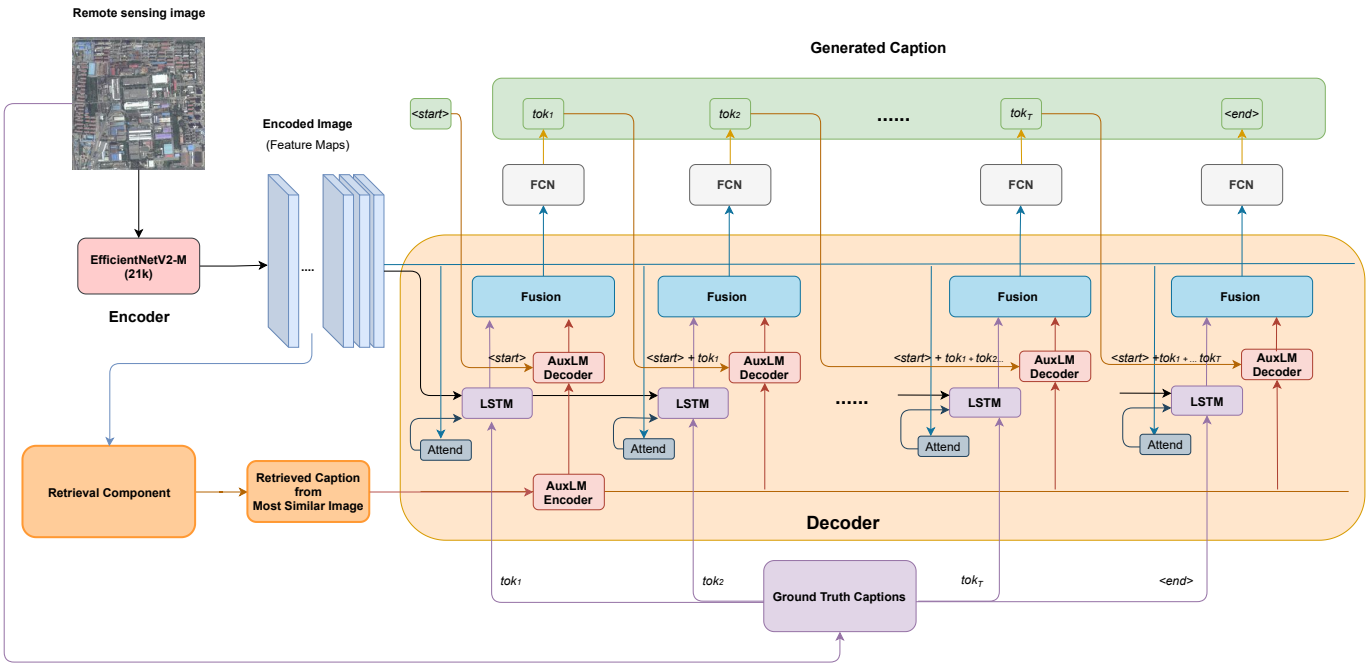


Fig. 3. Illustration for the proposed encoder-decoder architecture for image captioning, including the retrieval module.

The CF approach is slightly more complex, using a fine-grained gating mechanism to control how much information should be shared between the captioning model and the AuxLM. The method is formally described next, where $\odot$ represents an element-wise product, and the remaining parameters are similar to those of the SF method.

$$\mathbf{c}_t^{LM} = \text{ReLU}\left(\mathbf{W}\left[\mathbf{h}_t^{LM}\right] + \mathbf{b}\right), \tag{7a}$$

$$\mathbf{g}_t = \text{ReLU}\left(\mathbf{W}\left[\mathbf{h}_t^{LSTM}; \mathbf{c}_t^{LM}\right] + \mathbf{b}\right), \tag{7b}$$

$$\mathbf{h}_t^{CF} = \left[\mathbf{h}_t^{LSTM}; \left(\mathbf{g}_t \odot \mathbf{c}_t^{LM}\right)\right], \tag{7c}$$

$$\mathbf{h}_t^{F} = \text{ReLU}\left(\mathbf{W}\left[\mathbf{h}_t^{CF}\right] + \mathbf{b}\right). \tag{7d}$$

The final output $\mathbf{h}_t^F$ is processed through a fully connected layer with dropout, to generate scores over the full vocabulary (i.e., the logits vector). The word with the highest score is selected as the output for that particular time-step:

$$p(\mathbf{y}_t \mid \mathbf{h}_t^F) = \text{softmax}\left(\mathbf{W}\mathbf{h}_t^F + \mathbf{b}\right), \tag{8}$$

where $\mathbf{W}$ represents a weight parameter, and $p(\mathbf{y}_t \mid \mathbf{h}_t^F)$ the conditional distribution over the vocabulary. Finally, given a target ground truth sequence $(\mathbf{y}_1^*, ..., \mathbf{y}_T^*)$, the captioning model is trained by minimizing the cross-entropy loss:

$$L(\theta) = -\sum_{t=1}^{T} \log\left(p_\theta\left(\mathbf{y}_t^* \mid \mathbf{y}_1^*, ..., \mathbf{y}_{t-1}^*\right)\right), \tag{9}$$

where θ denotes the trainable parameters of the model.

IV. Experiments

This section describes the experimental validation of the approach described in the previous section.

A. Datasets

Our experiments have mostly relied on the largest dataset used in remote sensing image captioning studies, namely the Remote Sensing Image Captioning Dataset (RSICD), originally proposed by Lu et al. [21]. A total of 10,921 images were associated to captions generated by human volunteers with annotation experience in the field, where each volunteer provided one or two sentences for an image, to promote diversity. The annotation process was constrained through a set of instructions to try to minimize the number of inappropriate captions and poor descriptions of the images (e.g., sentences have to contain a minimum of 6 words). Each image has three channels (RGB) and a size of 224 $\times$ 224 pixels although the images feature different ground-level resolutions and correspond to different types of scenes. The image data was collected using sources such as Google Earth, Baidu Map, MapABC, and Tianditu. The total number of sentences was 24,333 in a first phase but, in order to extend the dataset due to the fact that only 724 images were described using 5 sentences, the authors randomly duplicated the sentences in the remaining set of images, which resulted in a final number of 54,605 sentences. A detailed view of the scene classes in the dataset is represented on Table II.

Besides RSICD, there are also two smaller datasets that have been used in previous work, namely UCM-*captions* and Sydney-*captions*. The UCM-*captions* dataset was proposed by Qu et al. [27], extending the original UC Merced Land Use Dataset [50] by adding five detailed sentences to describe the content of each image. The dataset has a total of 2,100 RGB images, each with a size of 256 $\times$ 256 pixels. The image data was extracted from the United States Geological Survey (USGS) National Map Urban Area Imagery collection. There are a total of 10,500 descriptions and 21 classes, with 100

TABLE II
NUMBER OF IMAGES OF EACH CLASS IN THE RSICD DATASET.

Class	Number	Class	Number	Class	Number	Class	Number	Class	Number
Airport	420	Church	240	Industrial	390	Park	350	Railway Station	260
Bare Land	310	Commercial	350	Meadow	280	Mountain	340	River	410
Baseball Field	276	Dense Residential	410	Medium Residential	290	Square	330	Sparse Residential	300
Beach	400	Desert	300	School	300	Pond	420	Stadium	290
Bridge	459	Farmland	370	Parking	390	Viaduct	420	Resort	290
Center	260	Forest	250	Playground/Playfields	370/661	Port	389	Storage Tanks	396

images each. The Sydney-*captions* dataset was also originally proposed by Qu et al. [27], based on the Sydney Dataset from Fan et al. [7]. This dataset is considerably smaller, containing only 613 images. Each image has a size of 500 $\times$ 500 pixels, and is described with 5 sentences. The images belong to one of 7 different scene classes.

Note that we use the modified versions of the 3 datasets as proposed by Li et al. [20], given that these authors fixed a multitude of problems encountered in the original versions, such as misspellings of words, grammatical errors, punctuation errors, and singular/plural errors, among others. Despite the fact that the previous work improved the quality of the original datasets, recent work [29] has revealed additional problems in the RSICD dataset. First, RSICD is still far smaller and less diversified than popular natural image captioning datasets (e.g., COCO-captions has 330,000 images). When comparing the number of terms in RSICD with Flickr8K, the authors discovered that the latter had a vocabulary three times larger than RSICD, although having only 8000 images. Additionally, the amount of repeated captions is highly concerning, suggesting that a large number of images are described equally (e.g., the sentence "many buildings and green trees are in a dense residential area" is repeated over 500 times). Furthermore, the authors pointed out that the vocabulary used on the validation split differed significantly from that used on the train and test split, which might lead to substantial issues during training (e.g., defining the early-stopping criteria). We expect that a method such as the one proposed here can be beneficial for scenarios involving less diversified captions, although we also believe that future work in the area should address the aforementioned limitations, e.g through the proposal of other benchmark datasets.

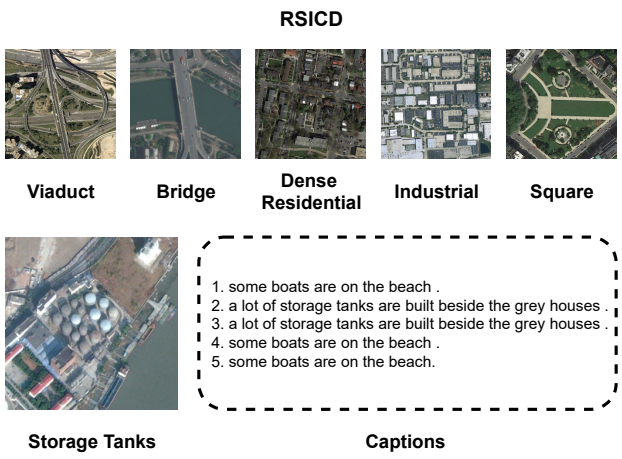


Fig. 4. Examples of different RSICD images with their corresponding classes. Below we have an example of an image, together with its captions.

B. Evaluation Metrics

In regard to evaluating model performance, we used traditional *n*-gram overlap metrics for assessing text generation, such as BLEU, METEOR, CIDEr, ROUGE-L, and the content overlap metric named SPICE. Additionally, we experimented with more recent BERT-based evaluation metrics, namely, BLEURT and BERTScore.

The BLEU (Bilingual Evaluation Understudy) metric, introduced by Papineni et al. [25], calculates the *n*-gram precision of the generated sentences. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) measures recall instead of precision and, more specifically, we opted to use ROUGE-L which employs the largest common subsequence of *n*-grams for matching. METEOR (Metric for Evaluation of Translation with Explicit ORdering) times is based on the harmonic mean of the unigram precision and recall as defined by Banerjee and Lavie [2]. Furthermore, this metric added flexibility to word matching, by taking into account morphological variants, synonyms, and word stems, in addition to exact word matches. CIDEr (Consensus-based Image Description Evaluation), introduced by Vedantam et al. [43], leverages concepts like Term-Frequency Inverse-Document-Frequency (TF-IDF) to measure the similarity between the generated sentence and the multiple reference captions. This metric was deemed in a study by Reiter [32] as the most robust and most correlated with human assessments, from all of the untrained automatic metrics mentioned previously.

Besides the aforementioned *n*-gram overlap metrics, we also used the SPICE (Semantic Propositional Image Caption Evaluation) metric introduced by Anderson et al. [1], which uses scene graphs to extract semantic similarities between generated and reference captions.

Finally, we also looked at some of the more recently proposed criteria for evaluating Natural Language Generation (NLG) tasks deciding on the use of BERTScore [56] and BLEURT [34]. BERTScore was proved to have a high correlation with human assessments, relying on the alignment of contextual of word embeddings from BERT. These embeddings are matched with candidate and reference sequences by cosine similarity. BLEURT is also a BERT-based evaluation metric, although it requires a training step that in a first phase

consists of fine-tuning a checkpoint from BERT on synthetic data using automatic metric scores and, in a second phase, uses fine-tuning on system-generated outputs.

C. Experimental Details

This section provides additional details on the experimental protocol, describing the data pre-processing and relevant training hyper-parameters.

1) Data Pre-processing:
Before feeding the images to the encoder model, they are resized to a dimensionality of 224×224 pixels, and normalized using the mean and standard deviation of ImageNet's RGB channels [4]. Furthermore, when performing fine-tuning with in-domain data, the images are also subject to different augmentations (see the methodology section).

In terms of the text, we maintain captions under 30 characters long and use the whole vocabulary for decoding, as early trials revealed that limiting the maximum length of a caption or the vocabulary to frequent tokens did not result in improvements. The raw captions are tokenized in an identical way to how the Transformer AuxLM model was trained.

2) Training details:
The EfficientNetV2-M(21k)[1] encoder model [41] was fine-tuned through a supervised contrastive approach, as described previously we follow the same approach taken in Khosla et al. [16], also propagating the different views of the same image through a projection layer. The batch size is set to 64 and we use the Adam optimizer with an initial learning rate of $1e^{-4}$, halting the training when the loss does not decrease for 5 epochs. All of the encoder's parameters are then fixed to be used in the subsequent experiments.

The fine-tuned encoder is used to extract the feature vectors in the caption retrieval module. A FAISS[2] index is set up with the dimensionality of the extracted vectors (1280 for EfficientNetV2-M). Finally, the L2 (Euclidean) distance is employed to retrieve the vectors of related images during the search. To speed up the training and conserve computing resources, the similar images are pre-computed and used during model training and evaluation.

In the decoder, the LSTM hidden state dimensionality is set to 512, similarly to the embedding layer and the attention context vector. The final hidden state dimensionality of the fusion architecture may vary depending on the type of fusion: in simple fusion, the final hidden state has a dimensionality equal to the sum of the hidden states of the AuxLM and the LSTM (e.g., given the final hidden state for the PEGASUS model we have a dimensionality of [1024 + 512]). For cold fusion, the AuxLM hidden state is projected through a linear layer to a dimensionality equal to that of the LSTM hidden state size, before vector concatenation, resulting in a final hidden state with a dimensionality of [512 + 512]. The final hidden state is followed by a dropout layer with probability 0.5, and then a fully connected layer.

When training the encoder-decoder architecture, the batch size is set to 32 and the Adam optimizer is used with an initial learning rate of $1e^{-4}$, only adjusting the weights of the decoder. Additionally, to speed up the training, we employ a teacher forcing strategy, meaning that the ground-truth is supplied at each time-step to the LSTM and AuxLM decoder. On the same note, we also apply dynamic batching (i.e., we process only the effective batches without padding). Finally, we implement a learning rate decay strategy, multiplying the learning rate by 0.8 if the validation BLEU-4 does not improve after 2 epochs, and stopping the training if the same occurs for 6 epochs.

In all our experiments, instead of using greedy decoding (i.e., choosing the word with the highest score) at inference time, we apply beam search to find the optimal sequence. At the first decode step, the beam search algorithm chooses top-k word candidates and then for each of these candidates, it predicts the top-k second candidates, forming a sequence. Next, the top-k generated sequences (i.e., [tok_1, tok_2]) are selected, considering their additive scores. This process is repeated at each decoding step. Finally, after the top-k sequences are complete, the one with the best overall score is chosen. In our experiments, we set the beam-width parameter k to 5.

All of our models and experiments were implemented using PyTorch[3] and the source code is publicly available[4].

D. Experimental Results

In the following subsections, we first validated the effectiveness of the proposed encoders, by reporting the classification accuracy on RSICD, and assessing the performance of the retrieval component. Then, we describe the results obtained with our captioning model, both in a quantitative and a qualitative way.

1) Assessing the Image Encoder:
Our first tests assessed the quality of the fine-tuned image encoder on the RSICD dataset, measuring its performance in terms of accuracy when classifying the images according to the type of scene. We compared the EfficientNetV2 model against a EfficientNet-B5 encoder similar to the one used in the work of Ramos and Martins [30, 29]. We considered two different settings: (i) fine-tuning with a classification task that uses a standard cross-entropy loss, (ii) fine-tuning in a supervised contrastive way as described previously, followed by training a classification layer with the cross-entropy loss. The tests were performed using images from RSICD, classified in 31 different scene classes.

As expected, the recently proposed EfficientNetV2 model outperformed the EfficientNet-B5 model in both settings. The

TABLE III
COMPARISON OF DIFFERENT APPROACHES FOR CLASSIFYING RSCID IMAGES.

Method \ Loss	Cross-Entropy (CE)	Supervised Contrastive (SupCon)
EfficientNet-B5	92.31	94.60
EfficientNetV2-M(21K)	94.76	95.88

[1]https://github.com/rwightman/pytorch-image-models
[2]https://github.com/facebookresearch/faiss
[3]https://pytorch.org/
[4]https://github.com/Jbarata98/remote-sensing-image-captioning

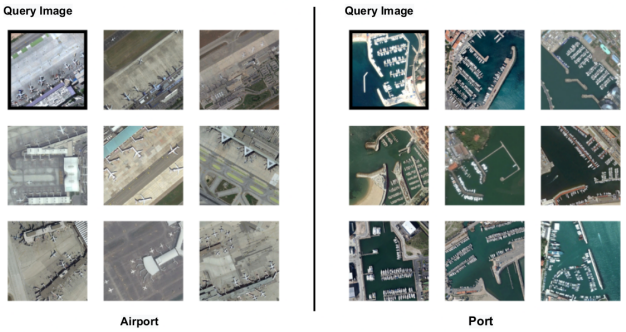


Fig. 5. Two examples of query images from RSICD ,and their respective most similar pairs. The most similar is the one to the right of the query image (i.e. the order of similarity is given by the rows from left to right.)

supervised contrastive loss, which we chose because of its potential benefits in image retrieval scenarios, also outperformed cross-entropy in the classification task. The accuracy results are presented in Table III.

To evaluate the performance of the retrieval module, leveraging the EfficientNetV2 representations, we used the label of the retrieved most similar image and compared it with the label of the query image. This allowed us to understand if the extracted features and the index itself were working properly, as images that share the same class are typically similar. The retrieval module had a near perfect score, with a precision of 0.9903. Out of the 31 classes in RSICD, it had a perfect score for 10 of the classes (i.e. baseball field, bridge, farmland, forest, meadow, parking, port, railway station, river and stadium).

2) Comparing Methods for Caption Generation: In this sub-section, our objective is to understand if integrating pre-trained Transformers in an encoder-decoder architecture is advantageous for the generation of remote sensing image captions. We performed a series of tests by altering various components in our architecture (see Figure 3).

***Quantitative Analysis*)** We started implementing a baseline model consisting of an EfficientNetV2-M(21k) encoder and a LSTM decoder with soft attention. Represented in Table IV, the results were already higher than those from several different previous studies, and we argue that this happens mainly due to the state-of-the-art encoder used in our architecture. We also noted that fine-tuning our encoder with domain data is beneficial for the model's performance, outperforming our first baseline (i.e., directly using the EfficientNetV2 model pretrained on ImageNet) by 0.0406 on the BLEU-4 metric.

Next, we implement the dual attention mechanism with pyramid feature maps. This type of attention slightly improved the performance of our architecture. As expected, the usage of multi-scale feature representations enriches the attention component. Most notably, on the RSICD dataset, this approach already achieved state-of-the-art scores on the SPICE metric. This architecture was chosen as the starting point for further experiments with auxiliary language models.

The initial tests with pre-trained Transformers were performed with the GPT-2 model [28], as the AuxLM. The GPT-2 model is used only as an auto-regressive decoder (i.e. without encoding the caption from the most similar images), where the AuxLM hidden state, with size equal to 768, is fused with the hidden state of the LSTM. Our results show that despite having competitive results when compared with previous studies, a decoder-only AuxLM without global context (i.e., the decoder only looks at previous positions to generate the next token), slightly decreases the performance. With this in mind, we integrated the pre-trained encoder-decoder PEGASUS Transformer model, as the AuxLM.

In most metrics, using the PEGASUS model as the AuxLM outperformed all previous models, reaching state-of-the-art results in BLEU-2, BLEU-3, BLEU-4, and CIDEr, as well as very competitive results on the remaining metrics.

We also tested our models on the smaller UCM-*captions* and Sydney-*captions* datasets, the results are represented in Tables VI and V. Again, we find that our base models perform quite well, achieving state-of-the-art results in some metrics (e.g. SPICE on UCM). In terms of the results with integrated pre-trained Transformers our models present very competitive results, specifically on the metrics that better correlate with human judgements (i.e., SPICE and CIDEr). We also note that in these smaller datasets, a simpler fusion scheme with fewer parameters seems to increase the overall performance.

***Qualitative Analysis*)** We start by showing an illustration of the results obtained with spatial attention on pyramid feature maps (Figure 6). The visualization was done by extracting the weight vector and overlaying the attention weights. The highlighted areas signify where the attention mechanism focuses on, when predicting a word. The model is able to detect several distinct components on a highly complex image of a viaduct. We argue that this is one of the benefits of using multi-scale feature representations when dealing with remote sensing images, as the original aerial view makes it difficult to distinguish the different objects of interest in the image.

In Table VIII, we present captioning results obtained with different architectures. Our pre-trained models are able to describe the most significant semantic information in the images, with the exception of the baseline which is more likely to generate inaccurate information. When we look at some of the instances more closely, we can see that the non fine-tuned

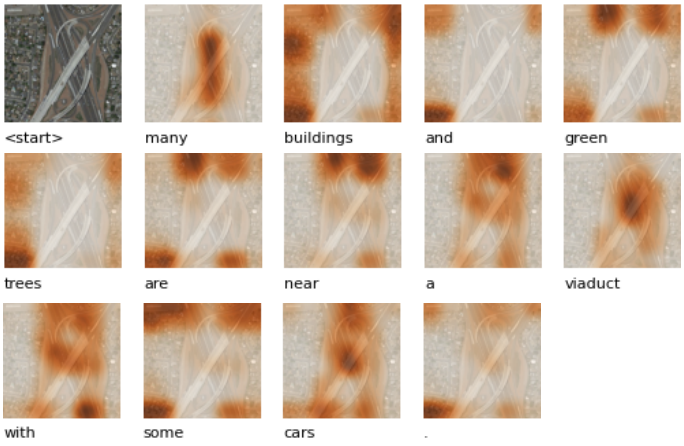


Fig. 6. Illustration of the results obtained with pyramid feature maps.

TABLE IV
COMPARISON OF OUR MODELS AND THE PREVIOUS STATE-OF-THE-ART ON RSICD. THE BEST RESULTS SHOWN IN BOLD.

Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE_L	CIDEr	SPICE	BERTScore	BLEURT
Baseline	0.7479	0.6270	0.5334	0.4581	0.3571	0.6443	2.4881	0.4660	0.6663	0.3561
Baseline (fine-tuned)	0.7758	0.6649	0.5733	0.4987	0.3875	0.6876	2.7946	0.5046	0.7052	0.4223
Dual Attention (no fusion)	0.7818	0.6728	0.5823	0.5067	0.3856	0.6898	2.7985	**0.5108**	0.7054	**0.4243**
GPT2 (SF)	0.7591	0.6464	0.5568	0.4842	0.3693	0.6657	2.6298	0.4807	0.6670	0.3707
PEGASUS (SF)	0.7947	0.6860	0.5955	0.5198	0.3775	0.6827	2.8441	0.4941	0.7008	0.4079
PEGASUS (CF)	0.8003	**0.6974**	**0.6126**	**0.5408**	0.3838	0.6931	**2.9316**	0.4998	**0.7099**	0.4202
Li et al. [20]	**0.8058**	0.6778	0.5866	0.5163	**0.4718**	**0.7247**	2.7716	0.4786	—	—
Ramos and Martins [30, 29]	0.7846	0.6794	0.5915	0.5190	0.3820	0.6810	2.7777	0.4913	0.7073	—

TABLE V
COMPARISON OF OUR MODELS AND THE PREVIOUS STATE-OF-THE-ART ON UCM-*captions*. THE BEST RESULTS SHOWN IN BOLD.

Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE_L	CIDEr	SPICE	BERTScore	BLEURT
Baseline (fine-tuned)	0.8521	0.8032	0.7605	0.7218	0.4645	0.8065	3.4001	0.4880	0.7916	0.4623
Dual Attention (no fusion)	0.8421	0.7879	0.7403	0.6955	0.4625	0.8109	3.4218	0.5018	0.7935	0.4965
GPT2 (SF)	0.8402	0.7826	0.7334	0.6861	0.4345	0.7822	3.2936	0.4557	0.7518	0.4317
PEGASUS (SF)	0.8555	0.8016	0.7530	0.7075	0.4624	0.8158	**3.5288**	**0.5184**	**0.8011**	**0.4960**
PEGASUS (CF)	0.8495	0.7930	0.7469	0.7070	0.4535	0.8065	3.4547	0.5021	0.7956	0.4651
Li et al. [20]	**0.8864**	**0.8233**	**0.7735**	**0.7271**	**0.5222**	**0.8441**	3.4239	0.5021	—	—
Ramos and Martins [30, 29]	0.8510	0.7810	0.7226	0.6666	0.4444	0.8026	3.4239	0.4857	0.8059	—

TABLE VI
COMPARISON OF OUR MODELS AND THE PREVIOUS STATE-OF-THE-ART ON SYDNEY-*captions*. THE BEST RESULTS SHOWN IN BOLD.

Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE_L	CIDEr	SPICE	BERTScore	BLEURT
Baseline (fine-tuned)	0.7348	0.6549	0.5865	0.5247	0.3701	0.6750	2.2770	0.4232	0.6554	0.2792
Dual Attention (no fusion)	0.7496	0.6667	0.5967	0.5331	0.3731	0.6803	2.2048	0.4147	0.6608	0.2781
GPT2 (SF)	0.7562	0.6772	0.6056	0.5417	0.3674	0.6933	2.3752	0.4319	0.6611	0.2648
PEGASUS (SF)	0.7762	0.6896	0.6088	0.5377	0.3747	0.7116	**2.4286**	0.4412	**0.6831**	0.2878
PEGASUS (CF)	0.7630	0.6761	0.5970	0.5284	0.3757	0.7007	2.2534	**0.4570**	0.6760	**0.3012**
Li et al. [20]	**0.7900**	**0.7108**	**0.6517**	**0.6052**	**0.4741**	**0.7352**	2.1810	0.4089	—	—

baseline incorrectly associates storage tanks with buildings, trees with cars, or a small group of trees with a park, whereas, we find that, in most cases, the same does not happen with the pre-trained models.

Comparing the two attentions, we find that the dual attention mechanism is often superior when dealing with images of higher complexity, including images from classes like dense residential, viaducts, airports, or even storage tanks, where it is able to capture their different dimensions. Regarding the models with Transformers as an AuxLM, using PEGASUS outperforms the use of GPT2 in terms of correctness. Despite the fact that captions generated with our PEGASUS model generated captions are the more similar to the ground-truth in every case, the semantic diversity in the RSICD dataset, as described in Li et al. [20], is not very high, implying that the model is reliant on the vocabulary and phrases from the training split.

We also examine the retrieval and generation outputs in detail. In Table VII we display examples of retrieved images and generated captions using the PEGASUS model. The retrieved information, as expected, can have a favorable impact on the correctness of the sentences, as depicted on the airport example. However, we also find that the retrieved captions can have a slightly negative impact when guiding the model's prediction whenever the retrieved image is significantly different from the query image, as shown in the baseball field example. In this case, the model fails to capture both baseball fields, mentioning only one. Despite the fact that our retrieval component is practically flawless when it comes to retrieving images from the same classes, the intra-class caption variability can be a noise introducing factor. However, we argue that this does not have a significant negative impact on the quality of the generated captions, as the current remote sensing image captioning datasets are very templated. With the exception of minor factors like omitting adverbs (for instance, in the example from the church image, the PEGASUS models omits the *several* before *green trees*, which could have been affected by the retrieved captions), or incorrectly mentioning the number of baseball fields or runways in an airport, the auxiliary information given by same class images will most often be beneficial for the generation process.

E. Conclusions and Future Work

We presented a novel remote sensing image captioning method, integrating pre-trained transformers in a traditional encoder-decoder architecture. We explored a novel attention mechanism and the fine-tuning of a state-of-the-art encoder on domain data, althought we mostly focused on a fusion

TABLE VII
EXAMPLES OF RETRIEVED IMAGES AND GENERATED CAPTIONS WITH THE PEGASUS MODEL ON RSICD.

Query Image	Retrieved Image	Retrieved Captions	Generated Captions
		**1)** "many planes are parked in an airport." **2)** "there are many different planes in the open airport." **3)** "there are many different planes in the open airport." **4)** "many planes are parked in an airport." **5)** "many planes are parked in an airport."	• **Ground-Truth**: "some planes are parked in an airport" • **PEGASUS (SF)**: "many planes are in an airport" • **PEGASUS (CF)**: "many planes are parked near a terminal in an airport"
		**1)** "this area is a large baseball field." **2)** "a lot of trees are planted around the baseball field." **3)** "a lot of trees are planted around the baseball field." **4)** "this area is a large baseball field." **5)** "this area is a large baseball field."	• **Ground-Truth**: "some buildings and green trees are near two baseball fields" • **PEGASUS (SF)**: "some green trees are around a baseball field" • **PEGASUS (CF)**: "a baseball field is surrounded by some green trees and buildings"
		**1)** "some buildings and green trees are around a church" **2)** "some buildings and green trees are around a church." **3)** "some buildings and green trees are around a church." **4)** "some buildings and green trees are around a church." **5)** "some buildings and green trees are around a church."	• **Ground-Truth**: "some buildings and several green trees are around a church" • **PEGASUS (SF)**: "some green trees are near a church" • **PEGASUS (CF)**: "some buildings and green trees are around a church"

approach between the representations of a common LSTM decoder and a pre-trained Transformer decoder. The experimental results reveal that our method is effective, especially when combined with PEGASUS as an auxiliary language model leveraging captions from similar images, outperforming the current state-of-the-art in various metrics.

The main conclusions drawn from the work presented in this article are the following: 1) fine-tuning the encoder on domain data is important, due to the different characteristics between aerial and ground-level images; 2) multi-scale feature representations are beneficial when dealing with complex remote sensing images; 3) integrating large pre-trained Transformers on encoder-decoder architectures is a very effective and simple way of leveraging captions from similar images to boost the performance of the captioning task.

Auxiliary language models can be easily integrated in different encoder-decoder architectures from the previous literature. Thus, despite the interesting results, we propose as a future experiment to pair our method with attention mechanisms from previous work in remote sensing image captioning (e.g., multi-level attention [20], structured attention [60] or instance-aware attention [47]). Moreover, althought we only used fixed auxiliary language models for preserving computer resources, one could also fine-tune the AuxLM model on domain data, similarly to what was done in the encoder, or adjust the parameters during regular model training.

REFERENCES

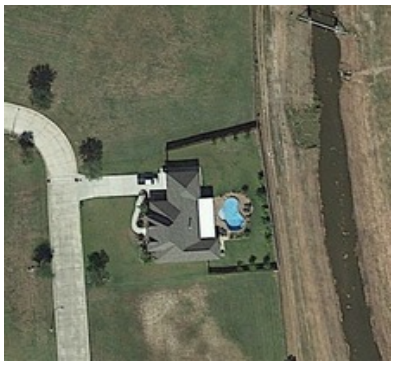
[1] P. Anderson, B. Fernando, M. Johnson, and S. Gould. Spice: Semantic propositional image caption evaluation. *CoRR*, abs/1607.08822, 2016.

[2] S. Banerjee and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005.

[3] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020.

[4] J. Deng, R. Socher, L. Fei-Fei, W. Dong, K. Li, and L.-J. Li. ImageNet: A large-scale hierarchical image database. In *2009 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

TABLE VIII
EXAMPLES OF GENERATED CAPTIONS WITH OUR DIFFERENT MODELS ON RSICD.

Storage Tanks	Generated Captions
	• **Ground-Truth**: "several storage tanks are near a river" • **Baseline**: "many storage tanks are near several buildings" • **Soft Attention**: "some storage tanks are near a river" • **Dual Attention**: "many storage tanks in different sizes are near a river" • **GPT2 (SF)**: "several storage tanks are surrounded by bare land" • **PEGASUS (SF)**: "some storage tanks are near a river" • **PEGASUS (CF)**: "several storage tanks are near a river"
Dense Residential	
	• **Ground-Truth**: "many buildings and green trees are in a dense residential area " • **Baseline**: "many buildings are in a park with many green trees and a pond" • **Soft Attention**: "many buildings are in a park with many green trees and a pond" • **Dual Attention**: "many buildings and some green trees are in a dense residential area" • **GPT2 (SF)**: "many buildings and green trees are in a dense residential area" • **PEGASUS (SF)**: "many buildings and green trees are in a dense residential area" • **PEGASUS (CF)**: "many buildings and green trees are in a dense residential area"
Sparse Residential	
	• **Ground-Truth**: "there is a building with a swimming pool in the green land villa" • **Baseline**: "a building is near a road with several cars" • **Soft Attention**: "a building with a swimming pool is surrounded by some sparse green trees" • **Dual Attention**: "many green trees and meadows are around a building with a swimming pool" • **GPT2 (SF)**: "a building with swimming pool is near a piece of green meadow." • **PEGASUS (SF)**: "many green trees and several buildings are around a swimming pool" • **PEGASUS (CF)**: "a building with a swimming pool is surrounded by some green trees and meadows"
Square	
	• **Ground-Truth**: "many green trees and a building are around a circle square" • **Baseline**: "many green trees are around a square square" • **Soft Attention**: "many green trees are around a square square" • **Dual Attention**: "many green trees are around a circle square" • **GPT2 (SF)**: "many green trees are around a circle square with many green trees." • **PEGASUS (SF)**: "many green trees are around a circle square" • **PEGASUS (CF)**: "many green trees are around a circle square"

[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[6] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. *CoRR*, abs/1411.4389, 2014.

[7] Fan, B. Du, Zhang, and L. Zhang. Saliency-guided unsupervised feature learning for scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 53(4), 2014.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[9] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8), 1997.

[10] I. Hrga and M. Ivašić-Kos. Deep image captioning: An overview. In *2019 In Proceedings of the International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2019.

[11] W. Huang, Q. Wang, and X. Li. Denoising-based multiscale feature fusion for remote sensing image captioning. *IEEE Geoscience and Remote Sensing Letters*, 18(3), 2020.

[12] M. Ivašić-Kos, M. Pobar, and S. Ribaric. Two-tier image annotation model based on a multi-label classifier and fuzzy-knowledge representation scheme. *Pattern Recognition*, 52, 11 2015.

[13] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus. *CoRR*, abs/1702.08734, 2017.

[14] M. Kalimuthu, A. Mogadala, M. Mosbach, and D. Klakow. Fusion models for improved visual captioning. *CoRR*, abs/2010.15251, 2020.

[15] A. Karpathy and F. Li. Deep visual-semantic alignments for generating image descriptions. *CoRR*, abs/1412.2306, 2014. URL http://arxiv.org/abs/1412.2306.

[16] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan. Supervised contrastive learning. *CoRR*, abs/2004.11362, 2020.

[17] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12), 2013.

[18] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *Proceedings of the Conference on Computational Natural Language Learning*, 2011.

[19] X. Li, X. Zhang, W. Huang, and Q. Wang. Truncation cross entropy loss for remote sensing image captioning. *IEEE Transactions on Geoscience and Remote Sensing*, 59(6), 2021.

[20] Y. Li, S. Fang, L. Jiao, R. Liu, and R. Shang. A multilevel attention model for remote sensing image captions. *Remote Sensing*, 12(6), 2020.

[21] X. Lu, B. Wang, X. Zheng, and X. Li. Exploring models and data for remote sensing image caption generation. *IEEE Transactions on Geoscience and Remote Sensing*, 56(4), 2017.

[22] X. Ma, R. Zhao, and Z. Shi. Multiscale methods for optical remote-sensing image captioning. *IEEE Geoscience and Remote Sensing Letters*, PP:1–5, 07 2020. doi: 10.1109/LGRS.2020.3009243.

[23] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *Computer Vision and Pattern Recognition*, 2015.

[24] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems*, 2011.

[25] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2002.

[26] M. Pobar and M. Ivašić-Kos. Multimodal image retrieval based on keywords and low-level image features. In *Revised Selected Papers of the COST Action IC1302 International KEYSTONE Conference on Semantic Keyword-Based Search on Structured Data Sources*, IKC 2015, 2015.

[27] B. Qu, X. Li, D. Tao, and X. Lu. Deep semantic understanding of high resolution remote sensing image. In *Proceedings of the IEEE the International Conference on Computer, Information and Telecommunication Systems*, 2016.

[28] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. 2018.

[29] R. Ramos and B. Martins. Using neural encoder-decoder models with continuous outputs for remote sensing image captioning. Unpublished Manuscript, 2021.

[30] R. Ramos and B. Martins. Remote sensing image captioning with continuous output neural models. In *Proceedings of the SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2021.

[31] R. P. Ramos, P. Pereira, H. Moniz, J. P. Carvalho, and B. Martins. Retrieval augmentation for deep neural networks. In *International Joint Conference on Neural Networks*, 2021.

[32] E. Reiter. A structured review of the validity of BLEU. *Computational Linguistics*, 44(3), 2018.

[33] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.

[34] T. Sellam, D. Das, and A. P. Parikh. BLEURT: Learning robust metrics for text generation. *CoRR*, abs/2004.04696, 2020.

[35] X. Shen, B. Liu, Z. Yong, J. Zhao, and M. Liu. Remote sensing image captioning via variational autoencoder and reinforcement learning. *Knowledge-Based Systems*, 203, 04 2020.

[36] Z. Shi and Z. Zou. Can a machine generate humanlike language descriptions for a remote sensing image? *IEEE*

Transactions on Geoscience and Remote Sensing, 55(6), 2017.

[37] M. Stefanini, M. Cornia, L. Baraldi, S. Cascianelli, G. Fiameni, and R. Cucchiara. From show to tell: A survey on image captioning. *CoRR*, abs/2107.06912, 2021.

[38] G. Sumbul, S. Nayak, and B. Demir. SD-RSIC: Summarization driven deep remote sensing image captioning. *CoRR*, abs/2006.08432, 2020.

[39] M. Tan and S. Gupta. Efficientnet-EdgeTPU: Creating accelerator-optimized neural networks with automl, Aug. 2019.

[40] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *Proceedings of the International Conference on Machine Learning*, 2019.

[41] M. Tan and Q. V. Le. Efficientnetv2: Smaller models and faster training. *CoRR*, abs/2104.00298, 2021.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[43] R. Vedantam, C. L. Zitnick, and D. Parikh. CIDEr: Consensus-based image description evaluation. *CoRR*, abs/1411.5726, 2014.

[44] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge. *CoRR*, abs/1609.06647, 2016.

[45] B. Wang, X. Lu, X. Zheng, and X. Li. Semantic descriptions of high-resolution remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 16(8), 2019.

[46] B. Wang, X. Zheng, B. Qu, and X. Lu. Retrieval topic recurrent memory network for remote sensing image captioning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 2020.

[47] C. Wang, Z. Jiang, and Y. Yuan. Instance-aware remote sensing image captioning with cross-hierarchy attention. *CoRR*, abs/2105.04996, 2021.

[48] X. Wang, Y. Wu, Y. Ming, and H. Lv. Remote Sensing Imagery Super Resolution Based on Adaptive Multi-Scale Feature Fusion Network. *Sensors*, 20:1142, 2020.

[49] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. *CoRR*, abs/1502.03044, 2015.

[50] Y. Yang and S. Newsam. Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2010.

[51] Y. Yang, C. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2011.

[52] C. Yaris, B. Huang, K. Bradbury, and J. M. Malof. Randomized histogram matching: A simple augmentation for unsupervised domain adaptation in overhead imagery. *CoRR*, abs/2104.14032, 2021.

[53] L. Yu, J. Zhang, and Q. Wu. Dual attention on pyramid feature maps for image captioning. *CoRR*, abs/2011.01385, 2020.

[54] Z. Yuan, X. Li, and Q. Wang. Exploring multi-level attention and semantic relationship for remote sensing image captioning. *IEEE Access*, 8, 2019.

[55] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. *CoRR*, abs/1912.08777, 2019.

[56] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. BERTScore: Evaluating text generation with BERT. *CoRR*, abs/1904.09675, 2020.

[57] X. Zhang, Q. Wang, S. Chen, and X. Li. Multi-scale cropping mechanism for remote sensing image captioning. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, 2019.

[58] X. Zhang, X. Wang, X. Tang, H. Zhou, and C. Li. Description generation for remote sensing images using attribute attention mechanism. *Remote Sensing*, 11(6), 2019.

[59] Z. Zhang, W. Diao, W. Zhang, M. Yan, X. Gao, and X. Sun. LAM: Remote sensing image captioning with label-attention mechanism. *Remote Sensing*, 11(20), 2019.

[60] R. Zhao, Z. Shi, and Z. Zou. High-resolution remote sensing image captioning based on structured attention. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–14, 2021.