الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Département : Génie électrique Et Informatique Industrielle

Mémoire de Master en Télécommunication
Spécialité
Systèmes de Télécommunications et Réseaux

Performance Evaluation Of An Automated Image Description Routine Considering Different Datasets

Réalisé par

Meriem BENYOUCEF

Présenté et soutenu publiquement le : 06/07/2022

Composition du Jury:

Présidente	Mme. Souhila BOUTARFA,	M.C.B	ENST
Promotrice	Mme. Seloua CHOUAF,	M.A.A	ENST
Examinatrice	Mme Samira BOUCHAMA,	M.C.B	ENST
Examinateur	M. Belkacem ABBADI,	M.C.B	ENST

Année Universitaire 2021 – 2022

Performance Evaluation Of An Automated Image Description Routine Considering Different Datasets

BENYOUCEF Meriem, CHOUAF Seloua
Génie électrique Et Informatique Industrielle GEII
Ecole Nationale Supérieure de Technologies ENST
Alger, Algérie
Email: m_benyoucef@enst.dz , seloua.chouaf@enst.dz

***Abstract*— The process of generating an accurate textual description based on a given image utilizing deep learning-based architectures has been applied in many fields. In this paper, we chose to develop and compare the performances of multimodal CNN and LSTM based architectures applied for the remote sensing task. First, we aimed to evaluate our architectures on the RSCID (Remote Sensing Image Captioning Dataset) in order to compare our results with what was achieved in literature. Second, we studied the performances of our architectures on a remote sensing dataset of our creation. Our various experiments demonstrated the efficiency of different CNN and LSTM multimodal architectures in describing remote sensing images accurately scoring satisfactory results in terms of natural language metrics.**

Keywords- Remote sensing, Convolutional Neural Networks CNN, Recurrent Neural Networks RNN, Long Short Term MemoryLSTM, deep learning, Natural Language Processing NLP

I. INTRODUCTION

Remote sensing technology has shown remarkable evolution over the years, it has been applied in several tasks such as scene classification and object recognition, these applications remain simple and don't take into account all the subjects present in the image and the relationship between them. Image description has shown a potential of compressing information of an image in a concise textual description that is grammatically and semantically correct. Applying automated image description methods in remote sensing tasks can be of a great help in image retrieval, scene classification and more. Many deep learning-based architectures have been designed for image description purposes, and our goal is to study the performance of four CNN (Convolutional Neural Networks) and LSTM (Long Short Term Memory) multimodal architectures when applied for remote sensing image description task. As a first step, we will present the general architecture of our architectures, then we will present our experiments and the datasets used for each study. First, we will study the performance of the different architectures on a standard remote sensing dataset which is the RSCID (Remote Sensing Image Captioning Dataset), the architectures are evaluated using natural language metrics namely: BLEU [1], ROUGE [2], METEOR [3] and CIDEr [4] and the scores are compared to the ones achieved in literature. Second, we proceed to create a new remote sensing dataset, the performances of the different architectures are evaluated in terms of time consumption, loss and accuracy, and the generated descriptions are evaluated using the previously mentioned language evaluation metrics. For each experiment, we present descriptions generated by each model based on sample images. The paper will be concluded by a general conclusion and future work.

II. METHODOLOGY

There are many architectures created for image description purposes, we chose to carry our studies using an encoder-decoder based architecture

- For the encoder part, we will be using a CNN (Convolutional Neural Network) architecture that reads the image input and encodes it into a fixed-length vector. The feature vector is obtained by deleting the last two fully connected layers destined for image classification applications.
- For the decoder part, we will be using LSTM (Long Short Term Memory) recurrent networks that reads the encoded vector image and outputs a textual description.

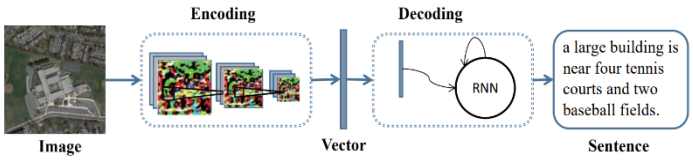


Figure 1. Outline of an encoder-decoder architecture[5]

We will be using the merge architecture presented by Marc Tanti, et al. in their papers [6-7] to combine the CNN and LSTM architectures since it has proven to be efficient in image description tasks.

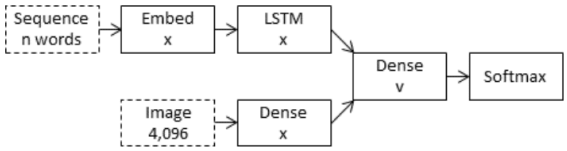


Figure 2. Encoder-decoder merge architecture [6-7]

The proposed model is trained with a set of images and their corresponding sentence descriptions called “the training set”.

The CNN learn the relation between the words of the provided descriptions and the image subjects by extracting features from the images and encoding them into a fixed-length vector. The RNN is fed the feature vector and the description sentences to learn which feature best represents what part of text. The previous outputs of an LSTM cell will form the input of the next until the end of the sequence.

Repeating this procedure on every image of the training set, the model learns how to associate image subjects with their proper text description and improve its textual description.

In our different experiments, we chose to study four CNN architectures of encoders which are: VGG-16[8], Inception-V3 [9], ResNet-50[10] and Xception [11], each combined with an LSTM based decoder part.

III. EXPERIMENTAL PROCEDURE

The experimental procedure will be divided into two parts, the first part concerns the study of the different multimodal architectures on an existing remote sensing dataset which is the RSCID (Remote Sensing Image Captioning Dataset) and the obtained results will be compared to literature. The second part consists of evaluating these architectures on a new remote sensing dataset of our creation.

A. Experiment 1: performance evaluation on a standard dataset for comparison with works cited in the literature

In this experiment, we seek to evaluate our architectures’ performances when applied for remote sensing purposes and compare our results with what is present in the literature. The comparison will be in terms of loss, accuracy but especially in terms of language evaluation metrics.

1) Dataset presentation

The Remote Sensing Image Captioning Dataset (RSICD) [12] is the largest dataset that has been made public in terms of remote sensing image captioning task. It contains a total of 10,921 images including thirty types of scenes namely: airport, industrial, parking, park, school, mountain, viaduct and more. These images were collected from Google Earth, Baidu Map, MapABC and Tianditu. All the images of the dataset have a fixed size of 224x224 pixels which display various resolutions, and each image is provided with five descriptive sentences. Figure 1 (a) shows an overview on the dataset’s images. Figure 3 (b) display two images and their five descriptive sentences.



Figure 3. Overview of the RSCID dataset

2) Results – presentation & discussion

Due to the important size of the RSICD dataset, we limited our experiments to three themes that we picked up among the thirty themes available in the original dataset. Each theme has been considered as an independent dataset.

As it is shown in figure 1 (b), scenes which present various elements are described with abundant and more detailed sentences.

Based on these two considerations, we choose to carry our experiments on the following themes:

- Airport theme
- Industrial theme
- Park and parking themes combined

Each subset has been divided into a training set, a testing set and a validation set, following the dataset’s original splitting as it is mentioned in the table below.

TABLE I. TRAINING, TESTING AND VALIDATION SETS DISTRIBUTION FOR EACH SUBSET

Subset	Subset size	training set size	testing set size	validation set size
Airport	360	276	42	42
Industrial	390	312	39	39
Park and Parking	740	592	74	74

For all the forthcoming experiments, the following settings were adopted.

TABLE II. TRAINING SETTINGS

Optimizer	**Adam**
Batch size	32 (default value)
Learning rate	0,001 (default value)
Loss function	Categorical cross entropy
Metrics	Accuracy
Maximum description length	32 characters

The four models were trained until convergence on each of the three subsets independently, the results are than averaged over the three subsets for each architecture. The advantage of Google collab's cloud service is used to lunch the training of all four models simultaneously.
Results of the four architectures will be presented and discussed below.

a) Time consumption

The table below presents an estimation of the time each architecture consumed to complete the training, the best results will be marked in bold.

TABLE III. TIME CONSUMPTION RESULTS OF THE FOUR ARCHITECTURES AFTER BEING TRAINED ON EACH SUBSET

Subset	**Model**	**Number of epochs**	**Time per epoch**	**Total time spent on training**
Airport	Inception-V3	300	Around 1.5 min	About 8 h
	ResNet-50	550	Around 3 min	About 28 h
	VGG-16	200	Around 1.5 min	About 5 h
	Xception	**120**	**Around 1 min**	**About 2 h**
Industrial	Inception-V3	300	Around 1.5 min	About 8 h
	ResNet-50	200	Around 1.5 min	About 5 h
	VGG-16	250	Around 1.5 min	About 6 h
	Xception	**150**	**Around 1 min**	**About 2.5 h**
Park And Parking	Inception-V3	250	Around 1.5 min	About 7 h
	ResNet-50	200	Around 1.5 min	About 5 h
	VGG-16	200	Around 1.5 min	About 5 h
	Xception	**160**	**Around 1 min**	**About 4.5 h**

The table above shows that the different models have different convergence speed, as a result, the time they consume to complete the training varies from two hours to twenty eight hours, taking between one and three minutes per epoch. The Xception model consumes the shortest period of time to reach its' convergence on each subset compared to the other architectures. Due the slow convergence speed of ResNet-50 architecture when trained on the airport subset, we chose to stop the training after five hundred epochs, and since small improvement was achieved from the two hundredth epoch, we chose to stop the training after two hundred epochs on the industrial and park and parking subsets.

b) Loss and accuracy

The figures below display average loss and accuracy values achieved by the four architectures after being trained on the three previously mentioned subsets independently.

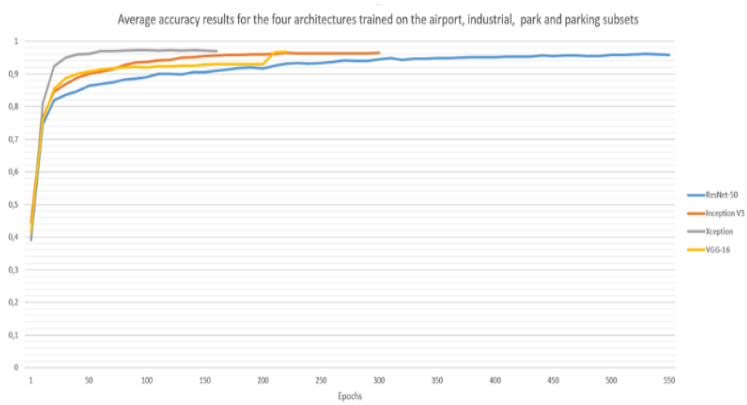


Figure 4. Average Loss results for the four architectures trained on the airport, industrial, park and parking subsets

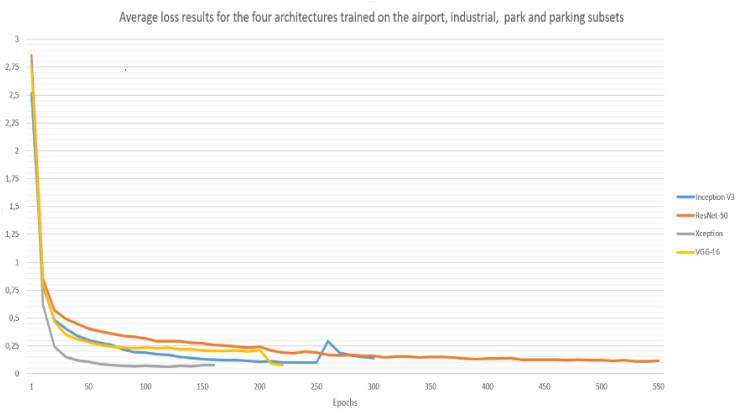


Figure 5. Average accuracy results for the four architectures trained on the airport, industrial, park and parking subsets

The table below includes the best results achieved by each model in terms of loss and accuracy, the best results will be marked in bold.

TABLE IV. BEST LOSS AND ACCURACY RESULTS OF THE FOUR MODELS AFTER BEING TRAINED ON THE THREE SUBSETS

subset	Model	Epoch of best results	Loss	Accuracy
Airport	Inception-V3	257	0.0795	96.95
	ResNet-50	517	0.1134	96.00
	VGG-16	149	0.0668	97.35
	Xception	**119**	**0.0551**	**97.57**
Industrial	Inception-V3	287	0, 1037	0, 9639
	ResNet-50	197	0, 2150	0, 9270
	VGG-16	191	0, 0697	0, 9711
	Xception	**127**	**0, 0517**	**0, 9757**
Park and parking	Inception-V3	243	0, 1103	0, 9650
	ResNet-50	199	0, 2150	0, 9270
	VGG-16	194	0, 0697	0, 9711
	Xception	**132**	**0, 0517**	**0, 9757**

Graphs above show loss and accuracy values of Inception-V3, Xception, VGG-16 and ResNet-50 models trained on the three subsets until they reach convergence. The results of the three subsets are then averaged for each architecture.

A fast improvement rate in the average of loss and accuracy values for the four models from the beginning of training until around the twentieth epoch. Following that a slower improvement rate until loss and accuracy values start fluctuating around stable values. Xception and VGG-16 loss and accuracy values start stabilizing from around the fiftieth epoch, from the hundred and fiftieth epoch for Inception-V3 model and starting from the three hundredth epoch for the ResNet-50 model which presents the slowest improvement in accuracy and loss results.

The estimation of loss and accuracy revealed results that have been improved along with the training. The best results have been collected at different epochs fluctuating between 0.0551 and 0.1134 in terms of loss and between 96.00 and 97.57 in terms of accuracy. In the light of these results, we could rank models' performance in terms of loss and accuracy decreasingly as it follows: Xception, VGG-16, Inception-V3 and finally ResNet-50. Generated descriptions.

The figures below present examples of descriptions generated by the four models based on the same image. The quantitative evaluation is provided by the natural language evaluation metrics. The evaluation in this case is qualitative, it is based on the similarities between the reference and generated descriptions and how much the generated descriptions are faithful to image. The correctly predicted words are marked in bold.

In most cases, the four models could identify the main elements of the pictures that the references included.

The models produced satisfactory results and human-like descriptions. However, they confused words that describe quantity such as many and some, words that describe position such as next to and near and use words with the same meaning such as many and several.

Image	*Generated descriptions*
	Inception V3 : many of the planes were docked on the wide
	ResNet-50 : many planes are parked in **an airport near** many **runways**
	VGG-16 : four planes are parked **in an airport near** several **buildings** with parking lots
	Xception : **several** planes are in **an airport near** some **buildings** and green trees
Reference descriptions	
we can see a simple terminal building and an apron connected with runways	
some building with a parking lot are near an airport with several runways	
a brown ground divided by the grey runway	
the tarmac and airport runways divide the field into several orderly arranged rounded rectangles next to which is buildings and a road	
the tarmac and airport runways divide the field into several orderly arranged rounded rectangles next to which is buildings and a road	

Image	*Generated descriptions*
	Inception V3 : some **trees are** planted **in the park**
	ResNet-50 : **many green trees** are planted **in the park**
	VGG-16 : many buildings are around **park** with **many green trees and pond**
	Xception : **many green trees** and some buildings **are in park**
Reference descriptions	
the park's facilities are complete	
there are many cars on the road near the park	
many green trees and a pond are in a park	
many green trees and a pond are in a park	
the park's facilities are complete	

Image	*Generated descriptions*
	Inception V3 : many **cars are** neatly arranged **in parking lot**
	ResNet-50 : **cars** and trucks **are parked in the parking lot**
	VGG-16 : many **cars are in parking lot** near several green trees and building
	Xception : lot of **cars parked in the parking lot**
Reference descriptions	
The parking lot is in a highway	
The car in the parking lot scattered distribution	
Some cars are parked in a parking lot	
Some cars are parked in a parking lot	
The parking lost is a highway	

Image	*Generated descriptions*
	Inception V3 : several **planes** are parked **in** an airport with large parking lot
	ResNet-50 : several **planes are near** terminal in an airport
	VGG-16 : some **planes are parked near** an airport with parking lot
	Xception : several **planes are in** an airport with some **green trees**
Reference descriptions	
Many planes are parked in runways near many green trees	
Many planes are parked in runways near many green trees	
Many planes are parked in runways near many green trees	
Many planes are parked in runways near many green trees	
Many planes are parked in runways near many green trees	

TABLE V. NATURAL LANGUAGE EVALUATION SCORES OF THE FOUR MODELS

	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr
Xception	**0,479539**	**0,31248833**	**0,226318**	**0,16880467**	0,13060867	**0,24253607**	**0,474552**
Inception-V3	0,39627967	0,21910567	0,13064667	0,08634133	0,09170067	0,17988767	0,269533
ResNet-50	0,43055833	0,25911967	0,167316	0,10832533	**0,14090443**	0,21044167	0,300814
VGG-16	0,43197967	0,263316	0,183465	0,13240133	0,11581867	0,22047663	0,362772
Average	0,43458917	0,26350742	0,17693642	0,12396817	0,11975811	0,21333551	0,3519177

c) Natural language evaluation

Table V and figure 6 display average language metrics scores achieved by the four models.

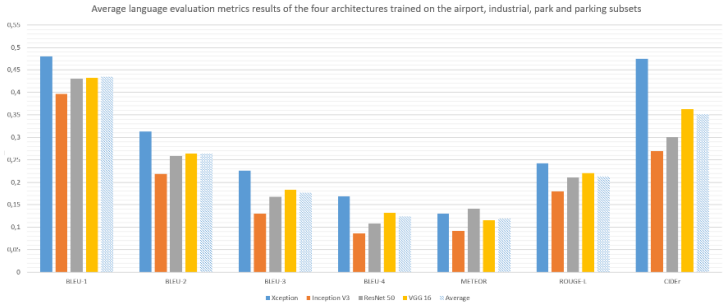


Figure 6. Average language evaluation metrics results for the four models trained on the airport, industrial and park and parking subsets

Xception scored the best result on all metrics except the METEOR score, followed by VGG-16, than ResNet-50 and finally Inception-V3 model. The average results of each metric are acceptable and demonstrate that CNN architectures are efficient in the remote sensing task.

d) Comparison with literature results

In this study [12], a multimodal architecture based on VGG-16 architecture for the feature extraction part and LSTM for the textual generation part was trained on different ratios of training sets of the RSCID dataset, dedicating 10% of the dataset validation and use the remaining portion as a test set. The figure below displays language metrics results for each proportion of training samples.

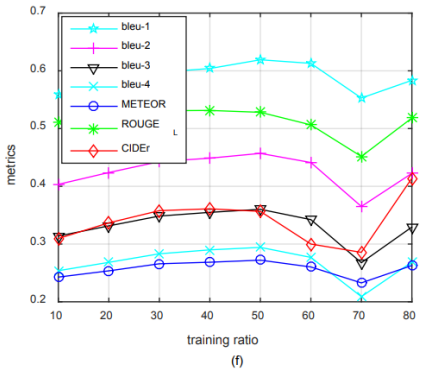


Figure 7. Metrics of multimodal method using LSTM on RSICD dataset

For our study, the total number of images we dedicated for training is 1180 images, which represents about 10% of the entire dataset.

The table below summarizes the best natural language evaluation metrics results achieved in the framework of our study.

TABLE VI. BEST LANGUAGE EVALUATION METRICS RESULTS ACHIEVED BY XCEPTION MODEL

BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE_L	CIDEr
0,54304	0,36650	0,28569	0,22137	0,165495	0,30743	**0,65231**

Comparing literature results with the best scores achieved in our study, we see that the Xception model achieved higher CIDEr and BLEU-1 than what was achieved in literature, as for the other metrics, our results remain close to literature. The scores can be influenced by many factors, namely the presence very similar images with almost the same descriptions which results in a reduced vocabulary size and enabling the models to learn the differences between the subjects.

B. Experiment 2: performance evaluation on a new remote sensing dataset

To further study the efficiency of these multimodal architectures, we chose to create a new remote sensing dataset in order to train and test the different architectures.

1) Dataset presentation

In our previous experiments, we carried our work of developing and comparing image descriptions architectures on already annotated datasets available online. For this experiment we proposed to create a new dataset by exploiting 50 cropped images extracted from a dataset that has been initially created for change detection purposes and never been used for remote sensing, the dataset [13] was chosen for its availability and for the simple scenes that its images display which offers the possibility of description without major difficulties. We provided each image with three textual descriptions of our own using a provided software.



Figure 8. An overview on the created dataset

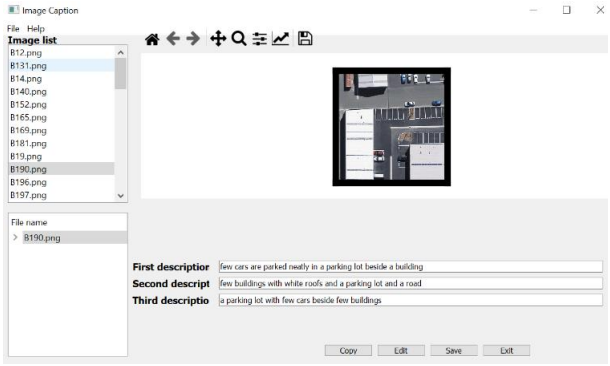


Figure 9. An overview on the provided software interface

2) Results – presentation and discussion

The dataset has been divided into a training set, a testing set and a validation set as it is mentioned in the table below.

TABLE VII. TRAINING, TESING AND VALIDATION SET DISTRIBUTION

Dataset size	Training set size	Testing set size	validation set size
50	40 (80%)	5 (10%)	5 (10%)

The results of the different architectures will be presented and discussed below.

a) Time consumption

The table below presents an estimation of the time each architecture consumed to complete the training, the best results will be marked in bold.

TABLE VIII. TIME CONSUMPTION OF THE FOUR MODELS TRAINED ON THE CREATED DATASET

Model	Total number of epochs	Time per epoch	Total time spent on training
Inception-V3	200	Around 3 sec	About 13 min
ResNet-50	200	Around 3 sec	About 13 min
VGG-16	200	Around 4 sec	About 13 min
Xception	200	Around 4 sec	About 13 min

The four architectures presented similar time consumption results, each architecture was trained for two hundred epochs, and spent around four seconds per epoch and a total of thirteen minutes to complete the training.

b) Loss and accuracy

The figures below show the evolution of loss and accuracy results.

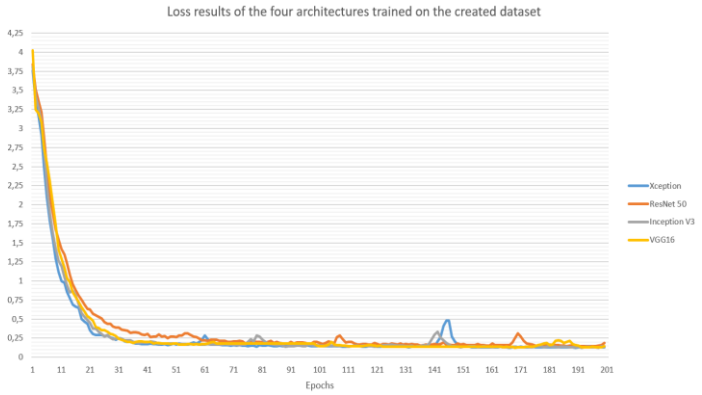


Figure 10. Loss results for the four architectures trained on the created dataset

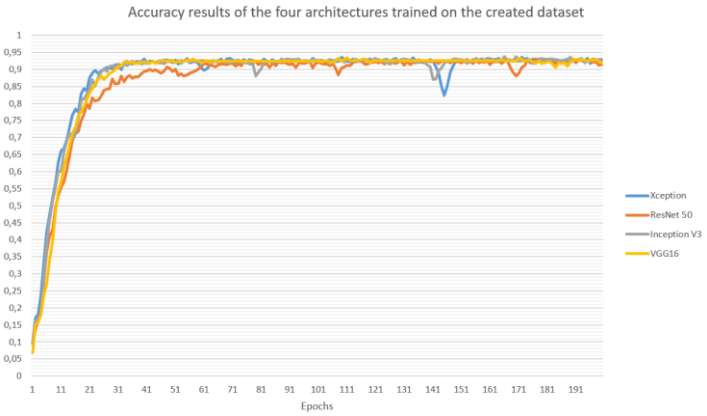


Figure 11. Loss results for the four architectures trained on the created dataset

The table below summarizes average loss and accuracy results achieved by the four architectures, the best results amongst the four models will be marked in bold.

TABLE IX. AVERAGE ACCURACY AND LOSS FOUR THE ARCHITECTURES TRAINED ON THE CREATED DATASET

Model	Epoch of best results	Loss	Accuracy
Inception-V3	169	0.1333	**0.9378**
ResNet-50	179	0.1422	0.9312
VGG-16	171	**0.1328**	0.9321
Xception	**132**	0.1378	0.9340
Average		0.1365	0.9337

Graphs above show loss and accuracy values of Inception-V3, Xception, VGG-16 and ResNet-50 models trained on the created dataset until they reach convergence.

The four models show a fast improvement rate in loss and accuracy values until they reach a stable value by the fiftieth epoch and fluctuate around this value until the end of the training process.
The four models achieved very close results in terms of loss and accuracy, the best loss value was achieved by VGG-16 followed by Inception-V3 than Xception and finally ResNet-50, the best accuracy value was achieved by Inception-V3 followed by Xception then VGG-16 and finally ResNet-50.

c) Generated descriptions

The figures below show descriptions generated by the four models based on the same images. The common words between the generated and reference descriptions are marked in bold.

Image	*Generated descriptions*
	**Inception V3 : few** trees and cars parked **in paved area beside building**
	ResNet-50 : houses **with few** cars **and** trees in the backyard
	VGG-16 : few houses **with trees in** their backyards **beside** road
	Xception : two **buildings** and road surrounded by bareland **and few trees**
Reference descriptions	
A building with a white roof beside a parking lot	
A parking lot in a paved area with few trees and a building	
A paved area beside a building and few trees	

Figure 12. Example 1 of generated descriptions by each model after being trained on the created dataset

Image	*Generated descriptions*
	**Inception V3 : few houses with trees in the backyard**
	ResNet-50 : few trees and building surrounded by bareland
	VGG-16 : road beside building with **few trees and** parking lot
	Xception : few houses with trees in paved area
Reference descriptions	
Few houses with trees and cars parked in their backyard	
A car in a road beside few houses with few trees in the backyard	
A car in a road beside few houses with cars in the backyard	

Figure 13. Example 2 of generated descriptions by each model after being trained on the created dataset

- Inception-V3 model could identify three subjects in the first example which are the trees, the building and the paved area and produced a sentence with all words correctly predicted.
- The other models only identified two subjects in each example, making the Inception-V3 descriptions the best amongst the four.
- The models confuse similar assets such houses and buildings, paved areas and roads. Nevertheless, the models produced acceptable descriptions.

d) Natural language evaluation

The diagram below display the natural language evaluation metrics results of the four architectures trained on the created dataset.

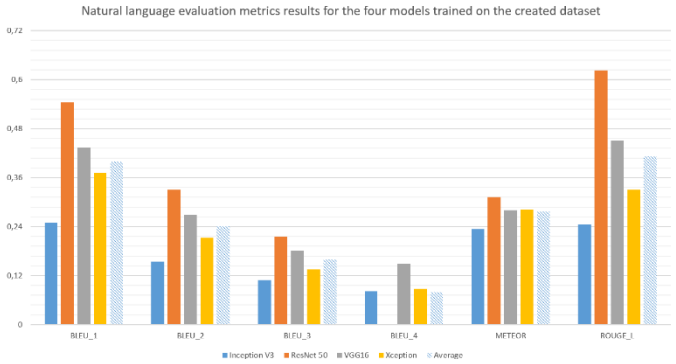


Figure 14. Natural language evaluation metrics results for the four models trained on the created dataset

The table below summarizes the natural language evaluation metrics results of the four models after being trained on the created dataset, the best results will be marked in bold.

TABLE X. AVERAGE NATURAL LANGUAGE METRICS RESULTS FOR THE FOUR ARCHITECTURES TRAINED ON THE CREATED DATASET

Model	**BLEU_1**	**BLEU_2**	**BLEU_3**	**BLEU_4**	**METEOR**	**ROUGE_L**
Inception-V3	0,25	0,154303	0,109772	0,081487	0,234242	0,245975
ResNet-50	**0,54386**	**0,329806**	**0,214639**	0,000026	**0,312177**	**0,622163**
VGG-16	0,43333	0,268742	0,181899	**0,14965**	0,28007	0,44986
Xception	0,37179	0,212288	0,135787	0,087868	0,28128	0,330124

Reducing the size of the dataset resulted in an improvement of metrics scores which can be explained by providing the model with lesser information to learn in the smaller dataset. Nevertheless, the results of the four models are satisfactory. ResNet-50 scored the best results in all the metrics except the BLEU-4 score, followed by VGG-16, than Xception and finally Inception-V3 model.

IV. CONCLUSION AND FUTURE WORK

In the first experiment, we have used the RSCID dataset to train and evaluate four multimodal architectures based on convolutional neural networks for the feature extraction part and LSTM architecture for the textual description part, the CNN architectures used are: Xception, Inception-V3, VGG-16 and ResNet-50 in order to test their performances when applied for remote sensing tasks. The different architectures were evaluated in terms of time consumption, loss and accuracy. Furthermore, the generated descriptions were evaluated using the BLEU-n, METEOR, ROUGE-L and CIDEr metrics, the scores were then compared to literature results. The achieved results were satisfactory compared to literature and could be improved by using larger training samples.

In the second experiment, we chose to create our own dataset by exploiting 50 images picked from a change detection dataset we attributed each picture with three descriptions, the four CNN and LSTM based architectures were trained and evaluated in terms of time consumption, loss, accuracy and natural language metrics. Finally, we concluded each experiment with descriptions generated by the four models.

The experimental results show that the image description methods based on CNN and LSTM architectures are efficient for remote sensing task.

For the next step, attention methods can be implemented to increase the robustness of our architectures, we propose to study attention architectures which use the same architectures like encoder-decoder but passes extra information to the decoder by using attention mechanism.

REFERENCES

[1] Papineni K., Roukos S., Ward T., and Zhu WJ. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, Association for Computational Linguistics, pages 311–318.

[2] Lin CY. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In Text Summarization Branches Out, Barcelona, Spain. Association for Computational Linguistics, pages 74–81.

[3] Banerjee S. and Lavie L. 2005. 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, Ann Arbor, Michigan. Association for Computational Linguistics, pages 65–72.

[4] Vedantam R., Zitnick C.L., and Parikh D. 2015. Cider: Consensus-based image description evaluation. In CVPR, pages 4566–4575.Lu, X.; Wang, B.; Zheng, X.; Li, X. . 2017, Exploring models and data for remote sensing image caption generation. IEEE Trans. Geosci. Remote Sens56, 2183–2195

[5] Tanti M., Gatt A. and Camilleri K. 2018. Where to put the image in an image caption generator. Natural Language Engineering.

[6] Tanti M., Gatt A. and Camilleri K. 2017. What is the Role of Recurrent Neural Networks (RNNs) in an Image Caption Generator? In Proceedings of the 10th International Conference on Natural Language Generation, Santiago de Compostela, Spain, pages 51-60.

[7] Simonyan K. and Zisserman A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, 7-9 May 2015, pages 1-14.

[8] Szegedy C., Vanhoucke V., Ioffe S., Shlens J., and Wojna Z. 2016. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2818-2826.

[9] Kaiming H., Zhang X., Ren S., and Sun J. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770-778.

[10] Chollet F. 2017. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition. Pages 1251-1258.

[11] Lu X., Wang B., Zheng X., and Li X. 2018. Exploring Models and Data for Remote Sensing Image Caption Generation. IEEE Transactions on Geoscience and Remote Sensing, 56, 2183-2195.

[12] Ji S., Wei S., and Lu M. 2019. Fully Convolutional Networks for Multisource Building Extraction From an Open Aerial and Satellite Imagery Data Set. IEEE Transactions on Geoscience and Remote Sensing, 57, 574-586.