Application of U-Net and Auto-Encoder to the Road/Non-road Classification of Aerial Imagery in Urban Environments

Amanda Spolti[1], Vitor C. Guizilini[3], Caio C. T. Mendes[4], Matheus D. Croce[5], André R. de Geus[1], Henrique C. Oliveira[2], André R. Backes[1] and Jefferson R. Souza[1]

[1]*School of Computer Science, Federal University of Uberlândia, Uberlândia, MG, Brazil*
[2]*Faculty of Civil Engineering, Architecture and Urbanism, State University of Campinas, Campinas, SP, Brazil*
[3]*School of Information Technologies, University of Sydney, Sydney, Australia*
[4]*Department of Computer Science, Federal University of São Carlos, São Carlos, SP, Brazil*
[5]*Institute of Mathematics and Computer Science, University of São Paulo, São Paulo, SP, Brazil*

Keywords: Road Detection, Deep Learning, Auto-Enconder, U-Net.

Abstract: One of the challenges in extracting road network from aerial images is an enormous amount of different cartographic features interacting with each other. This paper presents a methodology to detect the road network from aerial images. The methodology applies a Deep Learning (DL) architecture named U-Net and a fully convolutional Auto-Encoder for comparison. High-resolution RGB images of an urban area were obtained from a conventional photogrammetric mission. The experiments show that both architectures achieve satisfactory results for detecting road network while maintaining low inference time once DL networks are trained.

1 INTRODUCTION

In the past few years, the usage of high-resolution aerial images along with machine learning techniques applications have been extensively used for cartographic features extraction, mainly road and street geometry. However, extracting such features is not a trivial task considering the large number of different objects interacting with the roads (Mendes and Dal Poz, 2011). Several studies from the most diverse fields are being developed such as route optimization (Silva et al., 2016), landscape changes monitoring and natural resources preservation (Galo, 2000).

Over the years, artificial intelligence problems that had previously been solved using very complicated code began to be explored by Machine Learning (ML). This technique is based on the principle of extracting patterns and data features to feed the algorithm that learns automatically. The introduction of ML allowed solution using real-world data to help in the process of decision-making in several areas (Goodfellow et al., 2016).

Considering the evolution of this computational area, the simple learning classifiers are being replaced by more effective methods that better represent the human brain functionality, such as neural networks. Recently, the Deep Learning (DL) architectures have gained attention due to the current capacity of storage and processing large amounts of data, despite being developed a few decades ago. Nowadays, the scientific world has much more ability to manipulate such amount of data, and therefore several studies are being designed and applying DL. It can attack the most diverse problems since a DL architecture can be very flexible and created to fit a specific dataset.

One of the branches of DL is called Convolutional Neural Network (CNN). The CNN also can be applied in the Natural Language Processing (NLP) problems (Goodfellow et al., 2016). Classification is a task where convolution is vastly used, especially in visual recognition. In 2015, Ronnerberger (Ronneberger et al., 2015) built a fully convolutional network to attack the problem of biomedical image segmentation. However, the architecture was modified to fit the particularities of the task named U-Net. One of these particularities is the fact that thousands of biomedical training images are quite unlikely to have.

In this work the U-Net deep neural networks architecture was applied over the problem of road network detection to verify its accuracy in a dataset, where there are two class labels (road or non-road), but the context is equally important. We compared

Spolti, A., Guizilini, V., Mendes, C., Croce, M., R. de Geus, A., Oliveira, H., Backes, A. and Souza, J.
Application of U-Net and Auto-Encoder to the Road/Non-road Classification of Aerial Imagery in Urban Environments.
DOI: 10.5220/0009337306070614
In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020) - Volume 4: VISAPP*, pages 607-614
ISBN: 978-989-758-402-2; ISSN: 2184-4321

the results obtained with a fully convolutional Auto-Encoder. The dataset is composed of portions of aerial images with dimension 256×256 pixels from an urban area Figure 1. Also, the class labels are black for representing roads and white representing other objects. As mentioned, the context is crucial since many different objects can be easily mixed with roads such as roofs, cars, parking lots, sidewalks, etc.

Figure 1: The problem. The left image is acquired from an airplane, where it has road and non-road objects. The right image is the ground truth image.

The remainder of this paper is organized as follows: Section II discusses related works; Section III presents the proposed methodology, detailing the U-Net along with the architectures used for comparison; the experimental setup, results, and analysis are shown in Section IV; finally, Section V draws the conclusions and suggests directions for future work.

2 RELATED WORK

The work of (Jordan and Ana-Maria, 2013) proposed a new strategy to identify buildings from aerial image analysis. The training stage was carried out by an operator who selected points of interest in the image, structures, and aspects of non-interest, such as vegetation and streets. A combination of shape attributes with an automated region growth technique and Support Vector Machine is used to rank between points of interest and insignificant points. This strategy was tested by manipulating 20 portions of aerial images with a resolution of 256×256 pixels that contained different topologies. This strategy obtained higher results (97,8% accuracy) when compared to other techniques discussed in the literature. Thus, it concludes that the shape descriptors encode an appropriate set of features that allow identifying objects of interest correctly in most images, despite the different shapes and sizes of roofs.

Using deep learning architecture, the method presented in (Mendes et al., 2016) considered the problem of road detection, where given an image, the goal was to classify each pixel of the image in road or non-road. The architecture was a Convolutional Neural Network, targeting a model that employs a sizeable contextual window while maintaining efficient inference properties. For that, a Network-in-Network (NiN) architecture was applied, and the model was converted into a fully convoluted network after the training. The KITTI Vision Benchmark Suite was used for training and evaluation of the proposed architecture. This database offers 289 training images along with its ground truth and 290 test images. A precision of 92% was obtained both using the NiN architecture and without, however, with the NiN architecture, the time of inference was significantly lower. The results revealed that the inference time of the proposed approach is unique at this level of accuracy, with two orders of magnitude faster than other methods with similar performance.

A neural network architecture, named SegNet, was proposed by (Badrinarayanan et al., 2015), which uses a fully convolutive pixel-wise semantic segmentation. The core of the segmentation mechanism consists of an encoder network and a corresponding decoder network, that produces as output segmented pixel information corresponding to each considered class. The encoder network includes 13 convolutional layers. Each layer of the encoder has a similar decoder layer, so the decoder network also has 13 layers. The SegNet architecture performance was measured in two different situations, the first being the classification of roads, trees, walks, cars, etc. The second is the indoor scene segmentation that is of immediate interest to various augmented reality (AR) applications. In the case of the second scene, because it is a problem that contains several classes, the result obtained was not satisfactory. However, for the classification of roads, the SegNet was efficient and achieved results with 90% of accuracy. SegNet was compared to other architectures regarding training time, memory, and efficiency. Some of them had better results; however, more memory was required by the amount of data stored. SegNet is more efficient because it only saves the maximum pool of indexes of resource maps and uses them in its decoder network to achieve a good performance; then using less memory and still obtaining satisfactory results.

Another work using the DL technique was proposed by (Kussul et al., 2017) to classify satellite images about the soil coverage and their cultures. To restore missing data due to clouds and shadows present in the images, a preprocessing phase was necessary. For classification purposes, a fully connected supervised neural network (MLP - Multilayer Perceptron) and a Random Forest was applied, comparing such

techniques with a convolutional neural network. The experiments were carried out using 19 multitemporal images of the region of Ukraine acquired by RS satellites of Landsat8 and Sentinel-1A. Using two architectures variations of a convolutional neural network, called 1-D and 2-D to explore spectral and spatial features respectively. An accuracy of 93.5% and 94.6% respectively were obtained, while Random Forest and MLP obtained 88.7% and 92.7% respectively. Consequently, the proposed architecture proved more useful for the described problem.

Another approach based on deep learning for feature detection in the scope of remote sensing was proposed by (Zou et al., 2015) treating the problem of extraction as a problem of features reconstruction. The proposed method selects the most reconstructive characteristics as the discriminative ones. In the experiments, 2800 orbital images were divided into seven categories (grass, farm, industry, river, forest, residential, parking) for performance evaluation. To address the problem of feature reconstruction, a Deep Belief Network (DBN) was used. An iterative algorithm for learning features was developed to obtain reliable reconstruction weights and characteristics with small reconstruction errors. On average, an accuracy of 77% was reached, and the category with the most misclassification was the industry category with 65%, and the least confused class being the forest with an accuracy of 93.5%. Considering the complexity of the classification type, then the experiments validated the efficiency of the proposed method.

Our proposal uses CNN architectures U-Net and Auto Encoder, which have not yet been applied for extraction of cartographic features using aerial images, specifically on roads. Consequently, the proposal was not used in the scientific literature for such problem.

3 PROPOSED METHODOLOGY

Our methodology presents two DL networks architectures: U-Net and Auto-Encoder. We have chosen these two networks because they are representative of the state-of-the-art in the proposed task and, more importantly, are computationally efficient and capable of considering a large amounts of contextual information, which is crucial in this case. The purpose of this paper is showing comparison of both in the road network detection using aerial images. Each DL architecture is described in subsections below.

3.1 U-Net Architecture

The U-Net has two phases: contraction and expansion. In the contraction path, the input image goes through 2 convolutions 3×3, stride 1, generating 8 feature channels and the ReLU activation function in the first step followed by a 2×2 max pooling operation with stride 2. After each max pooling operation, the number of feature channels is increased by a factor of two, and the input size is reduced by the same factor due to the effects of the max pooling. In the contraction path, a step is defined by 2 convolutions and a max-pooling operation.

After 4 steps, the resulting output is fed to an up-convolution 2×2, stride 2 and 64 feature channels in the first step which is the beginning of the expansion path. Every step of this phase consists of the up-convolution with the parameters described above. Moreover, the output of the same stage of the contraction path is concatenated, and convolutions are applied as in the contraction path. After each step, the number of feature channels is reduced by a factor of two. In the last layer, convolution with a single kernel 1×1 is applied, and the resulting tensor passes through the sigmoid function. Output is a single channel image with pixel values in the interval $[0, 1]$, during inference, thresholded at 0.5 and mapped to black or white for visualization purposes. Black (zero) pixels are pixels classified as roads and white otherwise. Figure 2 represents the U-Net used in our work.

3.2 Auto-Encoder Architecture

It is based on (Long et al., 2015) proposing fully convolutional networks for semantic segmentation. A fully convolutional network does not have any fully connected layer, such that all neurons from one layer are connected to all neurons from the next layer. This was common in most topologies up to this point, especially in deeper layers. However, there are benefits in eliminating full connections: 1) decrease in the number of trainable parameters; 2) preservation of spatial correlation; 3) images of any size can be equally processed using the same network.

Aiming for future online applications and onboard processing, a relatively simple topology was used in this work, composed of three convolutional layers, with filter size 5×5 and three deconvolutional layers (the convolutional transpose), with the same filter size. Each layer is followed by a ReLU (Xu et al., 2015), to introduce non-linearities, and convolutional layers receive a max-pooling of 2, to downsample spatial dimensions, while deconvolutional layers upsample input images by the same factor, doubling their spatial dimensions. Convolutional and deconvolutional layers are connected by a fourth convolutional layer, with filter size 3×3, no activation func-

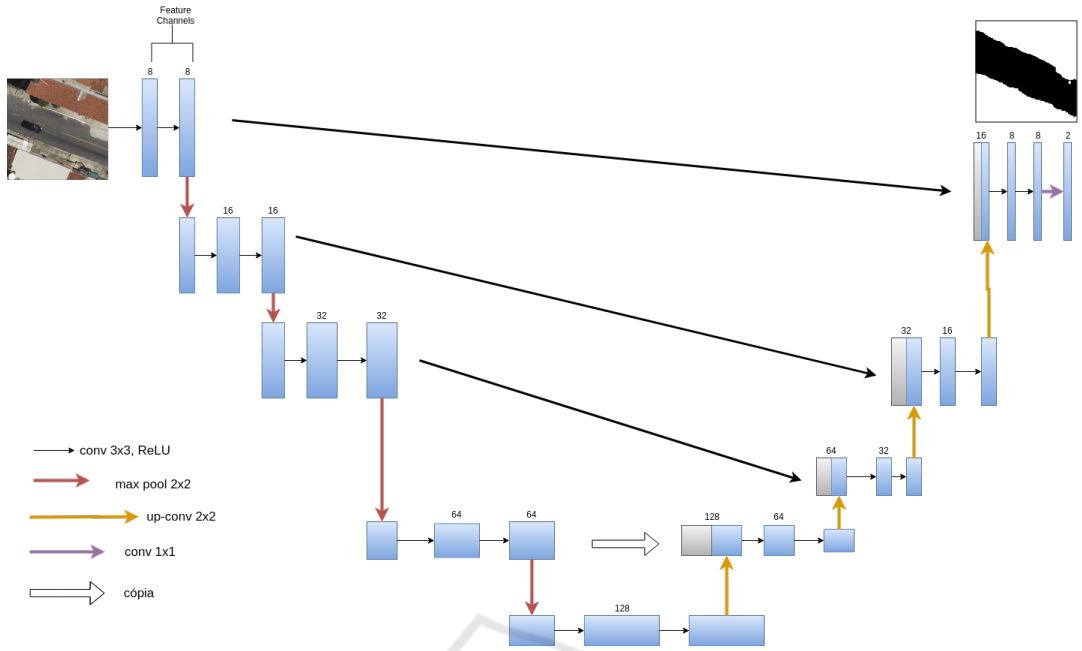


Figure 2: U-Net architecture diagram used in our work.

tion and dropout (Srivastava et al., 2014) of 0.6 (60% of nodes are randomly turned off during training, to increase the network's ability to generalize over inputs). The number of neurons in each convolutional layer was respectively 64, 128 and 256, and these numbers were inverted in the deconvolutional layers.

The final layer produces 1-channel outputs, with a sigmoid activation function to provide values between $[0,1]$ that serve as a probabilistic classification for each pixel. The crossentropy loss function was optimized during training phase, based on ground-truth labeled information, using an Adam optimizer (Kingma and Ba, 2014) with a learning rate of 10^{-4}. Figure 3 shows Auto-Encoder applied in our work.

3.3 Differences between U-Net and Auto-Encoder

The main difference between the architectures U-Net and Auto-Encoder are the skip connections, which connect each stage of the contraction path to the corresponding stage of the expansion. These skip connections tend to avoid local minima during training. Moreover, one can view them as simplifying or dividing the training. For instance, during early iterations of training, if the network has no skip connections, it has to adjust all the weights of the network to generate any meaningful result. Meanwhile, a network with skip connections has to modify only the weights of the layers that comprise the shortest path from input to output to do the same. Although this difference simplifies the training (at least the initial iterations), whether it is beneficial to the overall accuracy can be only accessed through experimental data.

4 EXPERIMENTAL RESULTS

To evaluate the performance of the proposed methodology, we implemented the described system in Tensorflow (Abadi et al., 2015) and Keras. We tested its performance on images test set. Python language was selected, due to its widespread use in DL applications, and a Titan X Pascal GPU card (12GB) was used to increase computational performance for model training. Once training is complete, model can be stored for later use, allowing fast inference in similar GPU cards, standard CPU machines or even in onboard computers, for online classification.

The dataset is composed of 3814 images of 256×256 divided into 80% for training, 10%, for testing and 10% for validation. The architectures were implemented in Python using the Keras (Chollet et al., 2015) library with Tensorflow. The network is trained for 300 epochs or until the validation accuracy stops improving (for 30 epochs in a row).

To analyze the U-Net and Auto-Encoder performance to predict images with different levels of difficulty, then five images were selected from the testing database. Figure 4 shows the original image, its label and how the U-Net and Auto-Encoder performed. In Image 1 and Image 2, the U-Net and Auto-Encoder

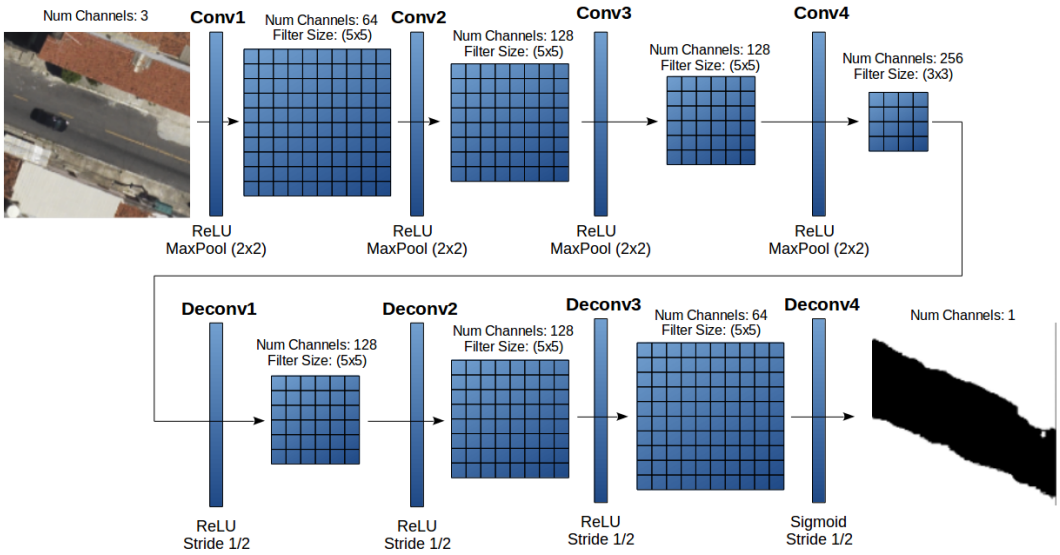


Figure 3: Auto-Encoder CNN diagram used in our work.

classification was very accurate. However, in Image 3, 4 and 5 the classification was a little weak in both architectures because these images contain objects that can be easily mixed with roads considering the color similarity. In Image 5, the U-Net architecture could distinguish the roof from the roads besides the color similarity however the classification is very different from the ground truth. On the other hand, with the Auto-Encoder architecture the prediction was wrong again because of the color that can be easily mistaken even for humans.

Table 1 shows a comparison between the U-Net and Auto-Encoder architectures. It is possible to notice that the Auto-Encoder showed the best performance. The metrics of the two architectures were very similar, however the Auto-Encoder still obtained better results except for the accuracy measurement where U-Net acquired 89.1% and the Auto-Encoder 88.0%

Table 1: Results.

	U-Net	Auto-Encoder
Accuracy	**0.891**	0.880
Precision	0.800	**0.899**
Recall	0.856	**0.919**
F-Measure	0.820	**0.907**

To verify the viability of the methodology, a new image with similar data to those used during training was classified. New aerial images obtained by an airplane were classified allowing a qualitative analysis of model capability in classifying roads and non-roads.

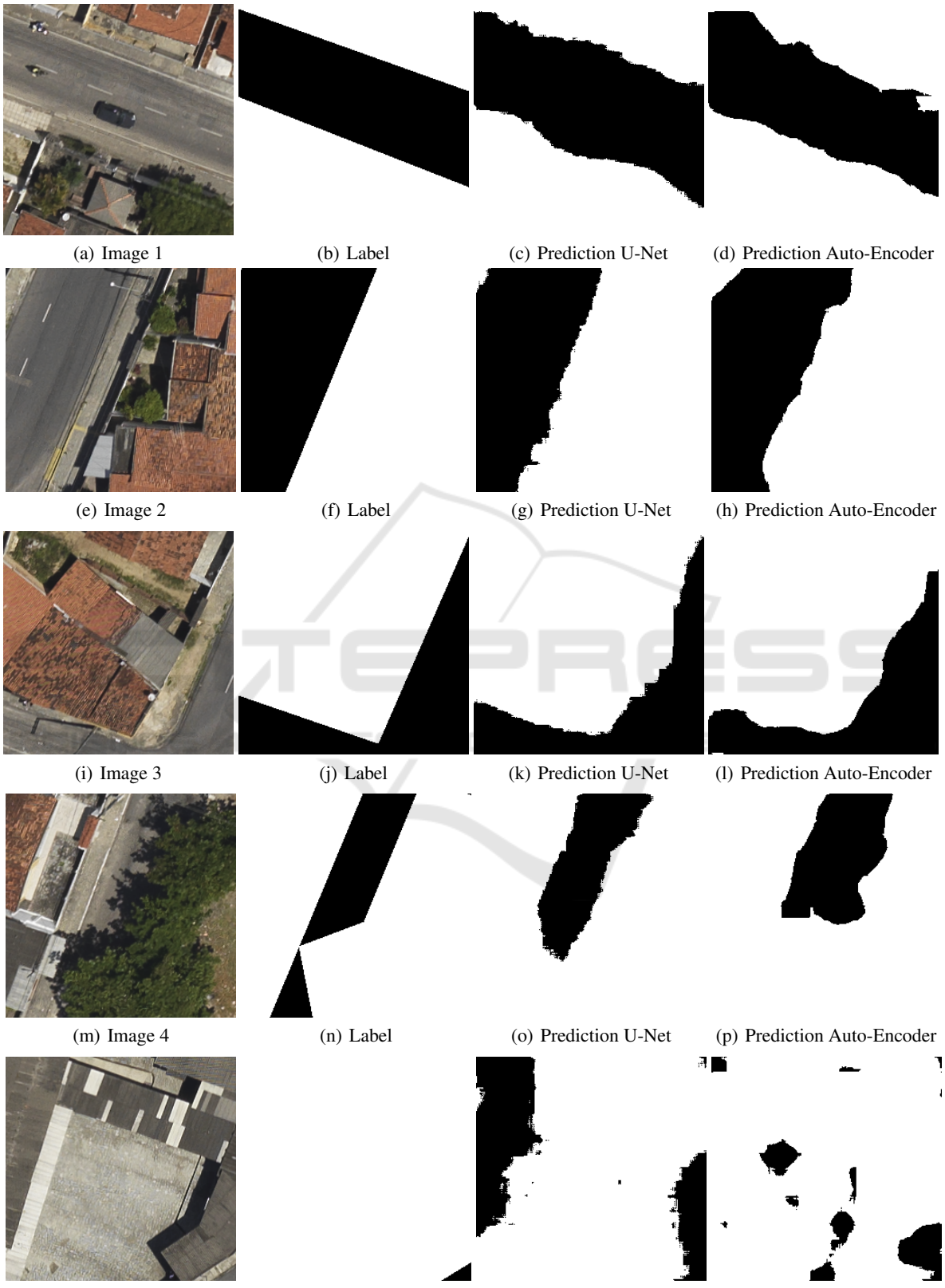
Figure 5 and 6, both architectures shows similar results. However, Figure 6 demonstrates the complete classified aerial image by Auto-Encoder showing that this CNN predicted more roads compared to U-Net.

Even though in specific scenarios the results are not satisfactory, techniques such as skeletonization and mathematical morphology can be applied to reconstruct the part of the image that was not well classified. This technique has the classified result as input and a geometric network (vectors) as output.

5 CONCLUSIONS

The proposed methodology applied two deep learning architectures to the problem of road classification from an aerial image. The aim is to classify the objects in an aerial image as a road or non-road helping in the mapping process of a region, saving resources and time of the responsible operator.

The results showed the performance of the U-Net to classify an aerial image and its objects in road and non-road. On the other hand, for future works, it is essential to improve the architecture so it is able to analyze the context where every pixel is located so objects with color similarity will not be mistaken with roads as shown in Figure 4 - Image 5. One possible solution is to observe the connection between the black pixels in the resulting image, in Image 5 for example using the Auto-Encoder architecture, the prediction is completely disconnected, and therefore it can be eliminated or improved with mathematical morphology.



(a) Image 1 (b) Label (c) Prediction U-Net (d) Prediction Auto-Encoder

(e) Image 2 (f) Label (g) Prediction U-Net (h) Prediction Auto-Encoder

(i) Image 3 (j) Label (k) Prediction U-Net (l) Prediction Auto-Encoder

(m) Image 4 (n) Label (o) Prediction U-Net (p) Prediction Auto-Encoder

(q) Image 5 (r) Label (s) Prediction U-Net (t) Prediction Auto-Encoder

Figure 4: Several levels images along with its label and prediction applying the U-Net and Auto-Encoder architectures.

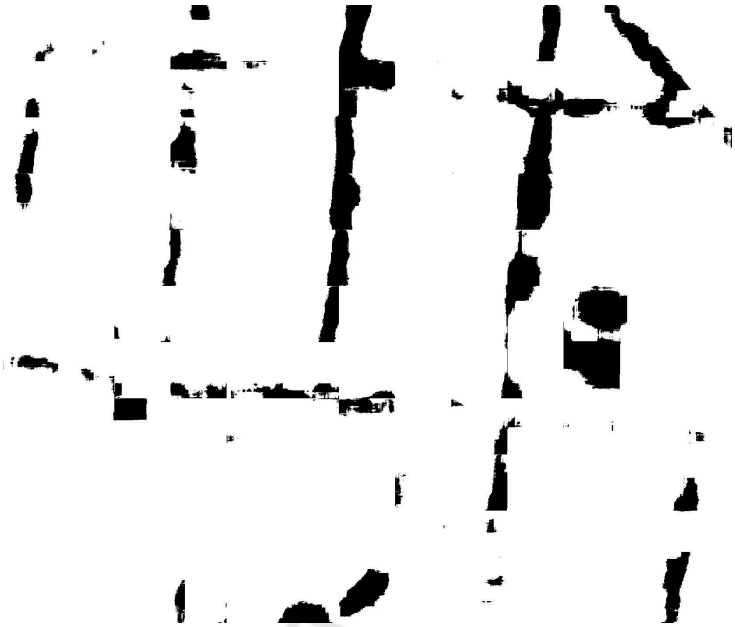


Figure 5: Image reconstruction with U-Net. The reconstructed image is the junction of the 256×256 classified figures in a test set, black represents roads and white non-roads.

Figure 6: Image reconstruction with Auto-Encoder. The reconstructed image is the junction of the 256×256 classified figures in a test set, black represents roads and white non-roads.

Also, a larger database containing different images perspectives would help the network to learn and therefore better classify these images. Ultimately, even with its mistakes, the Auto-Encoder architecture can be advantageous as input in techniques such as skeletonization mentioned before therefore assisting in many applications such as route optimization, minimizing resources, time and operators effort. Lastly, we can investigate the use of multitemporal images in our problem.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Federal University of Uberlândia and University of Campinas,

CNPq (National Council for Scientific and Technological Development) under Grant #400699/2016-8 and #301715/2018-1. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Finance Code 001.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems.

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2015). Segnet: A deep conv. encoder-decoder arch. for image segmentation. *arXiv preprint*.

Chollet, F. et al. (2015). Keras. https://github.com/ fchollet/keras.

Galo, M. (2000). Aplicacao de redes neurais artificiais e sens. remoto na carac. ambiental do parque est. morro do diabo. *Sao Carlos*.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT.

Jordan, T. and Ana-Maria, C. (2013). A supervised training and learning method for building ide. in remotely sensed imaging. In *SRSE*, pages 73–78.

Kingma, D. and Ba, J. (2014). Adam: Method for stochastic optimization. In *ICLR*.

Kussul, N., Lavreniuk, M., Skakun, S., and Shelestov, A. (2017). Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, pages 778–782.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440.

Mendes, C. C. T., Fremont, V., and Wolf, D. F. (2016). Exploiting fully convolutional neural networks for fast road detection. In *IEEE Int. Conference on Robotics and Automation (ICRA)*, pages 3174–3179.

Mendes, T. S. G. and Dal Poz, A. P. (2011). Classificacao de imagens aereas de alta-resolucao utilizando redes neurais artificiais e dados de varredura a laser. *Simp. Brasileiro de Sensoriamento Remoto*, pages 7792–7799.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer.

Silva, F., Minette, L. J., Souza, A. P. d., Moraes, A. C. d., and Schettino, S. (2016). Classification of forest roads and determination of route using geographic information system. *Revista Arvore*, pages 329–335.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, pages 1929–1958.

Xu, B., Wang, N., Chen, T., and Li, M. (2015). Empirical evaluation of rectified activations in convolutional network.

Zou, Q., Ni, L., Zhang, T., and Wang, Q. (2015). Deep learning based feature selection for remote sensing scene class. *GRSL*, pages 2321–2325.