Unsupervised Segmentation of Cattle Images Using Deep Learning

Vinícius Guardieiro Sousa*, André R. Backes*
*School of Computer Science, Federal University of Uberlândia, Brazil
viniciusguardieiro@gmail.com; arbackes@yahoo.com.br

***Abstract*—In this work, we used the Deep Learning (DL) architecture named U-Net to segment images containing side view cattle. We evaluated the ability of the U-Net to segment images captured with different backgrounds and from the different breeds, both acquired by us and from the Internet. Since cattle images present a more constant background than other applications, we also evaluated the performance of the U-Net when we change the numbers of convolutional blocks and filters. Results show that U-Net can be used to segment cattle images using fewer blocks and filters than traditional U-Net and that the number of blocks is more important than the total number of filters used.**

***Keywords*—U-Net, semantic segmentation, cow detection, deep Learning.**

I. Introduction

Today, information technology is a key tool to improve the management of cattle and dairy herds in the industry. Many technologies, like computer vision-based systems, help us to detect and track animals, and to analyze their social behavior, thus enabling us to detect changes in their usual behavior in a farm environment [1]–[3].

Although there are many challenges in the identification of animal species, this is a less complex task when compared with plant identification. Animal biology is more consistent among individuals, i.e., they have well-defined morphological patterns and have few intraspecies variations. As a result, many approaches have been developed over the years for this task [3]–[8].

Recently, convolutional neural networks (CNN) have emerged as a new approach to classify and segment images in highly complex situations. They are biologically inspired by the concept of the receptive field present in the human visual system and can learn the attributes from the data. They accomplish that by exploring spatial correlations in the image, transforming it, and by reducing the image information to more relevant semantic information [9]–[11].

In this paper, we address the problem of cattle segmentation from single images using CNN. To accomplish this task we propose the use of U-Net deep neural networks architecture. We also evaluated the impact of using different numbers of convolutional blocks and filters in the U-Net as cattle images present a more constant background than other applications. The remaining of this paper is organized as follows: Section II reviews the recent literature in the area of animal segmentation in a farm environment. Section III describes image segmentation and the semantic segmentation network used in this work. In Section IV we describe our experimental setup and the dataset used for the experiments. Section V shows the yielded results and discusses them, while Section VI concludes this work.

II. Related Work

Target detection is a fundamental step for many applications involving animals in a farm environment. This has motivated the development of many approaches for animals' segmentation, mostly focused on the analysis of video sequences for perceiving animal behavior.

In [12] an algorithm is proposed to segment side-view images of cows on a farm. The authors computed the bounding rectangle of cows using the frames difference method to extract the local background in frames. They also adjusted the summation coefficients on RGB channels to improve the contrast between the target and the background images in order to improve cow detection.

The authors in [2] used simple background subtraction techniques to extract cow's region in pasture using a fixed bird's eye video camera. They were able to estimate cow's size and location and to track the animal's movements to detect any interaction. To shorten the processing time, the authors proposed the use of a composite background image and brightness to improve the detection accuracy and to reduce the search images [13].

In [3] the authors proposed an algorithm to detect mounting behavior in dairy cattle. The proposed approach combines the black-and-white pattern and texture information of Holstein cows to detect their regions in the complex background of a farm. From each region, they extract geometric features and a SVM is used to classify each into mounting and non-mounting regions, which allowed the identification of mounting behavior.

An algorithm to segment and track piglets from the top was proposed in [14]. It combines image differencing with respect to a median background and a Laplacian operator to segment moving piglets present in a pen covered with plain straw. For tracking, piglets were modeled as ellipses initialized on the blobs previously detected.

In [1] an algorithm to segment sow in grayscale video images obtained from the farrowing pens is proposed. It combines dyadic wavelet transform with a Gaussian mixture model to perform illumination variant background subtraction. The authors stated that the method is efficient even when images present very poor quality.

Group-housed pigs segmentation is proposed in [5]. To accomplish that the authors used a combination of Mixture of Gaussians (MoG) using prediction mechanism (PM) and threshold segmentation algorithm to segment and detect foreground objects of pigs in overhead views of group-housed environments. They claim that their approach is robust to a series of variations in the scene, including light changes, the influence of ground urine stains, water stains, pigs' slow movement patterns, and varying colors of foreground objects.

In [6] a novel approach to estimate the illumination and reflectance of an image is described. The authors used a homomorphic wavelet filter (HWF) and defined a wavelet quotient image (WQI) model that is capable of segmenting sows in grayscale video captured in complex farrowing pens. They also claim that the approach could be applied to detect the domestic animals in complex environments under a great variety of conditions.

III. Image segmentation using deep learning

Image segmentation is usually described as the process of dividing an image into different regions, each one related to a different object. From a semantic point of view, this process enables us to simplify the image content, as it group pixels of the image according to their properties, thus creating different regions of interest that can be explored or analyzed by other algorithms. This is a fundamental step for many other algorithms and applications, which makes image segmentation an important research field. Its use is well documented in many, and different applications, ranging from computer-aided diagnosis [15], [16] to the development of autonomous vehicles [11], [17].

Until recently, the main approach to perform image segmentation was to explore the relationship between pixels in order to detect the regions with similar properties. To accomplish that, researchers relied on hand-crafted descriptors to compare pixels in nearby regions. However, these descriptors are suitable for specific types of image content and may fail for other classes of images.

This has changed with the emergence of segmentation networks. These networks use the same concept as other Convolutional Neural Networks (CNN), with the difference that they aim the classification of image pixels into different classes, while traditional CNN aims to classify at the image level. As the CNN, segmentation networks are able to compute the best set of descriptors for a given image segmentation problem.

One of the most used segmentation networks is the U-Net [18]. It was initially proposed to address the problem of biomedical image segmentation. This network is structured into two segments, named contracting and expanding networks, given the network its characteristic "U" shape.

The contracting network is similar to a traditional CNN and it is the first segment of the image. It is composed of 5 blocks of two convolutional layers (each block followed by a ReLU unit) and a 2×2 max-polling layer. In this segment, as the input's dimensions halves, the number of filters doubles.

The expanding network operates in an inverse way: it is also composed of 5 blocks of two convolutional layers (this time without a ReLU unit). However, instead of a 2×2 max-polling layer, each block is followed by a 2×2 Transpose convolution layer to restore the original image size. Additionally, it concatenates the features maps from the respective step of the contracting network. This is performed so that the network can properly reconstruct the image. In this segment, as the input's dimensions doubles, the number of filters halves, so that both networks segments present the same number of blocks.

Lastly, a 3×3 convolution layer is applied to the output of the expanding network. This layer is responsible to convert the output network of each pixel to a given segmentation class.

IV. Experiments

In order to perform our experiments, we created a dataset with cattle images. Initially, our proposal was to collect images of cattle from a farm to train the segmentation network, thus ensuring control in the acquisition process. However, this task proved to be fruitless and very laborious due to the stress caused to the animal, and it was, therefore, abandoned. The number of images obtained by this process was not enough to train a network (only 34 images) and, thus, we collected cattle images from the Internet to increase the number of images in our dataset. We manually selected a total of 179 RGB images containing the side view of the cattle, without watermarks or other elements that could hinder the training process. After collecting the dataset (213 images) cropped and adjust the image to ensure that all of them have 512×512 pixels size. Additionally, we manually labeled the cattle in all images. Figure 1 shows some examples of the images in our dataset.

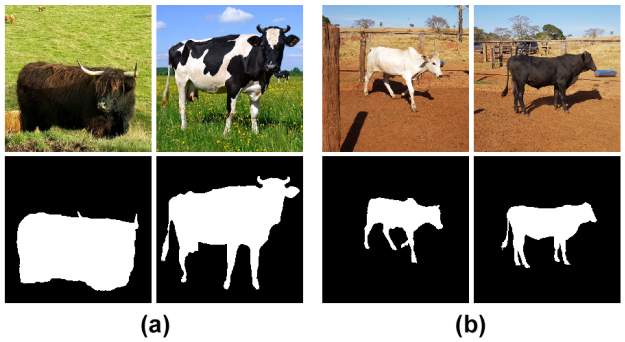

Fig. 1. Examples of cattle images in our dataset: (a) Images collected from Internet; (b) Images acquired by the authors.

For this segmentation task, we created a personalized U-Net with an input size of 256×256 pixels. Traditionally, this network has 5 convolutional blocks in each contracting and expanding path. We kept this structure but we used a different configuration in the number of filters. We opted to start with 16 filters in the first block and to double this value as we advance to the next block until the maximum of 256 filters is achieved in the 5th block. We opted for this approach as the number of

TABLE I
U-NET CONFIGURATIONS USED IN THE EXPERIMENTS.

# of blocks	# of filters	# of parameters
5	16, 32, 64, 128, 256	3,331,697
4	16, 32, 64, 128	823,921
3	16, 32, 64	196,721
2	16, 32	39,793
1	16	2,785
1	32	10,177
1	64	38,785
1	128	151,297
1	256	597,505

parameters in the original U-Net may be prohibitive for some computers to handle.

Since cattle images present a more constant background than other applications, we also evaluated the impact of using different numbers of convolutional blocks in the segmentation results. To accomplish that, we removed the last block of the contracting path and its respective block in the expanding path. We also evaluated if a single block is enough if a proper number of filters is provided. Table I presents the U-Net configurations evaluated.

We compared the segmentation results provided by the network with our manual segmentation using the Dice coefficient, D:

$$D = 2\frac{|A \cap B|}{|A| + |B|}, \quad (1)$$

where D, $0 \leq D \leq 1$, is the level of similarity between images; A and B are the segmentation provided by our method and by the expert, respectively. The more the value D is close to 1, the more similar the images are.

V. RESULTS AND DISCUSSION

This section reports the results achieved for each network. We conducted the experiments on a personal computer with Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz, 32 GB RAM, 64-bit Windows OS, and GPU NVIDIA GeForce GTX 1050 Ti, 4 GB GDDR5. We also used Python 3.6 and Keras 2.1.6-tf with TensorFlow 1.10.0 and CUDA Toolkit 9.0 to implement and test the experiments. We executed the training for 1,000 epochs and to compose the training set we randomly selected 80% of the images.

Figure 2 presents the boxplot distribution of the Dice coefficient obtained when using different numbers of convolutional blocks in the network. Notice that the reduction from 5 to 3 blocks has almost no impact on the performance of the network, as these networks obtained high values for most of the samples. This is corroborated by Figure 3, which shows that both average and median Dice coefficients remained high although the network had undergone a great reduction in the number of parameters. For example, the network with 3 blocks has only 5.9% of the trainable parameters of one with 5 blocks. This confirms our original idea that cattle images present a more constant background, thus requiring less semantic levels of feature extractions in the network. However, a further reduction in the number of convolutional blocks compromises the network's ability to segment images, as shown by the drastic reduction in the Dice coefficients achieved.

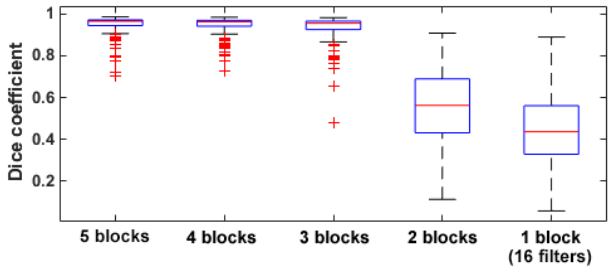

Fig. 2. Dice coefficient distribution obtained for different numbers of convolutional blocks.

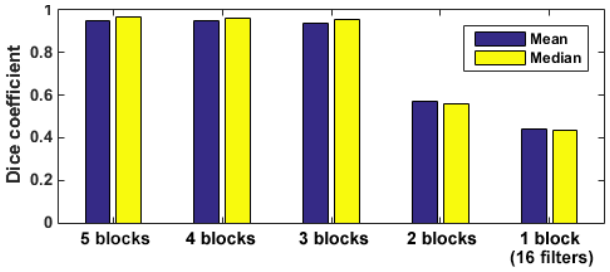

Fig. 3. Average and median Dice coefficient obtained for different numbers of convolutional blocks.

By reducing the number of convolution blocks we also reduce the number of trainable parameters. So, the deterioration in the results may be due to the reduction in the number of parameters instead of the number of blocks. To test this theory we also evaluated if a single block is enough if a proper number of filters is provided.

Figures 4 and 5 show, respectively, the boxplot distribution and the average and median Dice coefficient as we change the number of filters in a single convolutional block. Notice that, independent of the number of filters used, the network is unable to achieve a segmentation result similar to those of the networks containing more blocks. Although a single block with 256 filters contains more trainable parameters (597,505) it performs poorly when compared to a network with 3 convolution blocks (196,721), which contains only 32.92% of the parameters. As we can see, the number of blocks improves the ability of the network to extract semantic features and to segment cattle from image, a more important feature of the network than its total number of parameters.

Figures 6, 7 and 8 show some examples of images segmented using different networks. It is noticeable that the deterioration of the results is a consequence of the reduction of the number of blocks from five to two, and how a single block, despite the number of filters used, is not suitable for the task.

VI. CONCLUSION

In this paper, we addressed the problem of cattle image segmentation. The main goal was to segment side view images

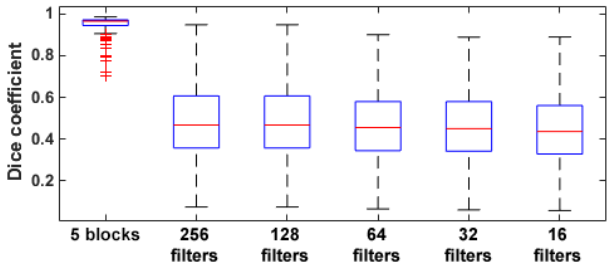

Fig. 4. Dice coefficient distribution when using a single convolutional block and a different number of filters. Original U-Net refers to the use of the 5 convolutional blocks.

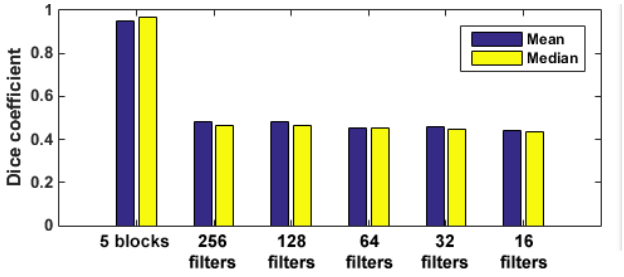

Fig. 5. Average and median Dice coefficient when using a single convolutional block and different number of filters. Original U-Net refers to the use of the 5 convolutional blocks.

of cattle with different backgrounds. To accomplish that we used a personalized version of U-Net. We also evaluated the impact of using different numbers of convolutional blocks and filters in the U-Net architecture. Results showed that U-Net is able to segment the images using fewer blocks and filters than traditional U-Net and that the number of blocks is more important than the total number of filters used. In future work, we aim to expand the dataset used in the experiments, explore other models of semantic segmentation networks and embed our approach in a smartphone.

Acknowledgment

André R. Backes gratefully acknowledges the financial support of CNPq (National Council for Scientific and Technological Development, Brazil) (Grant #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

[1] G. J. Tu, H. Karstoft, L. J. Pedersen, and E. Jørgensen, "Segmentation of sows in farrowing pens," *IET Image Processing*, vol. 8, no. 1, pp. 56–68, 2014.

[2] R. Nishide, Y. Hosomi, T. Ohkawa, K. Oyama, and C. Ohta, "Detecting and tracking breeding cows from bird's eye video of pasture," in *Proceedings of the 5th IIAE International Conference on Intelligent Systems and Image Processing*, 2017, pp. 239–246.

[3] Y. Guo, Z. Zhang, D. He, J. Niu, and Y. Tan, "Detection of cow mounting behavior using region geometry and optical flow characteristics," *Computers and Electronics in Agriculture*, vol. 163, p. 104828, 2019.

[4] L. C. Silva Júnior, M. B. S. Pádua, L. M. Ogusuku, M. Keese Albertini, R. Pimentel, and A. R. Backes, "Wild boar recognition using convolutional neural networks," *Concurrency and Computation: Practice and Experience*, vol. n/a, no. n/a, p. e6010. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.6010

[5] Y. Guo, W. Zhu, P. Jiao, and J. Chen, "Foreground detection of group-housed pigs based on the combination of mixture of gaussians using prediction mechanism and threshold segmentation," *Biosystems Engineering*, vol. 125, pp. 98–104, 2014.

[6] G. J. Tu, H. Karstoft, L. J. Pedersen, and E. Jørgensen, "Illumination and reflectance estimation with its application in foreground detection," *Sensors*, vol. 15, no. 9, pp. 21 407–21 426, 2015.

[7] G. Ding, Y. Song, J. Guo, C. Feng, G. Li, B. He, and T. Yan, "Fish recognition using convolutional neural network," in *OCEANS 2017-Anchorage*. IEEE, 2017, pp. 1–4.

[8] J. G. A. Barbedo, L. V. Koenigkan, T. T. Santos, and P. M. Santos, "A study on the detection of cattle in uav images using deep learning," *Sensors*, vol. 19, no. 24, p. 5436, 2019.

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

[10] 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, pp. 770–778.

[11] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, and K. Dietmayer, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[12] Z. Kaixuan and H. Dongjian, "Target detection method for moving cows based on background subtraction," *International Journal of Agricultural and Biological Engineering*, vol. 8, no. 1, pp. 42–49, 2015.

[13] M. Aotani, R. Nishide, Y. Takaki, C. Ohta, K. Oyama, and T. Ohkawa, "Refined cattle detection using composite background subtraction and brightness intensity from bird's eye images," in *Proceedings of the Ninth International Symposium on Information and Communication Technology*, 2018, pp. 243–250.

[14] N. J. McFarlane and C. P. Schofield, "Segmentation and tracking of piglets in images," *Machine vision and applications*, vol. 8, no. 3, pp. 187–193, 1995.

[15] T. Stan, Z. T. Thompson, and P. W. Voorhees, "Optimizing convolutional neural networks to perform semantic segmentation on large materials imaging datasets: X-ray tomography and serial sectioning," *Materials Characterization*, vol. 160, p. 110119, 2020.

[16] G. Bras, V. Fernandes, A. C. de Paiva, G. B. Júnior, and L. Rivero, "Transfer learning method evaluation for automatic pediatric chest x-ray image segmentation," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 2020, pp. 128–133.

[17] M. Treml, J. Arjona-Medina, T. Unterthiner, R. Durgesh, F. Friedmann, P. Schuberth, A. Mayr, M. Heusel, M. Hofmarcher, M. Widrich *et al.*, "Speeding up semantic segmentation for autonomous driving," in *MLITS, NIPS Workshop*, vol. 2, no. 7, 2016.

[18] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

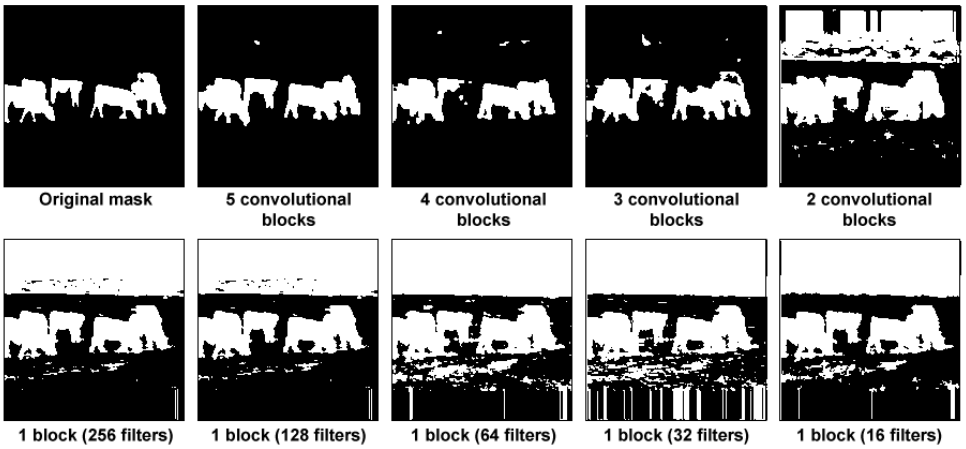

Fig. 6. Examples of cattle images and their respective segmentations.

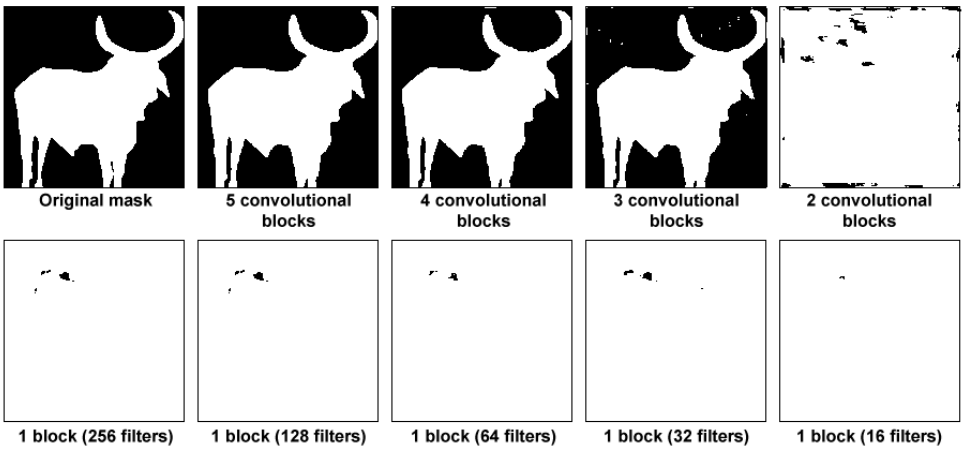

Fig. 7. Examples of cattle images and their respective segmentations.

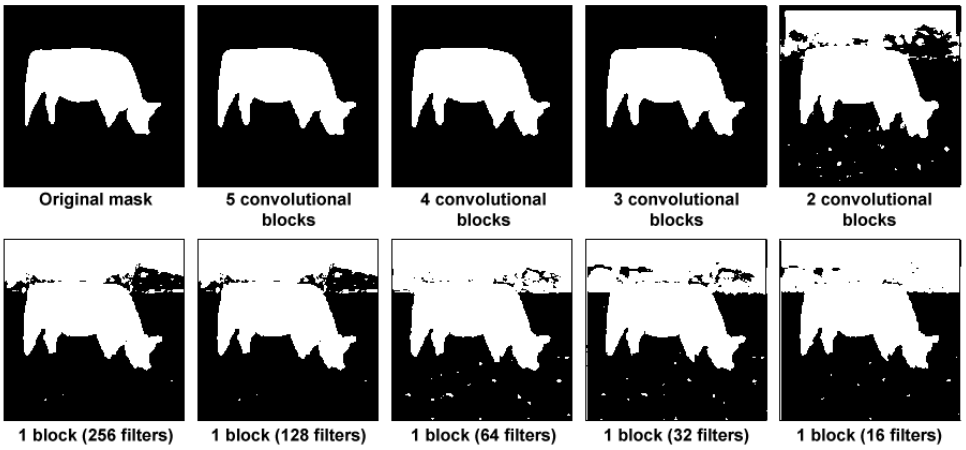

Fig. 8. Examples of cattle images and their respective segmentations.