Segmenting Live Cattle using a New Approach to Combine Superpixels and SegNets

Diogo Nunes Gonçalves[1], Wesley Nunes Gonçalves[2], Rodrigo da Costa Gomes[3],
Anderson Viçoso de Araujo[4], Julia Gindri Bragato Pistori[5], Gabriel Toshio Hirokawa Higa[6],
Vanessa Aparecida de Moraes Weber[7], Hemerson Pistori[8]
[1, 5, 6, 7, 8]*Universidade Católica Dom Bosco*, Campo Grande, Brazil
[1, 2, 4, 8]*Universidade Federal de MS*, Campo Grande, Brazil
[3]*EMBRAPA Gado de Corte*, Campo Grande, Brazil
[7]*Universidade Estadual de MS*, Campo Grande, Brazil
[7]*KeroW Soluções de precisão*, Campo Grande, Brazil
gabrieltoshio03@gmail.com[6]

***Abstract*—A new strategy for cattle image segmentation is proposed by combining the strengths of SegNets and superpixel classification using CNNs. The individual strengths of these segmentation techniques can be seen as complementary. Thus, we investigate the combination of both through the following operators: MEAN, MULT, MAX, OR, and AND. This new approach is tested on a dataset containing 154 labeled images from cattle captured in a real livestock farm scenario, with complex poses and background. A pixelwise accuracy of 94.1% has being achieved by the proposed approach, which is higher than the original methods applied separately.**

I. Introduction

The use of technologies has become a great ally of livestock farmers in the last few years, as these technologies are reliable solutions to increase productivity. Precision livestock farming technologies can help the farmer in executing several tasks, such as real-time monitoring of the herd, body temperature analysis (which can help diagnose the presence of inflammation), heart and respiratory rate monitoring, behavior assessment, among others. By using computer vision-based techniques, these tasks can be performed without the need to get close to the animals, avoiding the stress caused by daily management, which may result in increased quality of life and well-being, productivity and profitability [1].

A lengthy procedure is required to acquire an efficient automated system for herd analysis. First, videos of the animals in the field are obtained by means of cameras, which may be installed in fixed structures or attached, for instance, to drones [1]. Then, the preprocessing and segmentation of the images is carried out [2]. This is an important step. By image segmentation, irrelevant objects (*i.e.* image noise) are removed in order to eliminate possible sources of error in the automatic processing, thus developing a system with greater performance. High-performance methods for automatic evaluation of parameters within the herd require, therefore, some preprocessing labor before they can be effectively applied. In general, it is well acknowledged that correctly segmenting images is important for automated precision livestock farming. As Bello *et al.* [3], [4] write, there are many variables that, concerning the body of the animal, require the correct segmentation so that the posterior automated processing and analysis will give the correct results.

As Chen *et al.* [5] show, the task of segmenting livestock animals such as cows and pigs in images has accompanied the rise of deep learning over traditional computer vision techniques. In general, deep learning has become extensively used for semantic segmentation, with SegNets and its derivations being one of the most cited deep networks to this end [6]–[8]. One weakness of SegNets, however, is that they usually fail to correctly classify pixels that are near object edges, therefore producing sub-optimal outlines [9]. This problem is particularly relevant when dealing, for instance, with prediction of livestock weight via automated analysis of images (*i.e.*, computer vision-based techniques). On the other hand, superpixel-based segmentation techniques can produce better outlines, but with inferior performance on pixels outside the object borders [10]. Both techniques, therefore, can be taken as complementary to each other.

The main contribution of this paper is a novel way to combine SegNets and superpixel classification. We show that by combining both techniques for automated image semantic segmentation, it is possible to achieve better performance than that of each of those techniques used separately. Their results are combined using post-processing with pixelwise operators. Along with this technique, a new annotated cattle image dataset captured in uncontrolled farm environments is presented. We support our claims by presenting the results of experiments using this new dataset. The results show that the proposed approach surpasses the previous two when used separately, when assessed by accuracy.

It is clear from the literature (e.g., in Bello et al. [3] and in Chen et al. [5]) that tasks in precision livestock farming usually require instance segmentation, instead of segmentation by class. Even so, the application of the proposed method for instance segmentation tasks is possible and can be investigated in further studies. Furthermore, although this work focuses on cow images, the possibility of applying the techniques herein proposed to images of other animals, such as pigs and sheep, remains open and is likely to yield good results, since they face some of the same difficulties. Witte *et al.* [11], for example, studied the question of instance segmentation of pigs, and Sant'Ana *et al.* [12] studied the possibility of segmenting sheep in images by classifying superpixels.

II. Materials and Methods

A. Proposed Approach

The proposed approach combines two image segmentation strategies, making use of the complementarity of each one's strengths to enhance the performance of the system as a whole. The first strategy utilizes convolutional neural networks (CNNs) to classify superpixels (*i.e.* parts of the image, groups of pixels with common features), since they are able to adjust to the edges of the objects in the image. The second strategy uses CNNs that classify each pixel, according to the specific task. In our case, the classes are cattle (C) and background (B), for any of the strategies. This second strategy differs from the first mainly because it is able to incorporate contextual information derived from the whole image, doing so through consecutive layers that capitalize on an increased receptive field. The proposed approach combines both strategies using operators to segment cattle with better edge details while still making use of contextual information.

1) Segmentation Strategies: The first strategy, which is called SCNN in this text, was proposed by Ferreira et al. [13] and uses the superpixels generated by simple linear iterative clustering (SLIC) to train a CNN. Given a set of n training images $(I_1, \ldots, I_n)$, each image I_i $(1 \leq i \leq n)$ is divided into approximately k superpixels using SLIC, producing thereby a set of superpixels $S^{I_i} = S_1^{I_i}, \ldots, S_k^{I_i}$. The superpixels from all training images are labelled as either background or cattle.

During the labelling process, it is possible that not all pixels in a superpixel belong to only one class. In this case, the superpixel is labelled with the visually dominant class (*i.e.*, the class to which most of the pixels in the superpixel belong). Therefore, at the end of this step, there is a unique label $r_j^{I_i}$ $(1 \leq j \leq k)$ for each superpixel $S_j^{I_i}$, as illustrated in the last column of Figure 1. After the labelling process, a new set $S = [S^{I_1}, \ldots, S^{I_n}]$ is constituted. This set S is then used to train a CNN. This process is illustrated in Figure 1.

After training, an image given as input is segmented as follows. First, superpixels are obtained and classified by the CNN. Then, each pixel of the image is classified according to the class of the superpixel to which it belongs.

The second strategy, SegNet, is a convolutional neural network for segmentation proposed by Badrinarayanan et al. [14]. The architecture of this network is divided in two parts: (1) the encoder, which is responsible for extracting features from the input image; and (2) the decoder, whose purpose is to reconstruct the segmented image by using the extracted features. In the original work, the SegNet encoder is topologically identical to the convolutional layers of a VGG16. In our experiments, both SCNN and SegNet were evaluated with three different backbones: VGG16, VGG19 and ResNet50.

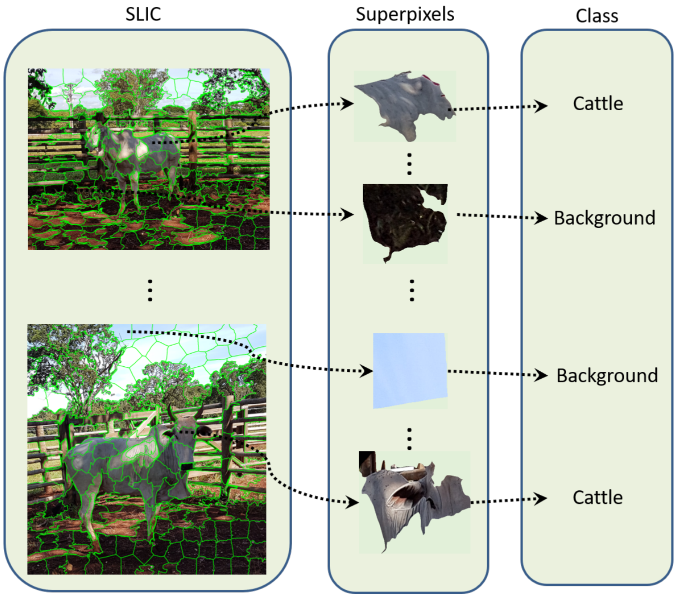


Fig. 1: Illustration of the labelling process used to create the dataset. In the first column, the SLIC algorithm is applied to each of the training images, resulting in the superpixels of the dataset. Some samples are shown in the second column, with their corresponding labels in the third column.

2) Combination of strategies: In our approach, an operator is applied to the output of the two strategies (SCNN and SegNet) to generate the segmented image. The proposed approach maintains the advantage of each strategy and is illustrated in Figure 2. First, an image is segmented using the two strategies investigated in this work (SCNN and SegNet). Then, the outputs are combined by using a pixel-level operator. We investigated our approach using SegNet and SCNN but it can be easily applied to any pair of semantic segmentation strategies.

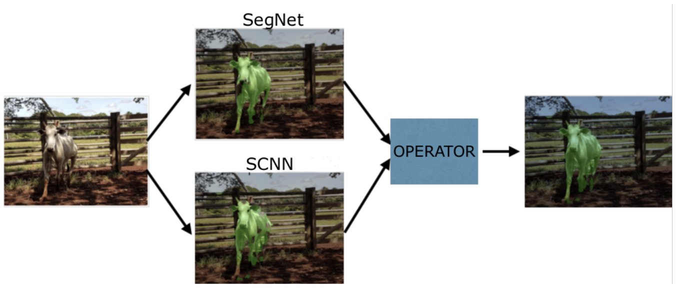


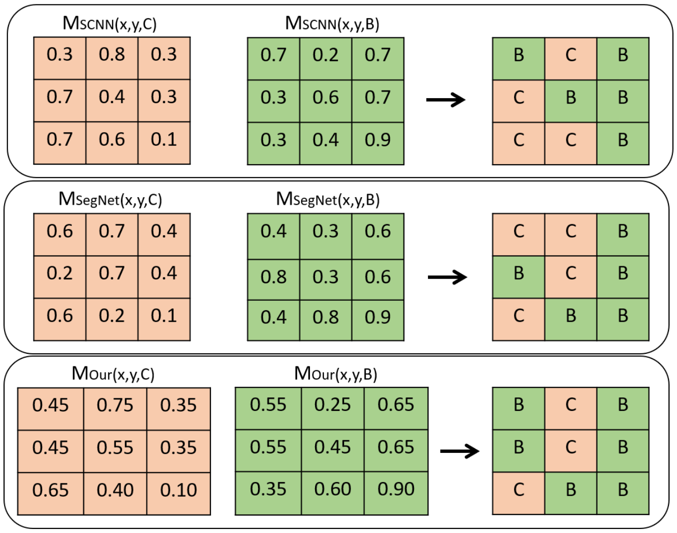
Fig. 2: Steps of the proposed approach for segmentation. First, an image is segmented through two separate techniques, and then their outputs are combined by means of a pixelwise operator to generate the final result.

Consider $M_{SCNN}(x, y, C)$ and $M_{SCNN}(x, y, B)$ as the probabilities of the pixel (x, y) belonging to the cattle and to the background, respectively, according to the SCNN. Similarly, consider $M_{SegNet}(x, y, C)$ and $M_{SegNet}(x, y, B)$ as the probabilities given by the SegNet. Given these probabilities, an operator is used to calculate new probabilities for each pixel. Following the convention above, let $M_{Our}(x, y, C)$ and $M_{Our}(x, y, B)$ be the probabilities given by the operation of the pixel belonging to cattle or to background, respectively. Five operators have been investigated: OR, AND, MAX, MULT and MEAN. Given the probabilities yielded by proceeding with the chosen operator, the pixel is considered to belong to the class associated with the highest probability. Next, we present each of the investigated operators.

The **MEAN operator** is defined in equations 1 and 2. This operator considers the certainties and uncertainties of each strategy in a weighted way. Figure 3 shows an example of the MEAN operator in action, applied on a 3x3 image. Given the probabilities calculated separately by the SCNN and the SegNet, the MEAN operator takes the mean between the probabilities for each class. As we can see, the result achieved through this strategy differs both from that of SCNN and from that of SegNet.

$$M_{Our}(x, y, C) = \frac{M_{SCNN}(x, y, C) + M_{SegNet}(x, y, C)}{2} \tag{1}$$

$$M_{Our}(x, y, 1) = \frac{M_{SCNN}(x, y, 1) + M_{SegNet}(x, y, 1)}{2} \tag{2}$$



MSCNN(x,y,C)

0.3	0.8	0.3
0.7	0.4	0.3
0.7	0.6	0.1

MSCNN(x,y,B)

0.7	0.2	0.7
0.3	0.6	0.7
0.3	0.4	0.9

B	C	B
C	B	B
C	C	B

MSegNet(x,y,C)

0.6	0.7	0.4
0.2	0.7	0.4
0.6	0.2	0.1

MSegNet(x,y,B)

0.4	0.3	0.6
0.8	0.3	0.6
0.4	0.8	0.9

C	C	B
B	C	B
C	B	B

MOur(x,y,C)

0.45	0.75	0.35
0.45	0.55	0.35
0.65	0.40	0.10

MOur(x,y,B)

0.55	0.25	0.65
0.55	0.45	0.65
0.35	0.60	0.90

B	C	B
B	C	B
C	B	B

Fig. 3: Each row shows the segmentation results for a 3x3 image, regarding SCNN, SegNet and our approach, respectively. In this case, the MEAN operator has been used. The first and second columns show the probabilities of a pixel being classified as cattle and background, respectively. The third column shows the final classification of the pixels, according to the applied strategy.

The **MULT operator** is defined in equations 3 and 4. This operator was inspired by the joint probability, considering the two independent strategies. The joint probability is used to observe simultaneous results of multiple variables, which in this case are the probabilities of the two strategies. Thus, this operator estimates the probability of simultaneous occurrence.

$$M_{Our}(x, y, C) = M_{SCNN}(x, y, C) * M_{SegNet}(x, y, C) \tag{3}$$

$$M_{Our}(x, y, B) = M_{SCNN}(x, y, B) * M_{SegNet}(x, y, B) \tag{4}$$

The **MAX operator** uses the maximum value between the probabilities given by the strategies. It is defined in equations 5 and 6. The main idea of this operator is to consider a strategy that provided a high probability (high chance of occurrence), even if the other strategy has a low probability. For example, consider that the strategies provide probabilities of 0.95 and 0.2 for the cattle (and 0.05 and 0.8 for the background, respectively). In this case, the probability of a pixel belonging to cattle using the MAX operator is 0.95, even though the second strategy yields that this probability is only 0.2. Therefore, this operator overlooks the lower probability, which does not occur with the previous operators, MEAN and MULT.

$$M_{Our}(x, y, C) = \max(M_{SCNN}(x, y, C), M_{SegNet}(x, y, C)) \tag{5}$$

$$M_{Our}(x, y, B) = \max(M_{SCNN}(x, y, B), M_{SegNet}(x, y, B)) \tag{6}$$

The **OR operator** is illustrated in Table I. This operator classifies a pixel as cattle if at least one of the strategies classified it as cattle. A pixel is classified as background if and only if both strategies classified it as background.

TABLE I: OR operation for pixels classified as cattle (C) and background (B).

SCNN	SegNet	OR
C	C	C
C	B	C
B	C	C
B	B	B

The **AND operator** classifies a pixel as cattle if and only if the two strategies classify it as cattle. Table II shows the logical AND operator that is applied to each pixel to obtain the final result of the segmentation process. The main difference between the AND and OR operators is in the preference for the cattle and background classes.

B. Image Dataset

The dataset used in the experiments consists of 154 images with resolution of 4032×3024 pixels. The images were taken from different angles and under different illumination conditions. Figure 4 presents four examples that illustrate the diversity and complexity of the dataset.

TABLE II: AND operation for pixels classified as cattle (C) and background (B).

SCNN	SegNet	AND
C	C	C
C	B	B
B	C	B
B	B	B

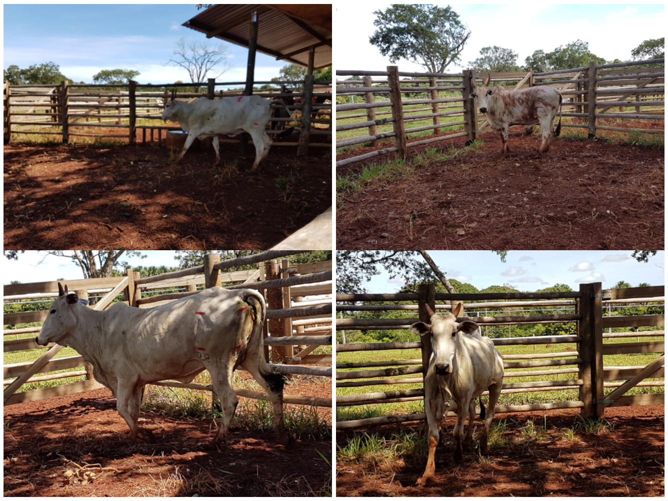


Fig. 4: Four of the 154 images present in the dataset used in the experiments. These images represent the diverse conditions with which a segmentation strategy is expected to deal.

All images were manually annotated using the software *Labelme*[1]. Figure 5 shows two images with their respective annotations.

C. Experimental Setup

The image dataset was randomly divided into training (60%), validation (20%) and test (20%) sets. No preprocessing technique other than resizing was applied to the images. The training and validation sets were used to train and obtain the best parameters for the neural networks used in each segmentation strategy.

For the SCNN and SegNet, the following hyperparameters were used:

- **SCNN:** the SLIC segmentation was evaluated using k values of 100, 500 and 1000 superpixels. The CNN weights were initialized using the pre-trained weights of the ImageNet (transfer learning). After some empirical experiments with the training and validation sets, a learning rate of 0.001 was chosen for ResNet and 0.0001 was used for the VGGs. The number of training epochs was set to 50. The batch size used was 32.
- **SegNet:** the SegNet encoder was also initialized using ImageNet pre-trained weights. On the other hand, the decoder was initialized with random weights. The learning rate was set to 0.01. The number of epochs was set to 150. For the SegNet, the batch size used was 16.

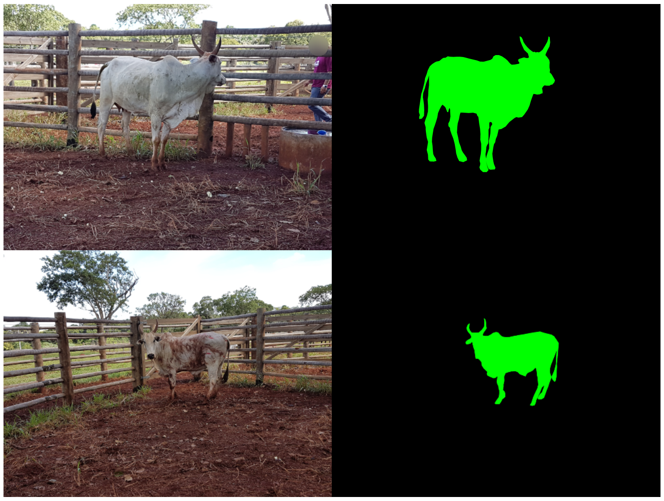


Fig. 5: Two of the images that were annotated manually for the experiments. The first column shows the original images. The second column shows the annotations. The green pixels in the annotations designate the cattle (foreground), while the black pixels indicate the background.

To monitor the training process, the cross-entropy loss function and the accuracy were used. To evaluate the segmentation results, two well-known metrics were used: pixel accuracy and intersection over union (IoU) [14], [15].

III. Results and Discussion

Table III shows the pixel accuracy and IoU for cattle segmentation using SCNN and SegNet separately. Initially, we evaluated the two strategies considering different backbones (VGG16, VGG19, and ResNet50) and k values (100, 500, and 1000) for the SCNN.

Using ResNet50, SCNN obtained superior results when compared to the variations of the VGG. This indicates that a deeper backbone is better for superpixel classification. In addition, the results achieved with $k = 500$ or $k = 1000$ were considerably higher than those for $k = 100$. For SegNet, using variations of VGG as backbone provided better results, either in accuracy or in IoU. Therefore, SegNet was able to classify each pixel even with a shallower backbone. A comparison of the two strategies shows that SegNet obtained better results, either in pixelwise accuracy or IoU. As the focus of our work is not the comparison of these techniques, but rather their integration in a better strategy, we shall not take much more time analyzing these results separately. Instead, let us focus on the strategy proposed by us.

Table IV presents the results of our approach using the five operators. The first two rows present the best results for SCNN and SegNet (see Table III). The OR operator with $k = 1000$ provided the best pixel accuracy for our approach (0.941), and also when the other strategies are considered (*i.e.*, pixel accuracy improved from 0.822 (SCNN) and 0.909 (SegNet) to 0.941 (Ours)). For IoU, the highest result achieved with the proposed operators was 0.815 using MAX, MULT and MEAN

[1] https://github.com/wkentaro/labelme

TABLE III: Comparison between segmentation strategies: SCNN and SegNet. The best observed values are in bold.

SCNN-k-backbone	Accuracy	IoU
SCNN-100-VGG16	0.654 (±0.17)	0.526 (±0.14)
SCNN-100-VGG19	0.544 (±0.23)	0.467 (±0.21)
SCNN-100-ResNet50	0.622 (±0.21)	0.508 (±0.18)
SCNN-500-VGG16	0.799 (±0.07)	0.700 (±0.09)
SCNN-500-VGG19	0.812 (±0.07)	0.706 (±0.08)
SCNN-500-ResNet50	**0.822 (±0.07)**	0.722 (±0.08)
SCNN-1000-VGG16	0.791 (±0.08)	0.709 (±0.09)
SCNN-1000-VGG19	0.777 (±0.08)	0.701 (±0.08)
SCNN-1000-ResNet50	0.821 (±0.07)	**0.736 (±0.07)**

SegNet-backbone	Accuracy	IoU
SegNet-VGG16	0.893 (±0.05)	**0.838 (±0.06)**
SegNet-VGG19	**0.909 (±0.04)**	0.835 (±0.07)
SegNet-ResNet50	0.874 (±0.04)	0.815 (±0.06)

with $k = 1000$. This result is slightly lower than that of SegNet (0.838), but it was higher than that of SCNN (0.736).

TABLE IV: Results of our approach using five operators. SCNN (ResNet50, $k = 1000$) and SegNet (VGG19) were used for comparison, as these configurations yielded the best results when the strategies were considered separately.

k	Strategy	Accuracy	IoU
	SCNN	0.822 (±0.07)	0.736 (±0.07)
	SegNet	**0.909** (±0.04)	**0.838** (±0.06)
100	OR	0.924 (±0.03)	0.774 (±0.09)
	AND	0.491 (±0.24)	0.482 (±0.24)
	MAX	0.688 (±0.15)	0.633 (±0.14)
	MULT	0.692 (±0.14)	0.638 (±0.13)
	MEAN	0.692 (±0.14)	0.508 (±0.13)
500	OR	0.939 (±0.02)	0.796 (±0.07)
	AND	0.750 (±0.10)	0.734 (±0.10)
	MAX	0.847 (±0.01)	0.779 (±0.07)
	MULT	0.848 (±0.06)	0.780 (±0.07)
	MEAN	0.848 (±0.06)	0.780 (±0.07)
1000	OR	**0.941** (±0.02)	0.799 (±0.08)
	AND	0.790 (±0.08)	0.773 (±0.08)
	MAX	0.878 (±0.04)	**0.815** (±0.05)
	MULT	0.879 (±0.04)	**0.815** (±0.05)
	MEAN	0.879 (±0.04)	**0.815** (±0.05)

For visual comparison, figure 6 shows some images with illustrative results of SegNet, SCNN and our approach using MEAN and OR operators. Inspection of these examples shows that our proposed method improves visual results. This improvement is more evident when the muzzle is to be segmented. In this case, both SegNet (Fig. 6.(a)) and SCNN (Fig. 6.(b)) provided inaccurate results. In addition, there are also flaws in some superpixels classified by SCNN, which can be explained by the lack of context. Accuracy can be improved by the proposed approach, as shown in Figure 6.(d), which corresponds to the OR combination. We can notice that the classification of the pixels belonging to the muzzle and the flaws of the SCNN were expressively improved by this combination.

IV. Conclusion

Livestock farming is one of the most important economic activities and the insertion of precision technologies in the field is an effective way of increasing productivity. For the development of such technologies, computer vision-based and artificial intelligence-based techniques are being increasingly researched, given their potential for the development of automated systems. Semantic segmentation of images is an important and complex step in many of these systems. Therefore, improving the performance of CV- and AI-based techniques for segmentation can improve the performance of the system as a whole.

With this in mind, in this work we proposed a novel technique to combine any two algorithms that perform semantic segmentation. We show that, in the task segmenting cattle in complex images, our technique improves the accuracy of what was here called SCNN (classification of superpixels with a CNN) and also of the SegNet (which classifies each pixel), when they are applied separately. These two techniques were chosen for being complementary to one another.

As stated, the proposed technique can be applied to any two pair of segmentation techniques (or even more). Each of the proposed operators has different characteristics and yields different results. Their potential is actually still unclear, and their application can be studied in diverse tasks. More importantly, it is expected that they shall improve the performance of any two complementary techniques, by making use of the strengths of each one to overcome their weaknesses. In our experiment, we showed that this is possible, by effectively doing it to segment cattle in complex images.

Acknowledgment

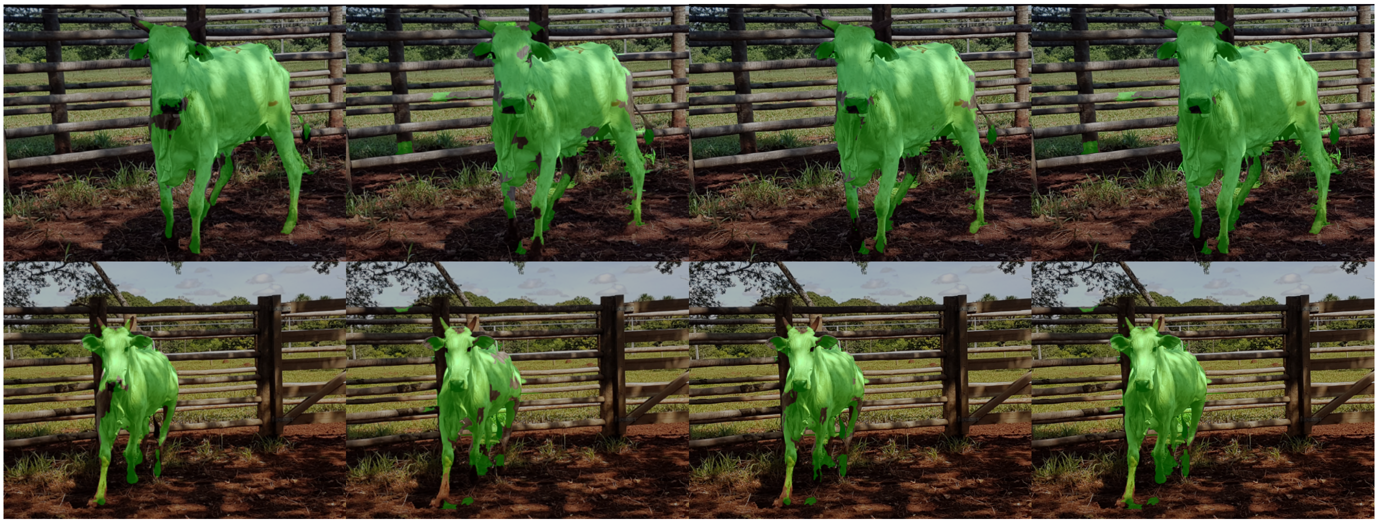
This work has received financial support from the Dom Bosco Catholic University and the Foundation for the Support and Development of Education, Science and Technology from the State of Mato Grosso do Sul, FUNDECT. Some of the authors have been awarded with Scholarships from the the Brazilian National Council of Technological and Scientific Development, CNPq and the Coordination for the Improvement of Higher Education Personnel, CAPES. We would also like to thank NVIDIA for providing the Titan X GPUs used in the experiments.

References

[1] B. Xu, W. Wang, G. Falzon, P. Kwan, L. Guo, G. Chen, A. Tait, and D. Schneider, "Automated cattle counting using Mask R-CNN in quadcopter vision system," *COMPUTERS AND ELECTRONICS IN AGRICULTURE*, vol. 171, APR 2020.

[2] J. Salau and J. Krieter, "Instance Segmentation with Mask R-CNN Applied to Loose-Housed Dairy Cows in a Multi-Camera Setting," *ANIMALS*, vol. 10, no. 12, DEC 2020.

[3] R.-W. Bello, A. S. A. Mohamed, and A. Z. Talib, "Contour extraction of individual cattle from an image using enhanced mask r-cnn instance segmentation method," *IEEE Access*, vol. 9, pp. 56 984–57 000, 2021.



(a) SegNet

(b) SCNN

(c) MEAN

(d) OR

Fig. 6: Comparative results of SegNet (VGG19), SCNN (ResNet50, $k = 1000$) and the combinations MEAN and OR in two images of the standing ox base.

[4] ——, “Enhanced mask r-cnn for herd segmentation,” *International Journal of Agricultural and Biological Engineering*, vol. 14, no. 4, pp. 238–244, 2021.

[5] C. Chen, W. Zhu, and T. Norton, “Behaviour recognition of pigs and cattle: Journey from computer vision to deep learning,” *Computers and Electronics in Agriculture*, vol. 187, p. 106255, 2021.

[6] T. Deng, B. Fu, M. Liu, H. He, D. Fan, L. Li, L. Huang, and E. Gao, “Comparison of multi-class and fusion of multiple single-class segnet model for mapping karst wetland vegetation using uav images,” *Scientific Reports*, vol. 12, no. 1, 2022.

[7] S. Son, S.-H. Lee, J. Bae, M. Ryu, D. Lee, S.-R. Park, D. Seo, and J. Kim, “Land-cover-change detection with aerial orthoimagery using segnet-based semantic segmentation in namyangju city, south korea,” *Sustainability (Switzerland)*, vol. 14, no. 19, 2022.

[8] P. Maheswari, P. Raja, and V. T. Hoang, “Intelligent yield estimation for tomato crop using segnet with vgg19 architecture,” *Scientific Reports*, vol. 12, no. 1, 2022.

[9] N. Dhingra, G. Chogovadze, and A. M. Kunz, “Border-seggcn: Improving semantic segmentation by refining the border outline using graph convolutional network,” in *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021*. IEEE, 2021, pp. 865–875. [Online]. Available: https://doi.org/10.1109/ICCVW54120.2021.00102

[10] Z. Chen, B. Guo, C. Lib, and H. Liu, “Review on superpixel generation algorithms based on clustering,” 2020, Conference paper, p. 532 – 537, cited by: 4. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85096351291&doi=10.1109%2fICISCAE51034.2020.9236851&partnerID=40&md5=18bc443d589c1200e4e38e2fdbbafb3f

[11] J.-H. Witte, J. Gerberding, C. Melching, and J. M. Gómez, “Evaluation of deep learning instance segmentation models for pig precision livestock farming,” in *Business Information Systems*, 2021, pp. 209–220.

[12] D. A. Sant’Ana, M. C. B. Pache, J. Martins, G. Astolfi, W. P. Soares, S. L. N. de Melo, N. da Silva Heimbach, V. A. de Moraes Weber, R. G. Mateus, and H. Pistori, “Computer vision system for superpixel classification and segmentation of sheep,” *Ecological Informatics*, vol. 68, p. 101551, 2022.

[13] A. D. S. Ferreira, D. M. Freitas, G. G. da Silva, H. Pistori, and M. T. Folhes, “Weed detection in soybean crops using convnets,” *Computers and Electronics in Agriculture*, vol. 143, pp. 314 – 324, 2017.

[14] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[15] E. Shelhamer, J. Long, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.