Revista de Informática Teórica e Aplicada - RITA - ISSN 2175-2745
Vol. 32, Num. 1 (2025) 243-249

RESEARCH ARTICLE

Deep Semantic Segmentation Applied to Honey Bee Comb Cells

Segmentação Semântica Profunda Aplicada a Células de Favos de Abelhas

Gustavo Vasconcelos[1], André Roberto Ortoncelli[1], Fabiana Martins Costa[1], Marlon Marcon[1]*

Abstract: Machine learning is increasingly prevalent in various fields, including beekeeping, where it has been applied to honeycomb analysis through Semantic Segmentation of images. This study evaluates different neural network architectures and encoders using the Intersection over Union (IoU) method to segment Apis mellifera honeycombs. A comparative analysis of semantic segmentation techniques was conducted, revealing advancements in the field with the SegNet architecture combined with the MobileNet encoder for feature extraction. We present a new dataset of 10 comb cell images, which we combine with the images from the DeepBee dataset in our experiments. Our results surpass the state-of-the-art results for comb cell segmentation in the DeepBee dataset.
Keywords: Semantic Segmentation — Honeycombs — Convolutional Neural Network — Encoders

Resumo: O aprendizado de máquina está se tornando cada vez mais comum em vários campos, incluindo a apicultura, onde tem sido aplicado à análise de favos de mel por meio da Segmentação Semântica de imagens. Este estudo avalia diferentes arquiteturas de redes neurais e codificadores utilizando o método de Interseção sobre União (IoU) para segmentar favos de mel de Apis mellifera. Foi realizada uma análise comparativa das técnicas de segmentação semântica, revelando avanços na área com a arquitetura SegNet combinada com o codificador MobileNet para extração de características. Apresentamos um novo conjunto de dados com 10 imagens de células de favo, que combinamos com as imagens do conjunto de dados DeepBee em nossos experimentos. Nossos resultados superam os melhores resultados anteriores para segmentação de células de favo no conjunto de dados DeepBee.
Palavras-Chave: Segmentação Semântica — Favos de Mel — Rede Neural Convolucional — Codificadores

[1] *Federal University of Technology - Paraná (UTFPR), Brazil*
*__Corresponding author__: marlonmarcon@utfpr.edu.br
DOI: http://dx.doi.org/10.22456/2175-2745.143318 • **Received:** 23/10/2024 • **Accepted:** 03/01/2025

1. Introduction

Beekeeping is widely practiced worldwide, providing raw materials such as honey, pollen, propolis, wax, apitoxin, and royal jelly. Additionally, it is an ecologically beneficial practice, as bees play a crucial role in cross-pollination. This process promotes the evolutionary adaptation of plants by increasing species vigor, enabling new genetic combinations, and enhancing the production of fruits and seeds, which are responsible for the fertilization of 73% of the world's flora [1].

Apiculture refers to the keeping of bees of the species *Apis mellifera*, a social bee of European origin belonging to the family Apidae and the Hymenoptera order. This species was introduced to the Americas by the English and Spanish, and in Brazil, it was introduced in 1839 to meet the demand for honey and wax in apiaries. The most commonly used hive for keeping these bees is the Langstroth hive, designed by American beekeeper Lorenzo Langstroth. The main advantage of this hive is the ease with which the honeycombs are built in frames, allowing them to be easily moved by the beekeeper.

Success in apiculture depends on beekeepers replacing their queens annually [2]. To maintain such success, it is necessary to analyze the egg-laying, monitoring the number of eggs laid, the arrangement of the egg-laying (starting from the central frames and moving towards the lateral ones), and the amount of honey produced concerning the available comb area. For this, usually, it is necessary to interrupt the bees' natural routine, remove them from the frames, and photograph the hive to analyze its evolution, estimate the frame areas, and evaluate the hive's health.

After capturing photos of the current state of the frames, a manual analysis of the images calculates the total area occupied by the hive on the frames. It distinguishes areas occu-

pied by brood combs, honeycombs, and empty combs. This analysis verifies whether a queen is healthy, whether she is laying eggs according to control parameters, whether brood and honey production is adequate, or whether it is necessary to adopt a new diet for the hive or even replace the queen. This manual process requires significant time and effort from the responsible professional, demanding detailed attention to many images.

Using Computer Vision in analyzing bee frames allows for a faster, more accurate, and less invasive evaluation of hive health compared to traditional manual methods [3]. The adoption of advanced techniques, such as semantic image segmentation, provides significant support in efficiently identifying and classifying the different areas of the frames, such as brood combs, honeycombs, and empty areas [4]. This approach improves the accuracy of the analysis and enables continuous and automated monitoring of the hives, contributing to more sustainable and productive beekeeping.

In this paper, we support modernizing and making the analysis of the frames more efficient, reducing the impact on the bees' routine and providing better and faster results than manual analysis. Previous works, such as Alves et al. [4], address the comb segmentation and cell classification problems with significant results but require many preprocessing stages. The main contributions of this work are 1) A new dataset of *Apis Mellifera* combs; 2) A comprehensive evaluation of end-to-end deep-learning-based approaches to semantic segmentation applied to Honey Bee Combs; 3) An update to the state-of-the-art results on the DeepBee dataset, proposed by Alves et al. [4]. The source code used in the experiments and the public dataset is available at the link: ⟨**https://github.com/ICDI/combcells-segmentation**⟩.

With these objectives in mind, the remainder of this work is organized as follows: Section 2 presents a related works; Theoretical aspects necessary for understanding the work are in Section 3; The methodology used is presented in Section 4; Section presents the experimental results; Finally, concluding remarks and future work are in Section 6.

2. Related Works

Image segmentation has been crucial in Computer Vision since its early days. As an essential component of many visual knowledge systems, it involves partitioning images (or video frames) into various segments and objects, holding central importance in a wide range of applications, including medical image analysis (such as tumor delimitation and tissue volume measurement), autonomous vehicles (e.g., detection of navigable surfaces and pedestrians), video surveillance, and augmented reality [5].

Segmentation techniques are also used in the detection and classification of honey bee comb cells, as in the work of Alves et al. [4], in which the authors adapted the UNet [6] solution to make it more efficient in detecting the frame area, using the CNN architecture. An example of the Comb Cells segmented by Alves et al. is presented in Figure 1. The input images were normalized by dividing each pixel by 255, thus separating the object of interest into white pixels and the image background into black pixels. As hyperparameters, the authors adopted Binary cross-entropy as the loss function, He Normal initialization [7] for weights, and a Sigmoid function as the output layer to transform the output into a binary image by applying a threshold of 0.5 [4].

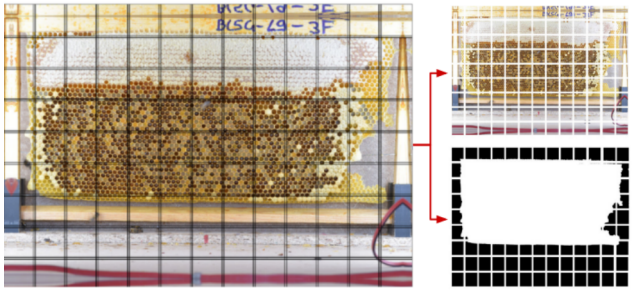


Figure 1. Blocks created from the original images and their labels. Source: Alves et al. [4]

Previously, in classifying cells, Alves et al. proposed a semantic-segmentation-based approach to select the interest area of images. They adopted an architecture based on the UNet [6] with an encoder MobileNet v2 [8], shown in Figure 2. This study provides a comprehensive evaluation of the performance of detecting the brood frame area in other CNN architectures.

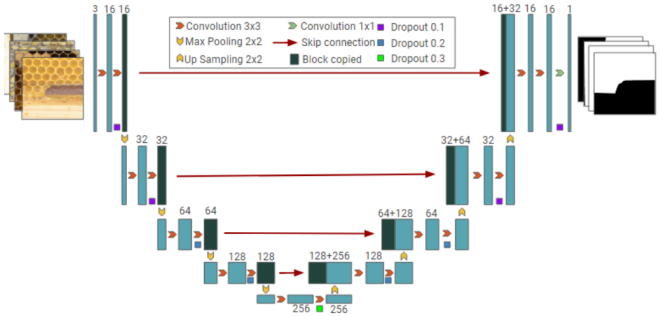


Figure 2. UNet Architecture for Honey Bee Comb Cells segmentation. Source: Alves et al. [4]

The work of Alves et al. [4] serves as a foundation for subsequent research in the segmentation and classification of honeybee comb cells. Building on these advancements, this study explores alternative CNN architectures, assessing their performance in the same context.

3. Theoretical Aspects

In this section, we delve into the theoretical foundations underlying the key concepts and architectures utilized in computer vision, particularly in image segmentation. We begin by exploring the role of encoders and decoders, which form the backbone of Convolutional Neural Networks (CNNs) by

converting input data into feature representations and then reconstructing the output. Following this, we examine semantic segmentation algorithms, focusing on fully convolutional networks (FCNs), UNet, SegNet, and PSPNet, which are crucial for pixel-level classification tasks. These theoretical aspects provide a comprehensive framework for understanding the advanced methodologies employed in the case studies and applications that follow.

3.1 Encoders/Decoders

Encoders and decoders are the primary operations used by CNNs to execute their training. The encoder network maps a given input signal $x \in X \subset R^{d0}$ to a feature space $z \in Z \subset R^{dk}$, while the decoder takes this feature map as an input, processes it, and produces an output $y \in Y \subset R^{dL}$. This paper considers a symmetric configuration so that the encoder and the decoder have the same layers and the input and output dimensions are symmetrical, as shown in Figure 3.

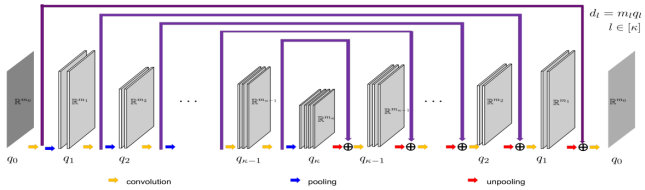


Figure 3. Encoder-Decoder symmetrical CNN architecture [6].

Different feature extraction methods will be used in this case study, employing different types of encoders and the standard ones for each CNN. These include:

1. *Visual Geometry Group* VGG: proposed by Simonyan *et al.* [9], VGG is a deep neural network architecture known for its simplicity and performance in image classification tasks, such as the well-known ImageNet challenge. The model uses multiple stacked convolutional layers with small 3x3 filters, followed by pooling layers to reduce spatial dimensionality. The innovation of VGG lies in the network's depth, with versions containing up to 19 layers, allowing for a greater capacity to extract complex features. The simplicity of its building blocks, using only convolutions and pooling, makes it easy to understand and implement, making it a reference for more modern networks.

2. Resnet [10]: is an innovative architecture that introduced the concept of residual connections, allowing deeper network layers to learn more efficiently. The main challenge in very deep networks is the vanishing gradient problem, which hampers learning. Residual connections solve this issue by creating shortcuts between layers, allowing gradients to flow directly through these connections. This enables the construction of extremely deep networks, such as ResNet-50 or ResNet-101, which achieved remarkable results in computer vision tasks without suffering from performance degradation in very deep networks.

3. MobileNet [11, 8]: is an architecture designed for mobile devices and environments with limited computational resources. Unlike traditional deep networks, it uses depthwise separable convolutions to drastically reduce computational complexity and the number of parameters without renouncing much accuracy. The authors proposed two convolutional stages: depthwise convolution, which filters data by channel, and pointwise convolution, which combines these channels. This efficiency makes MobileNet ideal for real-time computer vision applications, such as object detection on smartphones and drones, where processing power is limited.

   Designed by Google researchers, MobileNets [11, 8] aims to facilitate the use of resources on mobile devices, where part of the information is processed in the cloud, thus saving computational resources on the device. The accuracy level of a MobileNet is lower than traditional neural networks to speed up processing time. This is compensated by the number of samples MobileNet has in the cloud.

3.2 Semantic Segmentation Algorithms

Semantic segmentation consists of assigning each pixel in an image a label or class that describes a real-world object. This process involves two approaches: supervised classification, where the user identifies some pixels belonging to the desired classes to form a training area. Based on a supervised model, classify all pixels belonging to those classes according to a predefined statistical rule. The second approach is called unsupervised classification, in which the software decides through clustering analysis which classes to separate and which pixels belong to each of those classes [12]. Supervised approaches will be considered for the context of this work, as presented in the following subsections.

3.2.1 Fully Convolutional Networks (FCN)

Typical convolutional neural networks (CNNs) are not fully convolutional because they often contain fully connected layers (which do not perform the convolution operation). These layers are rich in parameters like filters and other similar operations. However, FCNs are neural networks that perform only convolutional operations (subsampling or upsampling). An FCN is a CNN without fully connected layers.

The primary motivation behind this CNN is to build "fully convolutional" networks that can receive inputs of arbitrary size and produce corresponding output sizes, learning through efficient inference [13].

The architecture of [13] was one of the first works to train FCNs end-to-end for pixel-by-pixel prediction using supervised pre-training. Both learning and inference are performed on the entire image simultaneously through feedforward computation and backpropagation. The upsampling layers in the

network enable pixel-by-pixel prediction and learning in networks with subsampled pooling. A visual representation of this architecture can be seen in Figure 4.

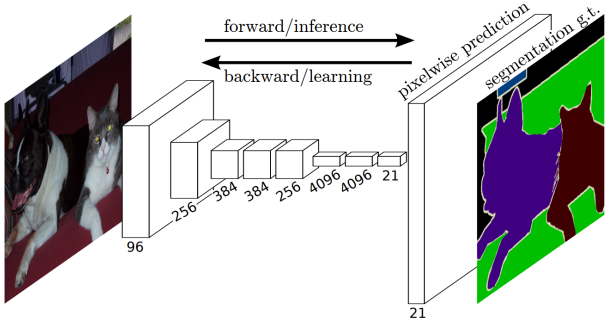


Figure 4. FCN architecture [13]

3.2.2 UNet

UNet is a neural network architecture primarily designed for image segmentation. The basic structure of a UNet architecture consists of two paths. The first path is the encoder, similar to a regular convolutional network, and provides classification information. The second is the decoder, consisting of convolutions and concatenations with features from the encoder [14].

Semantic segmentation requires pixel-level discrimination and a mechanism to project the learned discriminative features at different stages of the encoder back into the pixel space. Additionally, the expansive path increases the output resolution, which can then be passed to a final convolutional layer to create an entirely segmented image. The resulting network architecture is almost symmetrical, giving it a U-shape [14], A visual example of this can be seen in Figure 5

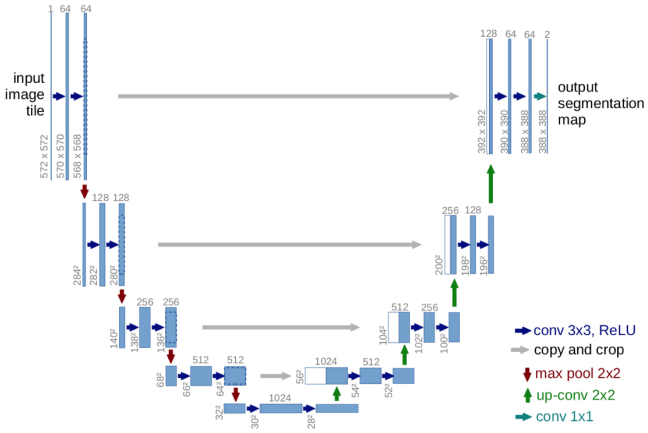


Figure 5. Unet architecture [6]

The encoder is the first half of the architecture diagram (Figure 5). It is typically a pre-trained classification network like VGG/ResNet, applying convolution blocks and max pooling to encode the input image into feature representations at various levels.

The decoder is the second half of the architecture, aiming to semantically project the discriminative features (lower resolution) learned by the encoder into the pixel space (higher resolution) to obtain dense classification. The decoder consists of upsampling and concatenation, followed by regular convolutional operations.

3.2.3 SegNet

SegNet is an architecture for pixel-wise semantic segmentation, motivated primarily by road scene understanding applications that require the ability to model appearance (road, building), shape (cars, pedestrians), and understand the spatial relationship (context) between different classes such as roads and sidewalks [15].

The encoder network's architecture is topologically identical to the 13 convolutional layers in the VGG network. The role of the decoder network is to map the low-resolution feature maps from the encoder to full input-resolution feature maps for pixel-wise classification. The novelty of SegNet lies in how the decoder upsamples its feature map(s). Specifically, the decoder uses pooling indices to perform non-linear upsampling, eliminating the need to learn to upsample [15]. A visual representation of this encoder-decoder architecture can be seen in Figure 6

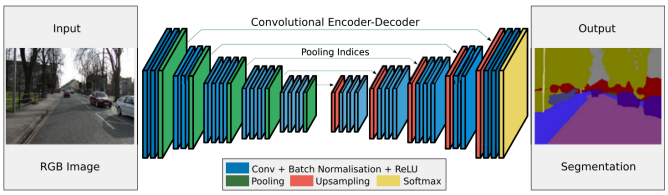


Figure 6. Segnet architecture [15]

3.2.4 PSPnet

Unlike some methods that incorporate global features other CNNs use, PSPNet proposes a pyramid scene parsing network. In addition to using a pre-trained ResNet model with the traditional dilated network strategy [16], [17] for pixel prediction, it extends the pixel-level feature into a specially designed global pyramid, making both local and global cues jointly improve the final prediction reliability [**??**].

As shown in Figure 7, PSPNet utilizes multi-scale contextual information to effectively distinguish between focused and blurred areas. At the same time, a convolutional conditional random field (ConvCRF) is added to the processing procedure to effectively optimize the network's prediction probability graph, helping to improve segmentation results and enhance the blending effect.

4. Metodology

A dataset is required for machine learning to train and evaluate the CNN architectures we used. In this context, we adopted the images in the project documentation by [4]. Out of the 60 pre-segmented images found in this project, 48 were selected

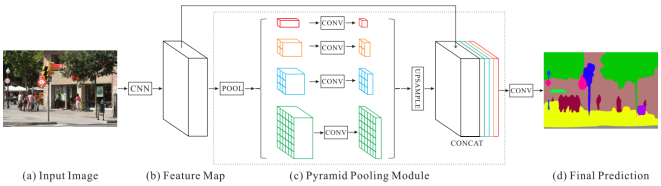
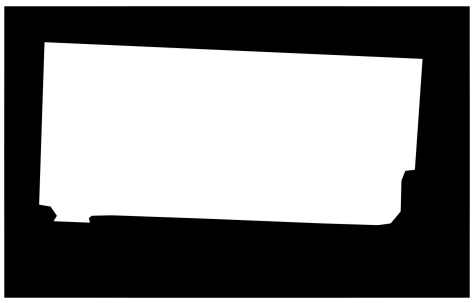


Figure 7. PSPNet architecture [**??**].

for training and 12 for testing. Additionally, ten other images were provided by the unepe, which were manually segmented to obtain their masks. An example of this segmentation can be seen in Figure 8. These additional images were used to test in a transfer learning scenario, i.e., to evaluate images of frames that are in different conditions than those used in training.

(a) Beehive frames



(b) Segmentation mask

Figure 8. Example of a segmented frame with the respective annotated mask.

4.1 Training

Once all the images were segmented, the next step was training the algorithms. A public GitHub repository[1] developed in Python was used, which implements all the neural network architectures listed below:

1. **Unet**: Original, encoder VGG, ResNet50, MobileNet;
2. **Segnet**: Original, encoder VGG, ResNet50, MobileNet;
3. **PSPnet**: Original, encoder VGG, ResNet50;
4. **FCN8**: Original, encoder VGG.

[1] https://github.com/divamgupta/image-segmentation-keras

We trained the networks through a Python Google Colab environment with GPU enabled. Initially, the images, both for training and testing, were resized so that all would have the same dimensions and meet the requirements of certain FCNs, such as the PSPnet, which requires that the image dimensions be multiples of 192. Therefore, the first step executed in the notebooks during the development of the work was the resizing of the training and testing images to a ratio of 576x384. After resizing all images, the training phase began. To execute the model, some standard training parameters were required, which are as follows:

1. **Image Path**: Path to the training images;
2. **Mask Path**: Path to the segmentation masks of the training images;
3. **Number of Classes**: Since this research only aims to extract the bee frame area from the background of the image, it has only two classes;
4. **Epochs**: Set by default to 50 for all networks.
5. **Batch Size**: Set to 2 images per epoch.

4.2 Evaluation Criteria

The Intersection over Union (IoU) metric, also referred to as the Jaccard Index, was used to assess the accuracy of the algorithms (Figure 9). This is a widely used evaluation metric to measure the accuracy of image segmentation models that compares the elements of two sets to see which members are shared and which are distinct. It is a similarity measure for the two data sets, ranging from 0% to 100%, and the higher the percentage, the more similar the two populations are [18].

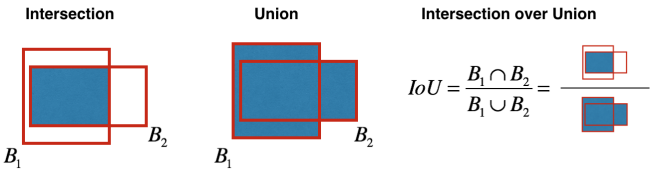
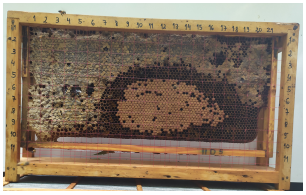


Figure 9. Intersection over Union

After evaluating all the test images selected from the DeepBee research dataset [4], the ability of the trained models to transfer knowledge to another dataset, with images collected at UTFPR, was analyzed. The UTFPR-DV dataset consists of 10 comb images labeled as presented in Figure 8.

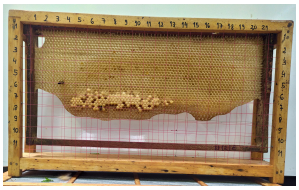
From Figures 10, it is possible to verify that the images from the DeepBee and UTFPR-DV datasets present different conditions, which may represent variations in the results.

As we can observe, there is a visible difference between the types of images used, with the original DeepBee images having better camera positioning, lighting, and, consequently, sharpness. Therefore, it is essential to evaluate two different types of image datasets so that during the verification and evaluation stage, it is possible to observe how each model behaves with images that present various challenges and circumstances.

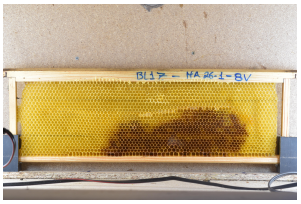


(a) Frame 11D5 UTFPR-DV

(b) Frame 1049 DeepBee



(c) Frame 11e2 UTFPR-DV



(d) Frame 1051 DeepBee

Figure 10. Comparison between images from DeepBee x UTFPR-DV datasets

5. Results

Upon completion of the tests, the following results were obtained based on an evaluation of the IoU percentage.

Table 1. IoU Results for DeepBee Models.

	UNET	SEGNET	PSPNET	FNC_8
ORIGINAL	97,42%	93,16%	95,19%	96,45%
VGG	97,51%	93,16%	89,93%	97,65%
RESNET50	96,99%	95,06%	86,76%	-
MOBILENET	95,70%	97,70%	-	95,67%

As observed in Table 1, which presents the efficiency of the semantic segmentation models tested on the DeepBee dataset, the SegNet CNN with the MobileNet encoder demonstrated the best performance (97.70%) compared to the other models. In Alves et al. [4], the UNET architecture with the MobileNet encoder proved more efficient. In our experiments, such architecture presented an IoU of 95,70%. The other architectures, changing the encoder, improved from 1 to 2% in absolute values and presented 97,70% IoU performance on the SegNet + MobileNet combination. Such results highlight the need for constant evolution in deep learning-based technologies.

Analyzing the transfer learning results on the UTPFR-DV dataset (Table 1), the choice of the SegNet + MobileNet architecture stands out even more, representing a gain of more than three percentage points compared to the model in [4].

Table 2. IoU Results for UTFPR-DV Dataset Images

	UNET	SEGNET	PSPNET	FNC_8
ORIGINAL	71,71%	81,64%	73,32%	79,77%
VGG	81,53%	83,78%	77,39%	78,46%
RESNET50	80,27%	78,10%	63,57%	-
MOBILENET	83,60%	86,65%	-	80,59%

After identifying the best-performing model among those trained, it is necessary to evaluate its effectiveness compared to the original model used by [4]. For this, the MobileNet + SegNet model was trained with the same parameters used by the author in their algorithm, i.e., 224x224 images with 40 images for training and the rest for testing in 50 generations. After this analysis, we obtained the following results.

Regarding training evaluations, the MobileNet network from [4] presented an accuracy of 93.31%. Unfortunately, as we cannot know which images were used for training and testing, it is uncertain that we can obtain the current model's accuracy concerning the paper results.

6. Conclusions and Future Works

After this research, it can be stated that the semantic segmentation algorithm that best classified images of *Apis Mellifera* bee hive frames among those trained was SegNet using the MobileNet feature extraction method. Among the CNNs studied, this model showed the highest accuracy in the IoU evaluation when analyzing the images from the original dataset used by [4] and when evaluating images from the UTFPR-DV dataset.

Feature extraction with VGG proved quite effective, ranking only behind the algorithms that used MobileNet. Furthermore, after comparing SegNet with MobileNet to the model originally used in [4]'s research, it can be concluded that, based on the loss and accuracy statistics of the best generations of both models, SegNet with MobileNet proved to be superior to the employed initially MobileNet model. This result suggests replacing the original proposal of [4] (UNet + MobileNet) with the SegNet + MobileNet combination.

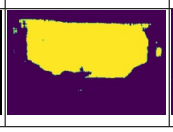
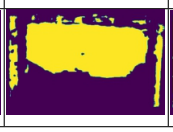
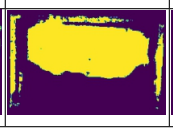
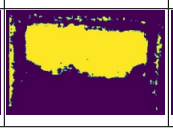
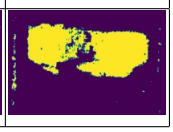
This outcome is likely because SegNet, designed initially to recognize roads, is more familiar with recognizing objects that occupy a large portion of the screen. Although MobileNet is inferior in processing power compared to other encoders, as also observed in [4]'s work, it proved to be the most effective method for extracting features from *Apis Mellifera* brood frames.

As future work, the extension of the UTFPR dataset is planned, increasing the quantity and variability of images and the provision of annotations at the honey bee comb cells classification level. This would provide a significant update to [4] work, enabling an end-to-end strategy for comb image classification.

Author contributions

Gustavo Vasconcelos: Conceptualization, database preparation, software implementation, execution of experiments. André Roberto Ortoncelli: Software testing and writing of the article. Fabiana Martins Costa: Conceptualization, preparation of the database, and review of the article. Marlon Marcon: Conceptualization, software implementation, software, execution of experiments, and writing of the article. All authors read and approved the final version of the text.

Table 3. Comparison of Results from the Top 5 CNNs.

	Original	Ground Truth	SEGNET + MOBILENET	FCN8 + VGG	UNET	UNET + VGG	UNET + MOBILENET
1060 deepbee							
1049 deepbee							
11D2 UTFPR-DV							

References

[1] COUTO, R. H. N.; COUTO, L. A. *Apicultura: manejo e produtos*. [S.l.]: Funep Jaboticabal, 2006.

[2] CHAVES, J. da S. et al. Produção de abelhas rainhas africanizadas apis mellifera l. pelo método de puxada artificial. *Brazilian Journal of Development*, v. 6, n. 10, p. 80839–80847, 2020. Disponível em: ⟨https://doi.org/10.34117/bjdv6n10-489⟩.

[3] BILIK, S. et al. Machine learning and computer vision techniques in continuous beehive monitoring applications: A survey. *Computers and Electronics in Agriculture*, Elsevier, v. 217, p. 108560, 2024. Disponível em: ⟨https://doi.org/10.1016/j.compag.2023.108560⟩.

[4] ALVES, T. S. et al. Automatic detection and classification of honey bee comb cells using deep learning. *Computers and Electronics in Agriculture*, Elsevier, v. 170, p. 105244, 2020. Disponível em: ⟨https://doi.org/10.1016/j.compag.2020.105244⟩.

[5] MINAEE, S. et al. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1–1, 2021. Disponível em: ⟨https://doi.org/10.1109/TPAMI.2021.3059968⟩.

[6] RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. *Medical Image Computing and Computer-Assisted Intervention*. 2015. p. 234–241. Disponível em: ⟨https://doi.org/10.1007/978-3-319-24574-4_28⟩.

[7] HE, K. et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In: *IEEE International Conference on Computer Vision*. [s.n.], 2015. Disponível em: ⟨https://doi.org/10.1109/ICCV.2015.12⟩.

[8] SANDLER, M. et al. Mobilenetv2: Inverted residuals and linear bottlenecks. In: *IEEE Conference on Computer Vision and Pattern Recognition*. [s.n.], 2018. p. 4510–4520. Disponível em: ⟨https://doi.org/10.1109/CVPR.2018.00474⟩.

[9] SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[10] HE, K. et al. Deep residual learning for image recognition. In: *IEEE Conference on Computer vision and Pattern Recognition*. [s.n.], 2016. p. 770–778. Disponível em: ⟨https://doi.org/10.1109/CVPR.2016.90⟩.

[11] HOWARD, A. G. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[12] CROSTA, A. P. *Processamento digital de imagens de sensoriamento remoto*. [S.l.]: UNICAMP/Instituto de Geociências, 1999.

[13] LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. In: *IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2015. p. 3431–3440.

[14] SIDDIQUE, N. et al. U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, v. 9, p. 82031–82057, 2021. Disponível em: ⟨https://doi.org/10.1109/ACCESS.2021.3086020⟩.

[15] BADRINARAYANAN, V.; KENDALL, A.; CIPOLLA, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 39, n. 12, p. 2481–2495, 2017. Disponível em: ⟨https://doi.org/10.1109/TPAMI.2016.2644615⟩.

[16] YU, F.; KOLTUN, V. *Multi-Scale Context Aggregation by Dilated Convolutions*. arXiv, 2015. Disponível em: ⟨https://arxiv.org/abs/1511.07122⟩.

[17] CHEN, L.-C. et al. *Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs*. arXiv, 2014. Disponível em: ⟨https://arxiv.org/abs/1412.7062⟩.

[18] STEPHANIE. *Jaccard index / similarity coefficient*. 2020. Disponível em: ⟨https://www.statisticshowto.com/jaccard-index/⟩.