Vision Pro: WildScenes Image Segmentation

I. Introduction

The goal of the current project is to develop an optimal method to perform image segmentation of the Wildscenes data set [1]. The WildScenes data set contains 9306 2-dimensional images, collected from nature. Alongside the raw images, the data set also contains two versions of each image that have been segmented and classified into 15 classes, one in a grayscale colour range with pixel intensity 0-14, and another in a human readable format with bright/distinct colours to differentiate classes.

The segmented images contain a class imbalance. As set out in Fig 1 below, different classes are not represented equally in all images (for example, there are many more pixel-wise samples of tree-foliage over rocks).

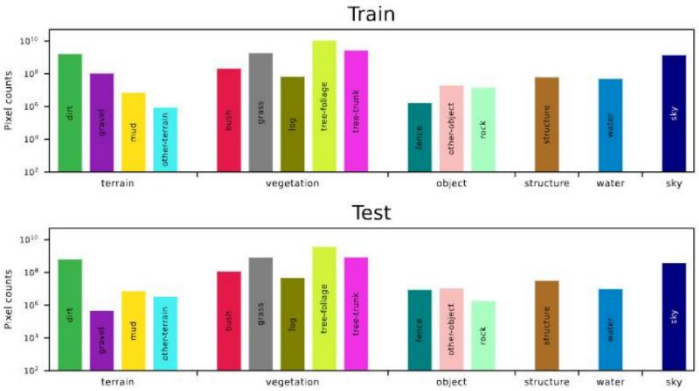


Fig 1 - Class Imbalance in WildScenes Data Set

II. Literature Review

Image segmentation in the literature is a well-described problem, with no shortage of papers and journals discussing the topics. Our team divided our literature research methods into two key areas of focus: deep learning-centred methods, and traditional image segmentation methods.

Literature databases such as UNSW library [2], Springer Link [3] and Papers With Code [4] were searched with relevant keywords including generic terms e.g. “image segmentation methods”, specific methods “deep learning image segmentation”. Additionally, searches were also run including the words “nature” or “natural” in order to find methods that had been used for image segmentation in natural environments. Due to the large amount of existing literature, focus was particularly given to papers that discussed image segmentation methods for autonomous driving, or on methods for image segmentation of natural images. Studies that were from the last 5 years were also prioritised over older research, due to the quickly advancing nature of this field.

Interestingly, there were limited resources dedicated directly to this specific use case (image segmentation of noisy natural image scenes) with most natural image papers focused on plant identification and involving images such as a single plant or flower against a clear background. Autonomous-driving problems typically contained images devoid of many natural features, as expected. Studies focusing on medical image problems were considered moderately useful, but only where they dealt with noisy images, and images with a large range of colour variation.

A. Existing Literature - Deep Learning Methods

In the past 5 years, the majority of papers published in relation to image segmentation used a deep learning method, particularly a Convolutional Neural Network (CNN). For this project, a key challenge was determining which form of Deep Learning would be the most effective. Through research on deep learning methods for image segmentation applied to either autonomous driving tasks or to natural images, CNNs (and in particular, the subset of Fully Convolutional Networks - FCNs) appeared many times with promising results. In the autonomous driving/scene analysis space, Sun and Wang (2023) [5] used FCNs (particularly HRNet and PSPNet) for remote sensing images (e.g. satellite data).

In the agricultural/natural image processing space, multiple studies also had good results with CNNs and FCNs. Lei, L., Yang, Q., Yang, L. et al.(2024) [6] compare many types of deep learning methods for image segmentation of natural images in various applications such as identifying plant species, counting wheat bundles, and weed identification. The paper considers CNNs, Generative Adversarial Networks, Graph Neural Networks and Transformer Models. They found that Convolutional Neural Networks performed well on relevant problems to this project, including image segmentation of corn in fields. Transformers and Generative Adversarial Networks also performed well. U-Net (a specific CNN) was found to be good at training with limited samples and simplistic architecture, and is extensively employed in weed identification tasks in agricultural contexts.

Shao et al. (2021) [7] proposed a hybrid approach that incorporated FCNs, founded on transfer learning and the watershed algorithm. This approach performed very well for an application that had to count the number of rice ears in a field. Wang et al (2022) [8] introduced a method for accurately counting wheat ears in field conditions using FCN and Harris corner detection.

B. *Existing Literature - Traditional Image Segmentation Methods*

Outside of deep learning approaches to image segmentation, Active Contour Models (ACMs) were widely used in recent literature for image segmentation, although the majority of use-cases are in medical imaging. Fang, L., Liang, X., Xu, C. et al (2024) [9] propose a novel Dual Active Contour Model (DACM). DACM overcomes one of the key challenges of ACM, which is the need to correctly initialise the curve (which is often at risk of falling into local minima), by instead using two evolving curves (an outer contour and inner contour).

Sasmal & Dhal (2023) [10] compared various clustering models (notably k-means and fuzzy c-means clustering) on images that had been preprocessed by creating superpixels. They tested on both plant images (a single plant on a clear background) and medical images e.g. cell images (which were typically much noisier images and may be suited to the current problem). In this paper, pre-processing using superpixels reduced the size of the object to be processed and the complexity of the subsequent processing, which meant that new images could be segmented in rapid time. Sasmal & Dhal recommended using the SLIC algorithm for preprocessing and creating the superpixels (although other methods including the watershed methods were also considered, with less optimal results).

After the images had been pre-processed using SLIC, the images were tested based on K-means or Fuzzy C-means clustering. Whereas K-means clustering puts all pixels into only one cluster, under fuzzy c-means a pixel can be a member of multiple clusters (with varying degrees of membership). The combination of pre-processing using SLIC and using the fuzzy c-means clustering worked best in segmenting the images.

III. METHODS

A. *Data Pre-processing*

Before implementing any methods, it was necessary to split the data into subsets of training data (to fit a supervised learning model), validation data (to evaluate and fine-tune selected model) test data (to evaluate final performance of the model). Our team used the original test/train/validation split method developed by the WildScenes creators, with some slight modifications (notably to remove the need to split the 3D data into test/train/validation). This was done by running the scripts setup_data.py and wildscenes_converter.py from the original GitHub repository [11].

The output of these scripts creates a train set of 6052 instances, test set of 2134 instances and validation set of 284 instances. Importantly, the script ensures that there are samples of each class in the test/train/validation datasets.

Due to limited processing power available to this group for deep learning, each of the selected methods was trained on the same subset of 1400 training images, 300 testing images and 300 validation images. Images were resized to 512 * 512 for all instances (following the original approach from the WildScenes creators) in order to make best use of limited resources while still preserving the image detail.

The resulting training/test/validation data sets all contain examples of each category, although as is the case from the original data set, there is some class imbalance. Fig 2 shows the pixel-wise class distribution in the training data set as an example.

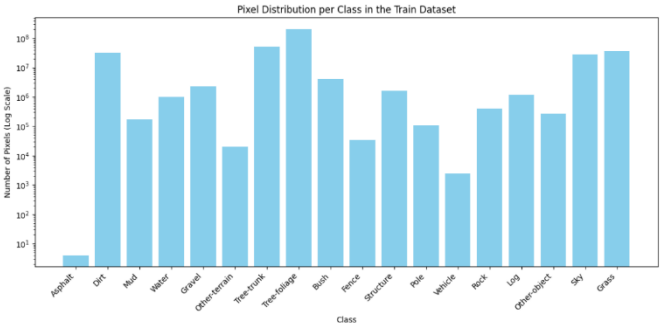


Fig 2 - Class Distribution in Reduced Training Data Set

B. *Selected Methods*

There were no studies that we could identify that used CNNs for noisy, natural images like the WildScenes dataset, however considering that FCNs/CNNs were found to be very effective for image segmentation problems, different variations of CNNs/FCNs were chosen to apply to this problem.

Our group decided not to pursue non-deep learning methods for various reasons. Primarily, the methods referenced in the literature review were largely used in medical imaging, and it was thought that they may not generalise well to the types of images in the WildScenes data set. In addition, although the DACM method developed by Fang, L., Liang, X., Xu, C. et al (2024) was promising, it was not publicly available. The fuzzy c-means approach developed by Sasmal & Dhal (2023) was more suited to an unsupervised learning problem - considering the data set was already labelled, supervised training methods were deemed to be more appropriate. The superpixel pre-processing approach from Sasmal & Dhal (2023) was selected to apply to one of the chosen methods (U-net) to see if it resulted in improved results.

C. *Background – CNNs and FCNs*

CNNs are neural networks that rely on convolutional functions and label each pixel within a detected object with a probability of being a certain class of object. FCNs are CNNs that make dense pixel-wise predictions based on an arbitrary input of images, making it ideal for semantic segmentation. FCNs are more efficient and require no additional training in comparison to other models, which simplifies and speeds up initial learning and inference of desired objects. In FCNs, each layer has three dimensions and like CNNs, they use kernels for

convolutional layers. However FCNs use non-linear filters which results in the receptive fields overlapping, allowing FCNs to be more efficient than general deep nets. The convolution networks within a FCN relies on convolution, pooling and activation functions. Fusing information from different subsampling factors improves the segmentation detail while upsampling (which is akin to backward convolution) within the network is most efficient within the FCN. An FCN can also easily learn a joint representation that predicts semantic and geometric categories [12].

D. Selected Method 1 - U-Net + Superpixels Preprocessing

1) Background

The UperNet (U-Net) architecture [13] was chosen as one method to implement because of its ability to address complex semantic segmentation tasks. This encoder-decoder framework effectively captures multi-scale features, and it can accurately segment images with varying size objects and terrains, which makes it a good candidate to consider for the WildScenes dataset [14].

U-Net is an end-to-end deep learning method (FCN) that converts input images into output segmentation maps via multiple upsampling layers and uses multiple learnable weights instead of fixed interpolation to upsample. During contraction of the image, the image is reduced in resolution while the number of feature channels are increased, allowing the deep learning network to capture higher quality features. After the final pooling operation, a deep but low resolution feature map is produced in which U-Net upsamples the map using skip connections to refine object boundaries of upsampled images. The final result has each pixel represent a different class in the foreground or background, accomplished via a 1x1 convolution layer. Each layer has a convolution filter before a ReLU filter [15].

Following our research on existing literature, we wanted to see if pre-processing the images by creating superpixels could enhance the outputs from a U-Net implementation.

2) Implementation

The SLIC algorithm was used to implement the superpixels (following the research approach developed by Sasmal & Dhal (2023). The below parameters were chosen for the superpixel creation using SLIC:

Compactness: The compactness parameter balances colour and space of the original image. The higher value of this parameter will give more weight to space proximity, making superpixel shapes more square/cubic, while a lower value allows more colour similarity [16]. We choose a value of 10, which provides a good balance between colour and space, ensuring that the segments preserve important features in the original images.

Number of Segments: The number of segments parameter is the approximate number of labels in the segmented output image [17]. As demonstrated in Fig 3 higher values result in segmentation that captures fine-grained features. After testing the model with multiple values, 500 superpixels was chosen as it provides enough detail for effective learning of different terrain types, including some difficult classes such like tree-trunk and bush.

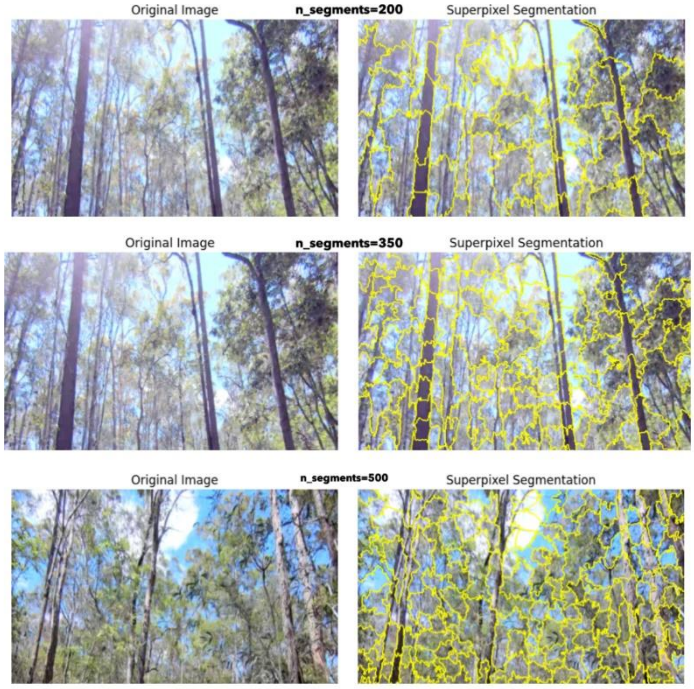


Fig 3. n_segments value difference on original images

The ConvNeXt backbone was selected as the encoder for U-Net because of its good performance compared to traditional convolutional architectures on this dataset. The ConvNeXt serves as the feature extractor for this model, each block uses large kernel sizes, depthwise convolutions and inverted bottleneck to improve model's learning ability of images' features [18]. ReLu was chosen as the activation function. CrossEntropy Loss was chosen as the loss function, due to the current problem being a multi-class segmentation problem. We selected the Adam optimizer because it is a popular choice in deep learning models for its efficiency. After running various experiments and reviewing the impacts on the training/validation loss, the learning rate was set to le-5.

The learning rate scheduler was set up to reduce the learning rate by a factor of 0.1 for every 10 epochs. This setup can help the model avoid overfitting and also converge better.

E. Selected Method 2 - Ensemble Deep Learning Model Using UNet++ and PSPNet

1) Background

The second method that was trialled was an ensemble deep learning model, combining both UNet++ and PSPNet.

UNet++ is an enhanced version of the original UNet architecture (described in method 1). Improvements on the traditional UNet model include nested and dense skip pathways (which aim to reduce the semantic gaps between encoder and decoder feature maps) and improved localisation and segmentation (which enhances the ability to segment fine details in the image).

Pyramid Scene Parsing Network (PSPNet) is an image segmentation architecture known for its ability to capture both local and global context [19]. PSPNet includes a pyramid pooling module, which allows the model to understand both fine grained and coarse details by aggregating contextual information for varying regions and scales in the image. This pyramid pooling also makes PSPNet robust to variations in object scale. It also incorporates global context, which assists PSPNet in making better predictions.

Combining UNet++ and PSPNet in an ensemble model leverages the strengths of both architectures, which we hypothesise will lead to improved performance. UNet++ generally performs well on fine-grained segmentation problems, whereas PSPNet excels in capturing global context. We hope that by combining these two models we can leverage the strengths of both models.

Generally speaking, ensemble learning combines predictions from different models. In theory, because the different models in an ensemble learning model make different errors, averaging their errors will result in stronger performance. In addition, ensemble learning can reduce overfitting, and can potentially generalise better to unseen data.

2) Implementation

For both UNet++ and PSPNet, the resnet50 encoder was used, with Imagenet used for the encoder weights. ResNet50 and Imagenet have been specifically trained on image data, which provides a strong starting point for image segmentation problems.

Cross Entropy Loss was chosen as the loss function, as it is suitable for multi-class classification problems, as well as in scenarios where classes are imbalanced - as is the case in the WildScenes data set.

The Adam Optimiser was also chosen for this method, with the scheduler chosen as StepLR due to its ability to overcome being trapped in local minima.

The results for the ensemble model were taken as the average result for the output of the UNet++ and PSPNet models.

F. Selected Method 3 - DeepLabV3 + Dual Encoders

1) Background

The third method that was trialled was a DeepLabV3 model, enhanced with dual encoders.

DeepLabV3 is a semantic segmentation architecture which has features including dilated convolution (which expands the receptive field without increasing parameters or computational complexity) and an Atrous Spatial Pyramid Pooling (ASPP) module which captures contextual information at different scales through parallel multi-scale dilated convolutions. It applies several dilated convolutions with different dilation rates on the feature map and then fuses these features to enhance multi-scale feature representation [20].

We wanted to enhance the standard DeepLabV3 implementation (which typically includes a single encoder) by implementing a dual encoder. We hypothesised that by extracting image features separately with each encoder we could then concatenate these features in the channel dimension, and then pass them through a classification head to generate the final segmentation result.

2) Implementation

After trailing many different loss functions (cross-entropy loss, dice loss, and a hybrid model involving both of these), the focal loss function was chosen as it was designed to handle class imbalance problems, it has good discrimination ability for small objects and difficult-to-classify samples. It can dynamically adjust the weights of different samples, focusing on difficult-to-classify samples. However the selection of hyperparameters α and γ are sensitive and needed to be adjusted. After running experiments we chose an α value of 0.2 and a γ value of 4.

The encoders that were chosen were ResNet-101 and EfficientNet-B3. EfficientNet-B3 is a lightweight class-based network which is trained to classify objects with high accuracy and compared with other models, EfficientNet101 has fewer parameters and is more computationally efficient.

ResNet-101 is used with the U-Net model mentioned previously, however was chosen again due to its ability to recognise complex features [21]. It has a depth of up to 152 layers that can be selected, which can capture very complex features but may lead to overfitting. To prevent this, we carefully monitored training and validation loss.

In order to run ResNet-101 with EfficentNet-B3, a series of dilated convolutions was made in order of the correct batch normalisation and ReLU layers. To ensure consistent output shapes in each section, the feature shape map was verified. The scale factor of the upsampling operation had to be adjusted so that the shapes of the output and index labels are the same size when passed into the loss function.

IV. EXPERIMENTAL RESULTS

The metric used to evaluate model performance was Mean Intersection Over Union (MIoU). This is taken by calculating the Mean IoU for each class, and then getting the mean value for the selected method. MIoU is a measurement of the accuracy of a model's segmentation, object annotation and object detection. As semantic segmentation involves classifying each pixel as part of a detected object class and identifying the boundaries of each

object, MIoU is the most appropriate metric of measurement for the models. MIoU is often used to evaluate semantic segmentation tasks [22].

1) Results Table

Class IoU (2 decimal places)	Method 1 (Superpixels + UNet)	Method 2 (UNet++ & PSPNet)	Method 3 (DeepLabV3 + encoder)
Bush	0.03	0.12	0.21
Dirt	0.61	0.78	0.78
Fence	0.01	0.00	0.00
Grass	0.52	0.65	0.63
Gravel	0.06	0.00	0.00
Log	0.05	0.00	0.21
Mud	0.01	0.00	0.00
Other Object	0.01	0.00	0.05
Other Terrain	0.00	0.00	0.00
Rock	0.00	0.00	0.19
Sky	0.52	0.73	0.62
Structure	0.04	0.00	0.00
Tree Foliage	0.67	0.82	0.77
Tree Trunk	0.25	0.52	0.29
Water	0.03	0.00	0.60
Mean IoU	**0.15**	**0.39**	**0.42**

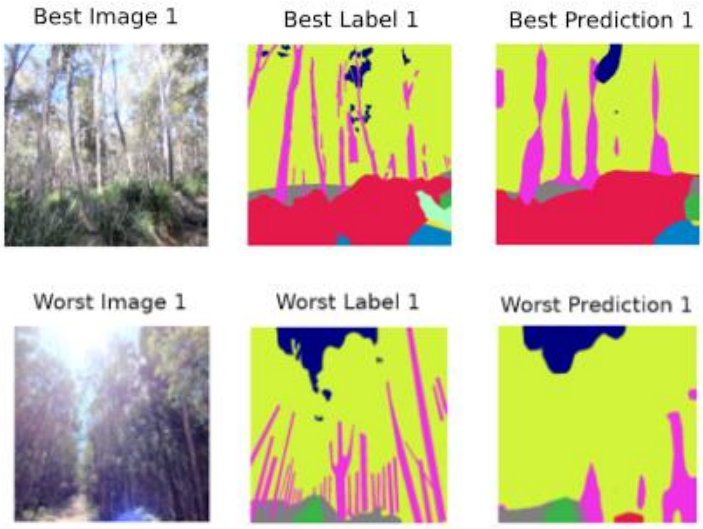
Some notes on the table of results:

- 0.00 values represent classes that were in the dataset, but that the model failed to correctly segment. Some of these values were slightly greater than 0 but when rounded to 2 decimal places became 0.00.

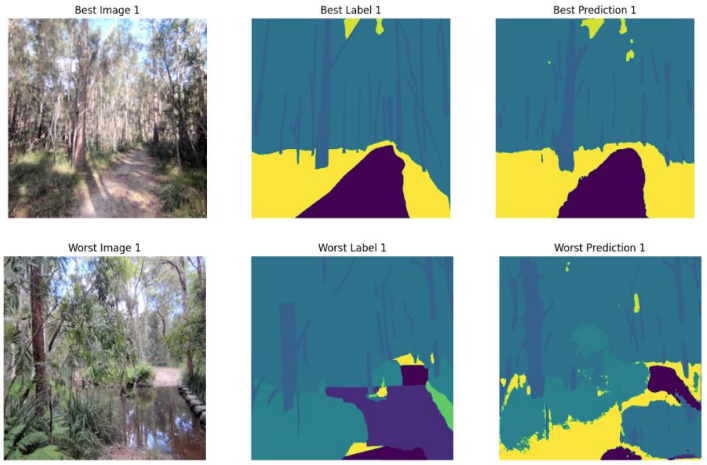
2) Examples of Segmentation From Each Model

Below are some examples of how each method performed. We have included 2 examples from each method to demonstrate the type of images that were well-segmented vs poorly segmented.

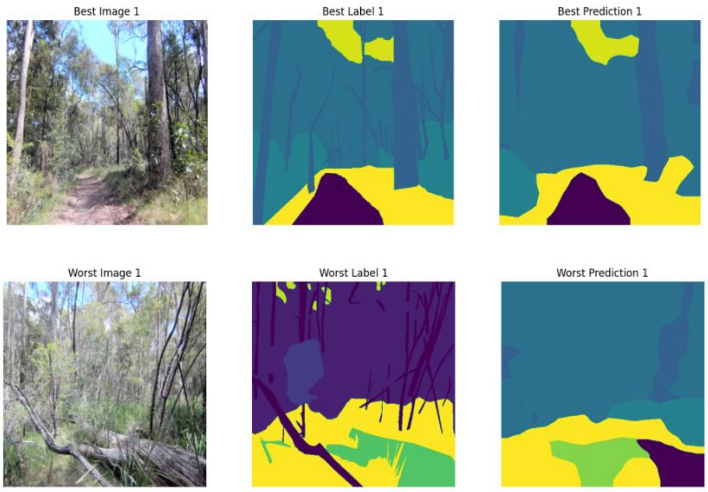
a) Method 1 - Superpixels + UNet



b) Method 2 - UNet++ & PSPNet Ensemble Method



c) Method 3 - DeepLab V3 + Dual Encoder



V. DISCUSSION

The WildScenes creators were able to achieve an MIOU of 47.49 using U-Net with a ConvNeXt-L backbone, and 42.93 by using DeepLabV3 with a resnet encoder.

Our three methods achieved MIOU scores of 0.15 (U-Net + Superpixels), 0.39 (Ensemble Model) and 0.42 (Dual Encoder Model), which unfortunately could not do better than the benchmarks established by the WildScenes creators.

All three methods performed best on objects that were represented most in the data set (tree trunks, tree foliage, sky, grass and dirt), and poorly on classes that were under-represented in the dataset (fence, mud). Across the three methods, generally speaking where one method performed well or poorly, the others did the same. The only exceptions were in the Log, Water and Rock classes - where the DeepLabV3 (Dual Encoder) method was able to correctly segment the image in these classes part of the time, compared with the other two methods that were not able to identify these classes at all.

Considering the benchmarks established by the WildScenes creators, the MIOU results for methods 2 and 3, while not ideal are largely acceptable. Separation between classes in noisy natural environments can be particularly challenging, the density of objects in the images, and the co occurring nature of items at the same time (e.g. leaves and trunks overlapping) making these problems very difficult.

Method 1 (Superpixels + UNet) was by far the worst performing model, achieving an MIOU of 0.15, which was quite a lot lower than the UNet benchmark established by the WildScenes creators (47.49). This drop in performance was likely due to the impact of the superpixel pre-processing stage.

1) Potential Impact of Superpixel Segmentation

Superpixel segmentation methods (including SLIC which was used here) group pixels with similar features [23]. One similarity measure used is colour. Considering that the WildScenes images contain many variations of similar colours (e.g. different shades of green for foliage), this could be one factor that limited the effectiveness of the superpixel pre-processing approach. This could explain why classes such as sky and dirt performed well with this model, as they would contain largely homogenous colour values.

Another factor which could explain the differentiation in effectiveness between classes is potentially due to the varying size and shapes of objects in the WildScenes images. Irregularly-shaped objects are typically more challenging for superpixels to segment, compared with regularly shaped objects.

2) Ensemble Learning Approach

Method 2 (Ensemble Learning) overall performed fairly well, with an MIOU of 0.39. It was clear that the ensemble learning method worked - when the individual models were run before combining their output, UNet++ achieved an MIOU of 0.2443 and PSPNet achieved an MIOU of 0.3494.

Individually, UNet++ performed particularly well on classes such as Dirt (IOU: 0.7704), Grass (IOU: 0.6342), Sky (IOU: 0.7278), and Tree-foliage (IOU: 0.7648). However, it struggled with other classes like Bush, Fence, Gravel, Mud, and Other-terrain, which had IOUs close to or equal to zero. This indicates that while UNet++ is effective for certain classes, it has limitations in handling diverse segmentation tasks across all categories.

PSPNet excelled in classes such as Dirt (IOU: 0.7585), Grass (IOU: 0.6157), Sky (IOU: 0.7068), and Tree-foliage (IOU: 0.7858). PSPNet's robust handling of global context likely contributed to its superior performance on these classes. However, it also encountered challenges with classes like Bush, Gravel, Log, and Other-object, with 0.00 values (or very close to) for these classes.

The Ensemble Model combining UNet++ and PSPNet resulted in the highest Mean IOU of 0.3947, showcasing the effectiveness of the ensemble approach. The class-wise IOU analysis shows notable improvements across several categories:

- Bush: Improved from near zero (UNet++) and 0.0590 (PSPNet) to 0.1171.
- Dirt: Further improved to 0.7803, showing robust performance across both individual models.
- Grass: Increased to 0.6509, indicating better handling of this class with the ensemble.
- Sky: Improved to 0.7344, reflecting the strengths of both models in capturing this class.
- Tree-foliage: Further improved to 0.8172, the highest among all tested models.

Classes like Gravel, Log, Other-object, and Other-terrain remained challenging, with the ensemble model not significantly improving their segmentation. This suggests that while the ensemble method improves overall performance, there are still inherent challenges in segmenting certain classes that may require further investigation and possibly different architectural approaches.

3) DeepLabV3 + Dual Encoders

Although the DeepLabV3 model performed the best from the three methods, it performed lower than the benchmark established by the WildScenes creators using DeepLabV3 (42.93).

This could be because the training data size used in this project (1400) training samples was approximately ⅔ of the size of the training data set used by the WildScenes creators. In order to determine whether this method improved or worsened the standard DeepLabV3 implementation, it would be interesting to run this model again, using the exact same data set as the WildScenes creators.

VI. CONCLUSION AND FUTURE RESEARCH

In summary, the best results were gained from using an ensemble learning method of UNet++ and PSPNet, and from using DeepLabV3 with dual encoders. While no method improved upon the benchmark established by the WildScenes creators, aside from increasing the training sample size, we have considered some directions for potential future research to improve these methods.

1) Potential Improvements

a) Method 1 (U-Net + Super Pixels)

We suggest trialling to improve this method by setting up early stopping to prevent overfitting and avoid fluctuations when training the model. Implementing early stopping will select the best model based on validation performance.

We also suggest experimenting with different learning rate schedules and adaptive learning rate methods to lower the level of convergence of the model. Further exploration of various combinations of hyperparameters could also be trialled to optimise U-Net's performance.

b) Method 2 (U-Net + PSP-Net Ensemble Learning)

By exploring additional model architectures and ensemble strategies, further enhancements in segmentation accuracy and robustness across all classes could potentially be developed.

c) Method 3 (DeepLabV3 + Dual Encoders)

Further research directions could include enhance data pre-processing and image scaling. We trialed various image preprocessing step combinations including contrast and brightness adjustments and gaussian blur, however all resulted in a drop in MIOU in the final result.

VII. REFERENCES

[1] Vidanapathirana, Kavisha, et al. "WildScenes: A Benchmark for 2D and 3D Semantic Segmentation in Large-scale Natural Environments." arXiv preprint arXiv:2312.15364 (2023).

[2] https://www.library.unsw.edu.au/

[3] https://link.springer.com/

[4] https://paperswithcode.com/

[5] Sun, Y., Zheng, W. HRNet- and PSPNet-based multiband semantic segmentation of remote sensing images. Neural Comput & Applic 35, 8667–8675 (2023).

[6] Lei, L., Yang, Q., Yang, L. et al. Deep learning implementation of image segmentation in agricultural applications: a comprehensive review. Artif Intell Rev 57, 149 (2024).

[7] Shao H, Tang R, Lei Y, Mu J, Guan Y, Xiang Y. Rice Ear Counting Based on Image Segmentation and Establishment of a Dataset. Plants. 2021; 10(8):1625.

[8] Wang D, Cao W, Zhang F, Li Z, Xu S, Wu X. A Review of Deep Learning in Multiscale Agricultural Sensing. Remote Sensing. 2022; 14(3):559.

[9] Fang, L., Liang, X., Xu, C. et al. Image segmentation using a novel dual active contour model. Multimed Tools Appl 83, 3707–3724 (2024).

[10] Sasmal, B., Dhal, K.G. A survey on the utilization of Superpixel image for clustering based image segmentation. Multimed Tools Appl 82, 35493–35555 (2023).

[11] Kavisha Vidanapathirana and Joshua Knights and Stephen Hausler and Mark Cox and Milad Ramezani and Jason Jooste and Ethan Griffiths and Shaheer Mohamed and Sridha Sridharan and Clinton Fookes and Peyman Moghadam, csiro-robotics/WildScenes, 2023, GitHub repository.

[12] J.Long, E.Shelhamer, T.Darrell Fully Convolutional Networks for Semantic Segmentation, 8 March 2015, Found at: https://arxiv.org/pdf/1411.4038.

[13] Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science(), vol 9351. Springer, Cham.

[14] Ruiping, Y., Kun, L., Shaohua, X., Jian, Y., & Zhen, Z. (2024). ViT-UperNet: a hybrid vision transformer with unified-perceptual-parsing network for medical image segmentation. Complex & Intelligent Systems, 10(3), 3819–3831.

[15] Ronneberger, Olaf & Fischer, Philipp & Brox, Thomas. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. LNCS. 9351. 234-241. 10.1007/978-3-319-24574-4_28.

[16] skimage.segmentation — skimage 0.24.1rc0.dev0 documentation. (n.d.).https://scikit-image.org/docs/dev/api/skimage.segmentation.html.

[17] skimage.segmentation — skimage 0.24.1rc0.dev0 documentation. (n.d.).https://scikit-image.org/docs/dev/api/skimage.segmentation.html.

[18] Z. Liu, H. Mao, C. -Y. Wu, C. Feichtenhofer, T. Darrell and S. Xie, "A ConvNet for the 2020s," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, 2022, pp. 11966-11976, doi: 10.1109/CVPR52688.2022.01167.

[19] Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2881-2890).

[20] Chen, L., Papandreou, G., Schroff, F., Adam, H. (5 December 2017) Rethinking Atrous Convolution for Semantic Image Segmentation, Google Inc., Academic Paper, Accessed on 30th of July 2024, Found at: https://arxiv.org/pdf/1706.05587v3

[21] Cheng, C. (23 November 2020) EfficientNet and ResNeXt101_wsl series, PaddleClas, E-Book, Accessed on 30th of July 2024, Found at: https://paddleclas.readthedocs.io/en/latest/models/EfficientNet_and_ResNeXt101_wsl_en.html

[22] Wang, Z., Wang, E. & Zhu, Y. Image segmentation evaluation: a survey of methods. *Artif Intell Rev* **53**, 5637–5674 (2020).

Wang, N., & Zhang, Y. (2021). Adaptive and fast image superpixel segmentation approach. Image and Vision Computing, 116, 104315. https://doi.org/10.1016/j.imavis.2021.104315.