Enhancing Aerial Image Semantic Segmentation of Tiny Objects using Hierarchical Constrained CNN Approach

Zhang Rongchuan[1], Azizi Abdullah[2], Adrus Mohamad Tazuddin[3]
Center for Artificial Intelligence and Technology
Faculty of Information Science and Technology
Universiti Kebangsaan Malaysia
a173165@siswa.ukm.edu.my[1], azizia@ukm.edu.my[2], p118049@siswa.ukm.edu.my[3]

Abstract*—*Semantic segmentation of tiny objects has excellent potential in rescue monitoring and medical applications. However, it faces several significant challenges, such as insufficient pixels and class imbalance problems. The small object class may have very few samples compared to the background or larger object classes, resulting in an unequal representation of classes. Thus, this can lead to a biased model that performs poorly in accurately segmenting small objects. This paper studies a simple FPN-based and proposed a hierarchical structure for semantic segmentation, which enables the model to perform a coarse-to-fine classification for organizing object classes into groups and subgroups at different levels of granularity. This allows the model to extract more spatial and context information on tiny objects. In addition, this structure emphasizes the importance of grouping tiny objects into general classes during training to prevent them from being mistakenly considered as other opposite object classes. Two models are proposed using different pixel constrain rules to emphasize difficult-to-recognize pixels and allow the model to capture the semantic information of tiny objects better. The experiments have shown that the segmentation of tiny objects on the UAVid and Danish golf dataset resulted in the two models achieving 90.6% and 83.1% accuracy, respectively. Improvement of tiny object segmentation performance compared to the baseline model shows the innovation with the proposed hierarchy-based structural model could be used further to improve segmentation performance on similar datasets and problems.

Keywords*—*computer vision; semantic segmentation; tiny object detection; hierarchy-based CNN

I. INTRODUCTION

Semantic segmentation is a fundamental task in computer vision that aims to assign a semantic label to each pixel in an image. It has wide applications in various fields, such as autonomous driving, robotics, and medical image analysis. Since the rise of modern deep learning and convolutional neural network, which led to the development of models such as FCN [3] and U-Net [4], semantic segmentation has been given more attention in computer vision due to its highly practical and reliable implementation. Thus, it can be observed that semantic segmentation has made remarkable progress and outstanding research and application achievements in recent years.

However, there are still significant challenges in semantic segmentation based on small objects. The reason is that tiny objects are small and have low pixel resolution, which are usually difficult to detect and segment accurately, especially in complex scenes where small objects could be obscured by other objects or become indistinguishable from large objects. Existing deep neural network-based semantic segmentation methods achieve impressive results on large objects however, the performance on small objects still needs to be improved due to the limitation of spatial details and context information. This issue hinders the accurate segmentation of small objects as it is difficult for deep learning algorithms to capture the characteristics of small regions accurately. Moreover, in the tiny objects semantic segmentation task, the ratio of the number of small and large object pixels is imbalanced data, which often causes biased predictions, and can lead to inaccurate results in deep learning models.

Fully Convolutional Networks (FCNs), U-Net, and Feature Pyramid Networks (FPN) are popular and widely used architectures for semantic segmentation tasks in computer vision. However, FPN has received more attention recently due to having a multi-scale representation that incorporating skip connections to combine features from different layers and capture context at multiple scales, making it effective for handling objects of varying sizes, which is crucial for accurate pixel-wise classification. FPN typically consists of a backbone network that extracts features from the input image and a top-down pathway that generates feature pyramids by upsampling lower-level features and fusing them with higher-level features. This enables the model to capture both local and global contextual information, allowing for more precise object boundary delineation and improved accuracy in segmenting objects with varying scales and shapes. FPN also includes lateral connections that facilitate information flow across different scales, enhancing feature reuse and reducing the risk of losing spatial details during the upsampling process. Overall, FPN is an effective approach for semantic segmentation, enabling accurate and robust segmentation of objects in images with varying levels of complexity and scale.

Although Feature Pyramid Network (FPN) has proven to be effective for many semantic segmentation tasks, it can encounter challenges when it comes to small object segmentation. Due to the down-sampling and up-sampling operations in FPN, small objects may lose their fine details and get aggregated with neighboring regions, leading to decreased segmentation accuracy. The loss of spatial resolution during the up-sampling process can result in inadequate localization of small objects, causing them to be overlooked or merged with their surroundings. Additionally, Feature Pyramid Network (FPN) may encounter challenges when dealing with imbalanced data in semantic segmentation tasks. Imbalanced data, where certain object classes are under-represented compared to others, can adversely impact the training and performance of FPN. The model may have biased learning towards the majority class, leading to poor segmentation accuracy for minority classes. This can result in misclassifications, poor object boundary delineation, and reduced overall segmentation performance. Additionally, during the training process, FPN may struggle to effectively learn features from the minority class due to the limited number of samples, resulting in reduced model generalization and performance on imbalanced data during inference. Addressing the issue of imbalanced data in FPN may require techniques such as class weighting, data augmentation, or resampling strategies to ensure fair representation of all object classes and improve the accuracy and robustness of semantic segmentation results. As a result, additional strategies, such as object proposal mechanisms or post-processing steps, may be required to mitigate these limitations and improve the performance of FPN in small object segmentation scenarios.

Object hierarchical grouping is a crucial process in computer vision. The scheme organises objects into a hierarchical structure based on such as visual attributes, relationships, or semantics. This process provides a more comprehensive and structured representation of visual data, which is particularly beneficial for object segmentation at different scales and levels of abstraction. Besides, hierarchical grouping allows for the identification of objects at different levels of granularity, from global scene understanding to fine-grained object details. It also facilitates extracting contextual information, as the hierarchical structure captures the relationships between objects in a scene, including their spatial arrangements and part-whole relationships. Moreover, hierarchical grouping can aid in semantic segmentation tasks by organizing objects into meaningful categories or classes based on their hierarchical position, as it enables a more structured and comprehensive representation of small objects, leading to improved visual understanding and interpretation. Furthermore, the human brain uses hierarchical grouping to organize and make sense of complex visual scenes, enabling efficient object recognition and scene understanding.

Two datasets are used in this project, namely UAVid and Danish golf. These datasets are chosen as it reflects the practical usage of image segmentation of aerial images that largely deal with tiny objects.

This article proposes an innovative method to improve tiny target semantic segmentation detection accuracy using hierarchy-based CNN model. Related works which used similar technique is discussed in the next section together with prior works on semantic segmentation which is used for comparison. Other components that are used in the proposed model such as the backbone which utilizes EfficientNet [7] and the neck of the model with FPN [8] will also be outlined. The third part introduces the methodology which consists of the proposed model's architecture, differences between the two proposed model pixel constrain rules, the hierarchy structure and the model's loss function. On the last section the experimental setup and evaluation results regarding the model performance compared to the baseline model will be discussed.

II. RELATED WORK

A. Background

Semantic segmentation is getting more attention in computer vision. Semantic segmentation aims to divide a picture into multiple parts by labelling each pixel in an image usually by masking the pixels with colour shade that represent a different class. Image segmentation have a broad implementation that covers broad scenarios, such as medical device images, aerial images and surveillance images. There are three types of image segmentation which include semantic segmentation, instance segmentation and panoptic segmentation. The type implemented in this project is semantic segmentation where each class is assigned with the same mask colour without the need to differentiate the individual object within the same class which is implemented in instance and panoptic segmentation.

The earliest papers on semantic segmentation can be traced back to the mid-1980s, and the earliest methods were based on rules and thresholds. Before neural network deep learning methods are widely used, manually handcrafted algorithm which is regarded nowadays as traditional methods are used to implement image segmentation. Traditional methods include methods based on region, edge detection and layer feature methods. One example is the non-parametric clustering algorithm based on mean shift where pattern of data points is figured out by iteratively moving the data points to the area with the highest local density [1]. Image segmentation methods based on graph theory where image is represented as a weighted undirected graph and the segmentation results is obtained by minimizing the normalized cut of the graph [2].

With the advancement of computer technology, deep learning and convolutional neural network approach, more sophisticated and accurate semantic segmentation methods have emerged. One of the earlier and well-known deep learning method for image segmentation is FCN (Fully Convolutional Networks) [3]. FCN extends the traditional convolutional neural network (CNN) into a network capable of processing input images of arbitrary size. By using deconvolutional layers to restore image resolution, FCN could perform semantic segmentation while maintaining pixel-level accuracy. Although it could perform well on large objects, the segmentation accuracy of small objects seen a drastic decline.

In [4], the model U-Net proposes a method for biomedical image segmentation where it combines the merits of the convolutional neural network architecture and the FCN model. Skip connections are used to help the network capture local details in the image. It can retain the details of small objects targets through fusing multiple layers of different scales, therefore improving the model's performance in small target segmentation. The combination of these features has led towards the

excellent capability of U-Net to accurately segment input images.

Another approach in [5] uses a semantic segmentation method to generate object candidate regions and then a convolutional neural network is used to classify each of the candidate. Furthermore, a context encoding semantic segmentation module is proposed in [20] where the main idea is to use the middle layer features of the convolutional neural network to learn context information from different scales. This context information is then fused with the input feature map to obtain more accurate semantic segmentation results.

B. Hierarchy-based CNN

Branch-CNN introduced in [6] outlined a hierarchical-based CNN where there are multiple branches in a neural network that is used to classify different coarse level to fine level labels. The placement of the coarse level classifiers is towards the lower layers of the neural network which is rich in spatial information and the finer level classifiers will be on the upper layers of the neural network where more semantic information is available. The model uses weighted loss function to focus learning from the coarse level before gradually shifting the focus towards higher levels as the epoch increases. This allows the model to learn more context information as it could learn the similar features of objects that belong in the same category first as an easy task before utilizing the knowledge to tackle more complex task of differentiating them.

C. EfficientNet

The Google Brain team created EfficientNet, an efficient convolutional neural network structure, in 2019 [7]. The structure of EfficientNet consists of two main components: the primary network and the composite scaling factor. The primary network structure is a series of repeated modules composed of convolutional blocks. These modules can be scaled at different depths, widths, and resolutions, resulting in multiple EfficientNet models of different sizes. EfficientNet has demonstrated high performance in many image classification tasks while maintaining computational efficiency. It achieves state-of-the-art results on the ImageNet dataset and performs well on other datasets. In this project, EfficientNetB0 is used as the backbone of the model. The main reason is that the compound scaling feature extraction network structure used by EfficientNet is very efficient and is suitable as a backbone for small target detection.

D. FPN

FPN (Feature Pyramid Network) is a feature pyramid network proposed by FAIR (Facebook AI Research), which is mainly used to solve the problem with differing scale of objects in object detection tasks [8]. The main idea of FPN is to fuse semantic information from different feature layers to construct richer feature representations. Firstly, bottom-up feature extraction extracts multi-scale features of an image through a series of convolutional layers to form a feature pyramid. Then, a top-down feature fusion transfers information layer by layer from the high-level feature map to the low-level feature map. It combines the detailed information in the low-level feature map with the semantic information in the high-level feature map to form a richer feature representation.

For small object semantic segmentation, through the multi-layer feature pyramid structure, FPN can extract target information of different sizes on feature maps of different scales, thereby effectively improving the detection and semantic segmentation accuracy of small targets. In this project, EfficientNet and FPN are used together as the backbone and neck part of the model to extract features as both models statistically have high synergy for small object semantic segmentation.

III. METHODOLOGY

A. Hierarchical Constrained-CNN

The aim of hierarchy-based CNN model is to guide the model to learn coarse-grained features first and then use the coarse-grained features as a basis to infer fine-grained results. Hierarchical structures based on B-CNN [6], where multiple branches with multiple classifiers are used as the main reference for the overall framework of our proposed model. Besides that, the hierarchy structure from [9] and tree-structured models from [10] are sources of ideas and building blocks to complete the hierarchy model. Instead of letting each classifier classify independently, a proposed constrain feature map is introduced so that coarse-grained outputs can improve fine-grained outputs. The proposed model Hierarchical Constrained CNN (HC-CNN) uses the segmentation from the coarse output as a restricting map that will be used to guide the fine segmentation output. The architecture of HC-CNN model is shown in Figure 1.

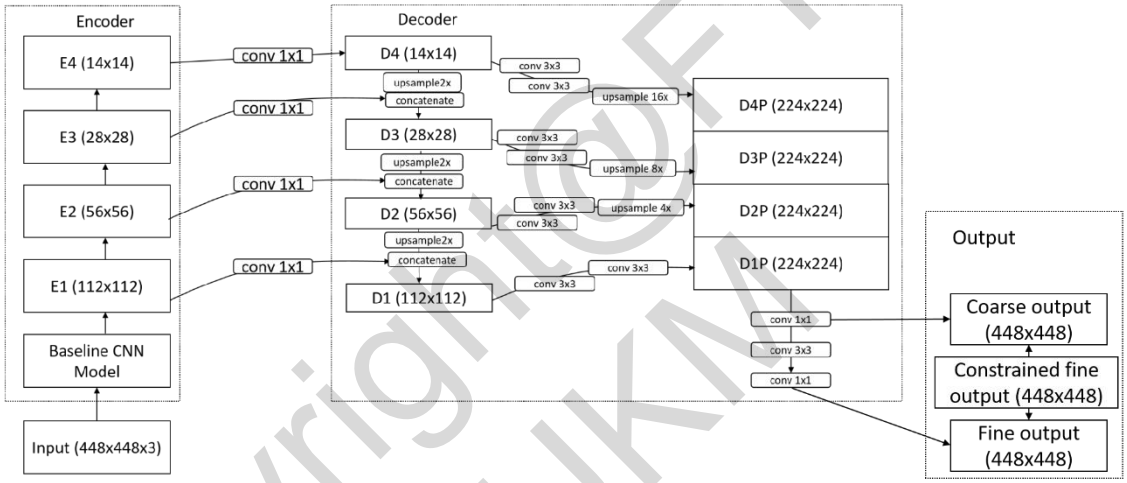


Figure 1. The architecture of the HC-CNN (the process flow of constrained fine output is not clear; arrows should show what is its input and where the output goes or being used with clear arrow directions)

B. Hierarchy class levels assignment

In a hierarchy-based model, the classification of the coarse category will affect the fine classification, but it is not absolute and should only be taken as a guide to mould the network during training. Therefore, we proposed Algorithm 1 to build a hierarchy structure that will be used to relate the two levels of coarse and fine classification. This mapping will be used by the model during training.

Algorithm 1 Rules to classify the hierarchy relation between the coarse class level and fine class level	
Step 1.	Number of coarse class categories are half as large as fine class categories.
Step 2.	According to observation, shape and colour, similar classes are classified as in the same coarse class group.
Step 3.	Draw a histogram. Large objects will first be excluded and be classified into the same coarse class group.
Step 4.	The rest are divided into categories according to Step 2 taking into consideration that and the number of coarse classes in Step1 must be met.

This method enforces the neural network to learn similarities between objects that are grouped in the same coarse group before then learning to differentiate them. In addition, we classify tiny and large object separately. Suppose the tiny object is mistakenly identified as a large object, such as the background at a coarse-grained level. The loss will be calculated again in the constrained output which inform the model to better distinguish tiny targets and the background.

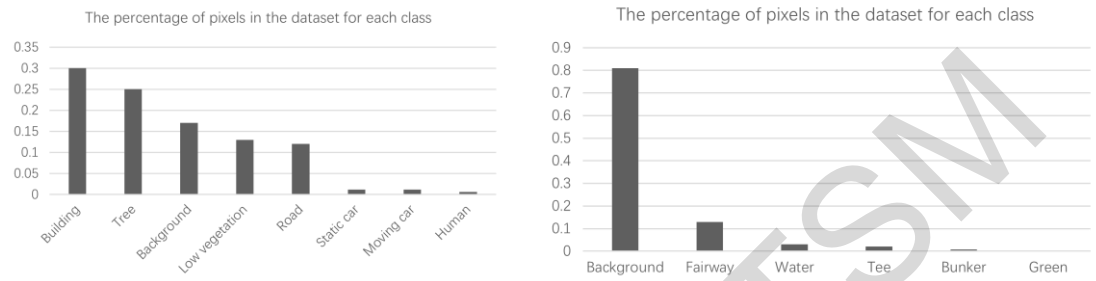


Figure 2. Left: Percentage of pixels for each class in UAVid dataset. Right: Percentage of pixels for each class in Danish golf dataset

This project used two datasets to evaluate our model: UAVid and the Danish golf dataset. Based on the Algorithm 1, there are 4 coarse classes and 8 fine classes for UAVid dataset. From observation, both static and moving cars are vehicle so that they will be placed into one category. Both low vegetation and trees are green plants, so they are in one category. Roads, background and the buildings are grouped into one category because they are considered as large objects. Human is a single entry of the last category which makes the total of 4 coarse classes. The hierarchy of coarse class category and fine class category in UAVid dataset is shown in Figure 3.

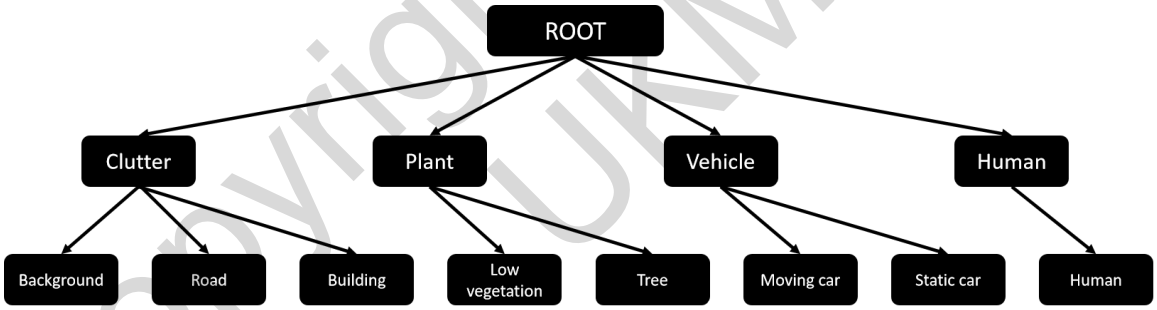


Fig. 3. Hierarchy of coarse class and fine class category in UAVid dataset

Based on the Algorithm 1, there are 3 coarse classes and 6 fine classes for Danish golf dataset. According to observation, both tee and green are similar in shape and colour, so we place them into one group. Fairway and background are considered large objects and are classified into one category. Water and bunker will be grouped into one category. The hierarchy relation between coarse and fine class category in Danish golf dataset is shown in Figure 4.

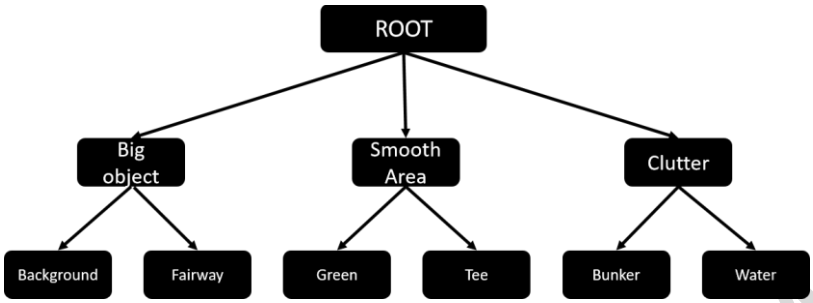


Fig. 4. Hierarchy of coarse class and fine category in Danish golf dataset Strong and weak confidence pixel

To guide fine prediction, the coarse level pixel prediction segmentation will be processed to generate a constrain map. This map will record the level of confidence of prediction from the coarse level. Each channel corresponds to a class and through the SoftMax function, the range of the confidence will be between 0 and 1 which also reflects its probability of coarse level classification. For example, a score of 1 on the class of vehicle means the model is one hundred percent confident that the pixel segmentation belongs to a vehicle.

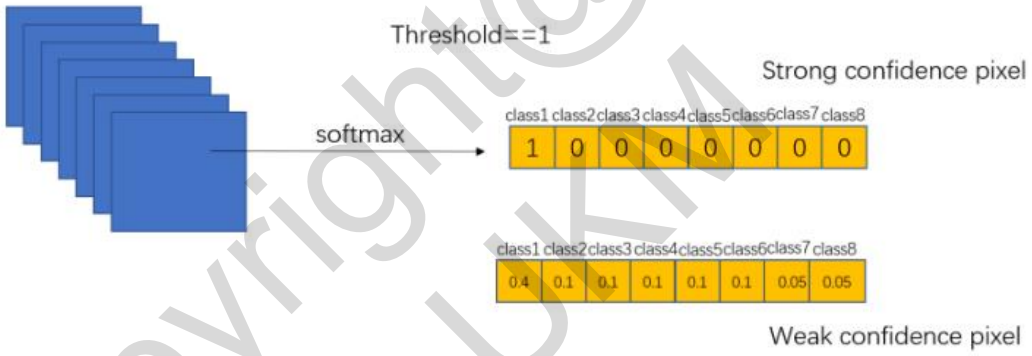


Figure 5. Strong and weak confidence pixel

The pixels whose probability of belonging to the class is greater than or equal to the threshold are defined as strong confidence pixels. Those probability of belonging to the class is less than the threshold are weak confidence pixels. In this project, the threshold is set to 1, which means that only when the model is 100% confident that the belongs to a pixel category, this pixel will be considered a strong confidence pixel. Otherwise, it is considered a weak confidence pixel as shown in Figure 5.

$$\max(pixel_{w,h}) \geq threshold \qquad (1)$$

$$\max(pixel_{w,h}) < threshold \qquad (2)$$

C. Constrained pixel map

From Figure 1, the structure of the proposed model could be observed. There are three outputs: coarse output, fine output, and constrained pixel map. Coarse output will classify each pixels in the input image based on the highest level of the hierarchy at the coarse level. This output is then

processed to generate a constrained pixel map. The input of the coarse classifier will go through another two convolutional layers; 3x3 and 1x1 before it will proceed to the fine level classifier which will also utilizes the constrained pixel map to improve its prediction (Figure 6).

During the mapping process, the mapped pixel can only belong to its coarse-grained subclass. For example, in the UAVid dataset, a pixel is classified as the vehicle at a coarse-grained level, and then this pixel can only be classified as a static car or moving car after mapping. Based on this concept, we proposed two different models based on two different mapping approaches namely HC-CNN (Hierarchical constrained CNN) and HPSC-CNN (Hierarchical pixel strength constrained CNN). If all points predicted in the coarse level is used in the mapping, we get a constraint feature that is the same as the coarse output. It is used for the model in which the mapping restricts all pixels which is called HC-CNN. On the other hand, for HPSC-CNN, only strong confidence pixels will be used for mapping. The weak confidence pixels will not be restricted and can be classified into any category during the fine level classification. For example, a weak confidence pixel in the UAVid dataset is initially classified as clutter in coarse level output. After mapping, this pixel can still be classified as a static vehicle as no constrain is placed from the pixel due to having a weak confidence pixel in the coarse classification.

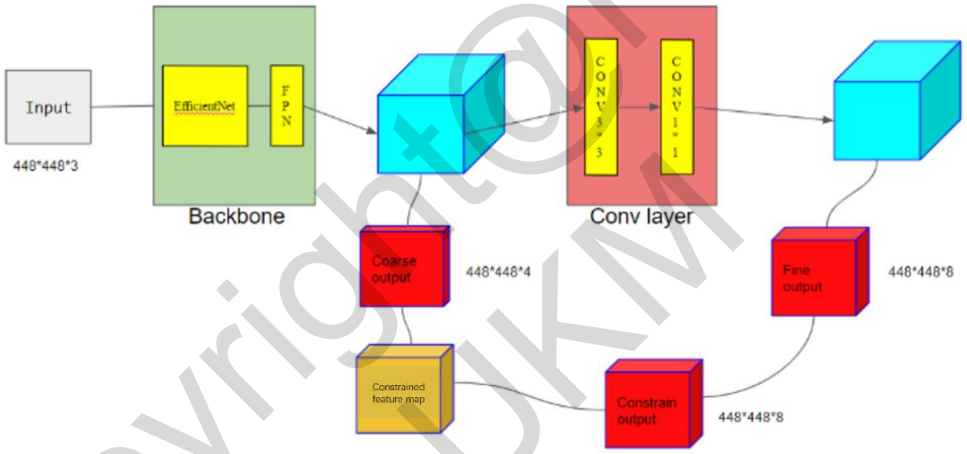


Figure 6. Structure of constrained pixel map

D. Loss function

In this project, we use all three outputs that consists of fine, coarse and constrained pixel map output to calculate the loss of the proposed model as shown in equation 3-5. Dice loss and focus loss are combined because both contribute advantageously for highly uneven samples in terms of positive samples and negative samples; in this case represented by small objects and background. Dice Loss is a measure of overlap that measures the similarity between the model prediction results and the actual label (Equation 6-7). It calculates the degree of overlap between the predicted result and the actual label, and its value ranges from 0 to 1. When the predicted result is entirely correct, the value of Dice Loss is 0. When the predicted result is completely different with the actual label, Dice Loss The value is 1. Compared with other loss functions, Dice Loss is more sensitive to the difference between the predicted result and the actual label.

$$Total_{loss} = Focus\_loss_{total} + Dice\_loss_{total} \quad (3)$$

$$Dice\_loss_{total} = Dice\_loss_{fine\ output} + Dice\_loss_{coarse\ output} + Dice\_loss_{constrain\ output} \quad (4)$$

$$Focus\_loss_{total} = Focus\_loss_{fine\ output} + Focus\_loss_{coarse\ output} + Focus\_loss_{constrain\ output} \quad (5)$$

$$Dice = \frac{2(predict \cap label)}{predict + label} \quad (6)$$

$$Loss_{Dice} = 1 - Dice \quad (7)$$

Each pixel is assigned to a different semantic category in semantic segmentation tasks. For multi-classification problems, the commonly used loss function is cross entropy (Equation 8). Each pixel has a corresponding label, which means that there are many background pixels in the data, which makes the ratio between positive and negative samples unbalanced. To solve this problem, Focus Loss can be used to balance positive and negative samples and improve the model's performance on meaningful pixels that consists of small objects (Equation 9). It focuses the model training on the sparse subset of complex samples and avoids overwhelming the detector with many simple negative samples during training.

$$Loss_{Cross\_entropy} = -log(p_t) \quad (8)$$

$$Loss_{Focus\_loss} = -(1 - p_t)^{\lambda} \log(p_t) \quad (9)$$

IV. RESULTS AND ANALYSIS

A. Experiment setup

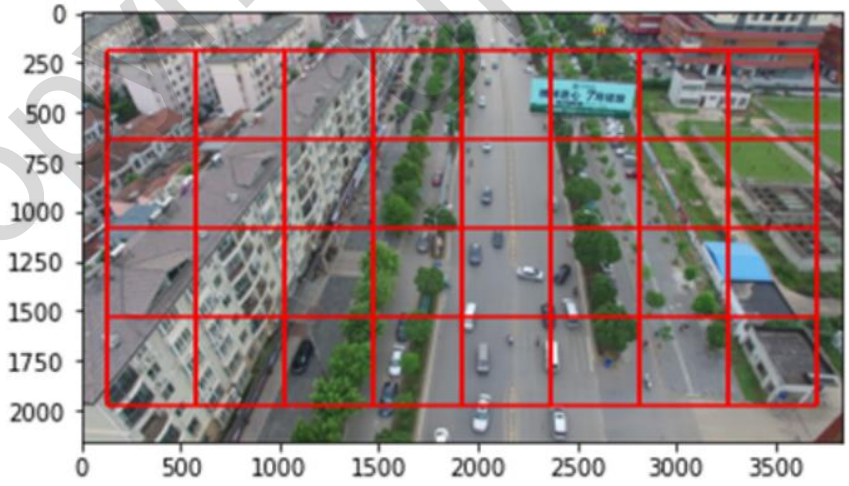


Figure 7. Partition grid of UAVid image

The UAVid [19] dataset is a UAV video dataset for semantic segmentation tasks in urban scenes. It is a 4K high-resolution dataset that can be used for semantic segmentation. The resolution of each image is 3840×2160 and has pixel-level labelling. Each image is labelled through segmentation masking with eight categories Building, Road, Static car, Tree, Low vegetation, Human, Moving car, and Background clutter. Among them, moving cars, static cars and humans

are considered small objects. In this project, the resolution of input images of the model is 448×448. However, the image resolution of our dataset is larger than the model's input, so we use fix partition strategy to train our model as shown in Figure 7. We divide each of the initial 270 original images into 32 small images. In total, there are 8640 smaller images after partition. 80% of the dataset is classified into training set (6400 images) for training the model, and 20% is classified into a test set (2240 images) for testing the model.

The Danish golf dataset [18] contains 1123 orthophotos of Danish golf courses during spring with a scale of 1:1000 and a resolution of 1600×900 pixels. The orthophotos are captured from 107 different Danish golf courses, where each orthophoto captures a broad portion of the physical layout and features of the golf course. Each image is divided into six categories of background, fairway, green, tee, bunker, and water. Water, tee, bunker, and green in the dataset are small targets. Since the resolution of the image is larger than 448×448, we still need to partition the original image into six small images. From 1123 initial images, and after splitting each of it into smaller images, 80% (5454 images) of the entire data set is used as a training set for training the model, and 20% (1284 images) is used for testing the model.

The proposed model will be trained and tested on NVIDIA GeForce RTX2070 GPUs with 8GB memory and Intel(R) Core (TM) i7-10875H CPU. Each model is trained using Adam as the optimizer at a learning rate of 0.00005 and batch size of 2 small images. The training amount is set to 20 epochs on the UAVid dataset and 28 epochs on the Danish golf dataset as the Danish golf dataset needs more training to reach convergence. Based on observation, all models reach convergence at those number of epochs.

B. Testing and Evaluation

The IoU (intersection over union) measures the overlap degree of two detection frames for semantic segmentation and is used to evaluate the performance of the model (Equation 10).

$$IoU = \frac{area(predict \cap ground\ truth)}{area(predict \cup ground\ truth)} \tag{10}$$

On the UAVid dataset, image segmentation is done using the baseline model, which is EfficientNet and FPN, and both of our proposed model, HC-CNN and HPSC-CNN. From Table 1, HC-CNN and HPSC-CNN obtained good results especially in the small object category. Compared with the baseline, considerable improvement can be observed. Comparing HC-CNN and HPSC-CNN, the later shows even better performance on small objects.

Table 1. IoU on UAVid dataset

Model	Background	Building	Road	Tree	Low vegetation	Moving car	Static car	Human
Baseline	42.299%±2.513%	77.025%±2.327%	71.34%±0.912%	60.95%±6.412%	52.689%±1.791%	87.255%±3.286%	80.352%±3.287%	83.096%±3.589%
HC-CNN	42.56%±2.267%	78.069%±2.439%	71.119%±1.123%	60.832%±5.290%	**53.28%±2.485%**	87.528%±3.394%	79.567%±3.718%	83.658%±2.935%
HPSC-CNN	**44.652%±0.394%**	**78.502%±2.483%**	**71.782%±0.765%**	**61.612%±5.530%**	53.021%±2.279%	**87.676%±3.503%**	**80.613%±3.493%**	**83.699%±3.015%**

For the Danish golf dataset, similar experimental setup as outlined with the UAVid is performed where the base line models are compared with two of our proposed models. The effect of the model on improvements on small target segmentation is evident as shown in Table 2. Both HC-CNN and HPSC-CNN have significant IoU improvements compared to the baseline model, which shows that the model is suitable for small targets segmentation tasks.

Table 2. IoU on the Danish golf dataset

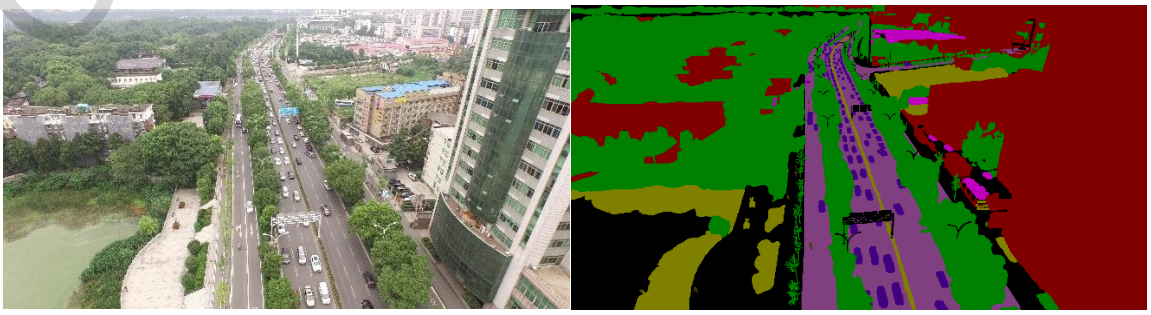
Model	Background	Green	Water	Tee	Bunker	Fairway
Backbone	88.477%	73.782%	84.858%	80.479%	89.730%	**47.581%**
HC-CNN	**88.967%**	74.581%	**85.074%**	81.387%	89.978%	46.603%
HPSC-CNN	88.814%	**75.443%**	84.940%	**82.165%**	**90.185%**	41.773%

To have a fair comparison with other related popular image segmentation models, whole image segmentation task needs to be done to get a comparable IoU values. Bilinear interpolation resize method is used to achieve whole image segmentation where new pixels is obtained by the weighted average of the pixel values in the quadrilateral surrounded by two straight lines. We first divide the UAVid image into smaller images with a pixel size of 240×240. Then the bilinear resize method is used to stretch the image to 448×448 to test the model. With this method, all pixels will be tested, and no pixels will be ignored. On the other hand, for the Danish golf dataset, the image from the dataset is first divided into smaller images of 400×300. Bilinear interpolation is then used to resize the image to a resolution of 448×448 to test and evaluate the model.

Table 3. IoU on UAVid dataset using whole images

Model	Background	Building	Road	Tree	Low vegetation	Moving Car	Static Car	Human
FCN-8s[11]	33.5	64.3	57.6	63.8	28.1	29.1	8.4	0.0
U-Net[11]	36.1	70.7	61.9	67.2	32.8	47.5	11.2	0.0
MSD[13]	57.0	79.8	74.0	74.5	55.9	62.9	32.1	19.7
CANet[14]	66.0	86.6	62.1	79.3	**78.1**	47.8	68.3	19.9
DANet[15]	64.9	85.9	77.9	78.3	61.5	59.6	47.4	9.1
CoaT[16]	**69.0**	**88.5**	80.0	79.3	62.0	70.0	59.1	18.9
SegFormer[17]	66.6	86.3	80.1	79.6	62.3	72.5	52.5	28.5
UNetFormer[13]	68.4	87.4	**81.5**	**80.2**	63.5	73.6	56.4	31.0
HC-CNN	38.3	73.8	76.1	58.6	55.5	92.0	**87.1**	**92.7**
HPSC-CNN	36.6	74.8	74.7	59.1	53.4	**92.1**	87.0	92.6

On the UAVid dataset, the IoU of our two proposed model is compared with other related image segmentation models as shown in Table 3. It can be observed that the proposed model performs very well in detecting small targets where all three small objects highest IoU is obtained by our proposed models. HC-CNN obtained highest IoU for Static Car and Human class while HPSC-CNN obtained highest IoU for the Moving Car class. Figure 8 shows the sample image segmentation result of the two proposed model on one of the images from the UAVid dataset.



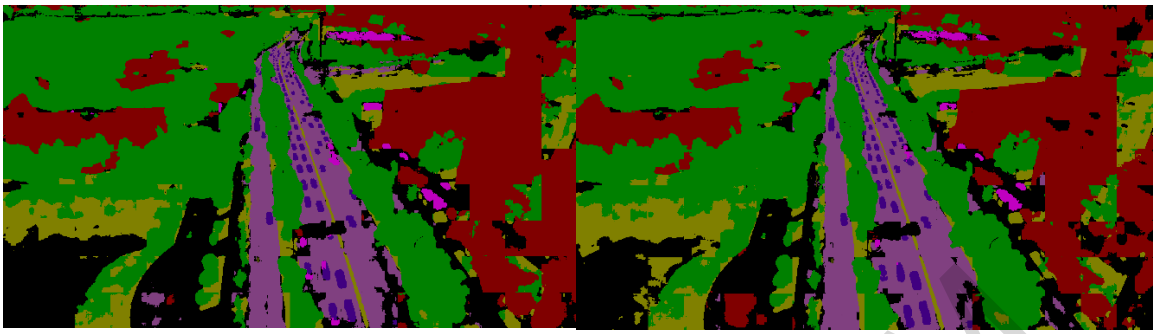


Figure 8. Top-left: the original image, Top-right: the ground truth label of the original image, Bottom-left: the segmentation result of HC-CNN, Bottom-right: the segmentation result of HPSC-CNN

Table 4. IOU on Danish golf dataset using whole images

Model	Background	Green	Water	Tee	Bunker	Fairway
Backbone		78.418%	87.436%	81.445%	90.766%	
HC-CNN		78.525%	86.464%	**82.894%**	90.826%	
HPSC-CNN		**78.981%**	**87.511%**	81.929%	**90.904%**	

Since the Danish golf dataset is a new dataset and no related papers and results is available, comparison of our proposed model is done with the baseline model. Through comparison, the methods we proposed has indeed improved the ability of small target segmentation. HC-CNN obtained the highest IoU in the Tee class while HPSC-CNN obtained the highest IoU in the Green, Water and Bunker class. Overall, HPSC-CNN has the best performance for all the classes in the small object category which proves its capability in improving segmentation of small objects.

V. CONCLUSION

Image segmentation have a huge potential for aerial images related task. There are however a number of challenges that restrict its practical implementation particularly involving small objects that have limited number of pixels and feature information. In this article, we proposed two end-to-end models to improve semantic segmentation of small objects, namely HC-CNN and HPSC-CNN. Both models use the hierarchical structure but use a different method of pixel restriction from coarse classification to fine classification. For HC-CNN, all the pixels classified in the coarse level will be constrained. On the other hand, only strong confidence pixels classified in the coarse level are constrained for HPSC-CNN. Tests on the UAVid and the Danish golf datasets shows that our proposed method is effective as considerable improvement on overall IOU could be observed especially on tiny object classes. For future tasks, a more efficient coarse-grained classification of the dataset that could better guide model for prediction and model convergence could be studied. In addition, the mechanism of the constrained map of the model can be improved so that it can better facilitate coarse-grained and fine-grained predictions and further improve the performance of the entire model.

ACKNOWLEDGMENT

This work has been supported by the Universiti Kebangsaan Malaysia Research University Grant GUP-2021-063.

REFERENCES

[1] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pp. 603-619, May 2002, doi: 10.1109/34.1000236.

[2] Jianbo Shi and J. Malik, "Normalized cuts and image segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 888-905, Aug. 2000, doi: 10.1109/34.868688

[3] J. Long, E. Shelhamer and T. Darrell, "Fully convolutional networks for semantic segmentation," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015, pp. 3431-3440, doi: 10.1109/CVPR.2015.7298965.

[4] 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. https://doi.org/10.1007/978-3-319-24574-4_28

[5] O Pinheiro, P. O., Collobert, R., & Dollár, P. (2015). Learning to segment object candidates. Advances in neural information processing systems, 28.

[6] Zhu, X., & Bain, M. (2017). B-CNN: branch convolutional neural network for hierarchical classification. arXiv preprint arXiv:1709.09890.

[7] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. ArXiv, abs/1905.11946, 2019.

[8] Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2117-2125).

[9] Abdullah, A. & Jani@Mokhtar, W. (2021). Soft Marking Scheme of SVM Hierarchical Classifiers for Attack Classification. Journal of Computer Science, 17(9), 803-814. https://doi.org/10.3844/jcssp.2021.803.814

[10] S. Nashat, A. Abdullah, S. Aramvith, and M. Z. Abdullah. 2011. Original paper: Support vector machine approach to real-time inspection of biscuits on moving conveyor belt. Comput. Electron. Agric. 75, 1 (January, 2011), 147–158. https://doi.org/10.1016/j.compag.2010.10.010

[11] Lyu, Y., Vosselman, G., Xia, G. S., Yilmaz, A., & Yang, M. Y. (2020). UAVid: A semantic segmentation dataset for UAV imagery. ISPRS journal of photogrammetry and remote sensing, 165, 108-119.

[12] Yingjie Liang, Yueying Han, and Feng Jiang. 2022. Deep Learning-based Small Object Detection: A Survey. In Proceedings of the 8th International Conference on Computing and Artificial Intelligence (ICCAI '22). Association for Computing Machinery, New York, NY, USA, 432–438. https://doi.org/10.1145/3532213.3532278

[13] Wang, L., Li, R., Zhang, C., Fang, S., Duan, C., Meng, X., & Atkinson, P. M. (2022). UNetFormer: A UNet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery. ISPRS Journal of Photogrammetry and Remote Sensing, 190, 196-214.

[14] Yang, M.Y., Kumaar, S., Lyu, Y., Nex, F., 2021a. Real-time Semantic Segmentation with Context Aggregation Network. ISPRS Journal of Photogrammetry and Remote Sensing 178, 124-134.

[15] Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., Lu, H., 2019. Dual attention network for scene segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3146-3154

[16] Xu, W., Xu, Y., Chang, T., Tu, Z., 2021. Co-Scale Conv-Attentional Image Transformers. Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 9981-9990.

[17] Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P., 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. Advances in Neural Information
[18] Jacotaco. Danish Golf Courses Orthophotos. Kaggle, https://www.kaggle.com/datasets/jacotaco/danish-golf-courses-orthophotos
[19] Madhuanand, L., Nex, F., & Yang, M. Y. (2021). Self-supervised monocular depth estimation from oblique UAV videos. ISPRS Journal of Photogrammetry and Remote Sensing, 176, 1-14.
[20] Zhang, H., Dana, K., Shi, J., Zhang, Z., Wang, X., Tyagi, A., & Agrawal, A. (2018). Context encoding for semantic segmentation. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (pp. 7151-7160).