2023 International Conference on Digital Image Computing: Techniques and Applications (DICTA) | 979-8-3503-8220-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/DICTA60407.2023.00027

Scalable Label-efficient Footpath Network Generation Using Remote Sensing Data and Self-supervised Learning

Xinye Wanyan[a], Sachith Seneviratne[a,b], Kerry Nice[a], Jason Thompson[a], Marcus White[c], Nano Langenheim[a], Mark Stevenson[a,b]

[a] Transport, Health, and Urban Design Research Lab, Melbourne School of Design, University of Melbourne, Australia
[b] Faculty of Engineering and Information Technology, University of Melbourne, Australia
[c] Centre for Design Innovation, Swinburne University of Technology, Australia
{x.wanyan, sachith.seneviratne, kerry.nice, jason.thompson, Nano Langenheim, mark.stevenson}@unimelb.edu.au, marcuswhite@swin.edu.au

***Abstract*—Footpath mapping, modeling, and analysis can provide important geospatial insights to many fields of study, including transport, health, environment and urban planning. The availability of robust Geographic Information System (GIS) layers can benefit the management of infrastructure inventories, especially at local government level with urban planners responsible for the deployment and maintenance of such infrastructure. However, many cities still lack real-time information on the location, connectivity, and width of footpaths, and/or employ costly and manual survey means to gather this information. This work designs and implements an automatic pipeline for generating footpath networks based on remote sensing images using machine learning models. The annotation of segmentation tasks, especially labeling remote sensing images with specialized requirements, is very expensive, so we aim to introduce a pipeline requiring less labeled data. Considering supervised methods require large amounts of training data, we use a self-supervised method for feature representation learning to reduce annotation requirements. Then the pre-trained model is used as the encoder of the U-Net for footpath segmentation. Based on the generated masks, the footpath polygons are extracted and converted to footpath networks which can be loaded and visualized by geographic information systems conveniently. Validation results indicate considerable consistency when compared to manually collected GIS layers. The footpath network generation pipeline proposed in this work is low-cost and extensible, and it can be applied where remote sensing images are available. Github: https://github.com/WennyXY/FootpathSeg.**

***Index Terms*—footpath segmentation, remote sensing, self-supervised learning, computer vision, deep learning, GIS.**

Fig. 1. An example of the ground truth footpath network in an area of Melbourne. The figure on the top is the ground truth footpath network and the figure on the bottom is generated by our pipeline automatically. The background map is provided by Bing Aerial.

I. Introduction

Walking as a mode of transportation provides many health and economic benefits and is crucial for accessible transport systems [1]–[4]. While city planners and local government agencies make significant investments in walking infrastructure, there is a lack of robust, real-time geolocated data capturing network detail. Such data is paramount for analysis such as walkability, safety, efficiency and maintenance of walking-oriented infrastructure.

Footpath networks require detailed and regular assessment, monitoring, and updating [5] to identify network flaws and potential risks to safety and accessibility ahead of time [6]. There are several existing efforts to detect and segment footpaths using Bird's-Eye-View (BEV) or images captured by mobile phone applications [5], [7]. However, the data availability of such crowd-sourced methods is limited because captured images depend on the paths more frequently used by application users and are difficult to extend to more regions. Instead, the development of remote sensing technology offers an opportunity as there is a large amount of open-access aerial imagery available to researchers. Therefore, the effective use of remote sensing images to generate map information has a

979-8-3503-8220-4/23/$31.00 ©2023 IEEE
DOI 10.1109/DICTA60407.2023.00027

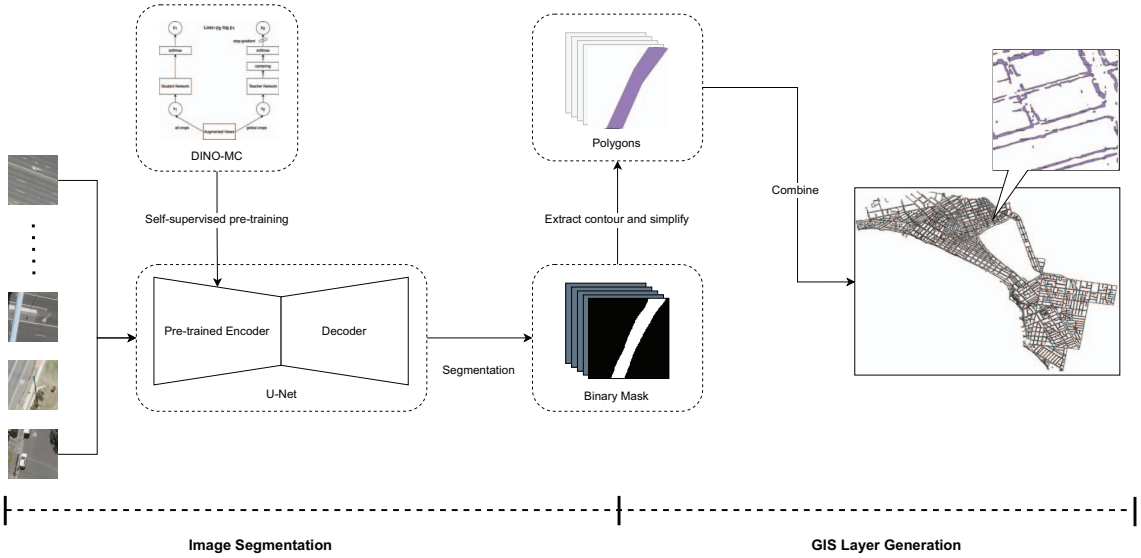

Fig. 2. The pipeline of generating geographic footpath network based on the remote sensing images. The input of the pipeline is a series of remote sensing images of a specific area and the output is a generated footpath network about this area. First, we use an unlabeled remote sensing dataset to pre-train a feature extractor in a self-supervised learning method. Then, we fine-tune the U-Net to do the downstream segmentation task on a labeled footpath remote sensing dataset. The encoder of the U-Net is a pre-trained backbone network of the self-supervised model which is mainly used to generate the representations for the input, while the decoder is a custom convolutional neural network that takes the extracted features from the encoder and reconstructs the segmentation masks with the same dimensions as the input images.

high potential for exploration. Combined with recent advances in computer vision models that have demonstrated their effectiveness in image segmentation, there is an opportunity to apply vision models to solve existing footpath identification and monitoring issues. However, annotation is very expensive, time-consuming, and error-prone, so there is a widening gap between the proliferation of satellite imagery and the limited availability of high-quality labels [8], [9]. Another problem is that the target segmentation object is obscured. Sidewalks are always adjacent to trees, so they are often partially or completely blocked in remote sensing images, i.e., canopy occlusion in the overhead images [10]. In this case, we still want our pipeline to identify the part of the footpath that is obscured in order to transform the segmented footpath into a whole network.

To address these issues, this work introduces a pipeline for constructing a footpath network from remote sensing images of a given area only requiring a minimum of manually produced annotations. As shown in Fig. 1, the real sidewalk network map is the top image, while the sidewalk map automatically generated by our proposed method is at the bottom. Considering that supervised training requires considerable labeled data, we use a self-supervised learning method to pre-train our model on an unlabeled remote sensing dataset, which is known to be an effective strategy for segmentation tasks [11]. Then, we obtain a generalizable feature extractor for the subsequent footpath segmentation task. We then attempt to solve for the problem of occluded segmented objects in terms of dataset construction. When annotating the segmentation task, we ignore the occlusion and directly generate the corresponding real footpath mask for the image. In conclusion, our contributions are: first, we create two datasets for pre-training and fine-tuning separately, then we present our pipeline which is able to generate footpath maps with minimal manual effort (Sec. III). We evaluate our segmentation model on both validation and test set and our model obtains better F1-score and mIoU results than the supervised pre-training baseline model. Our quantitative and visualized results on a remote sensing dataset in the Melbourne area are present in Sec. IV. In particular, our pipeline can be easily extended to new datasets or other map-building tasks at a very low cost.

II. Related Work

A. Self-supervised Learning

The main goal of self-supervised learning is to learn task-independent representations of the input data that can be easily generalized to downstream tasks [9]. The self-supervised model designs a pretext task whose annotations can be obtained from the dataset automatically without the need for manual annotations or a labeled dataset. Contrastive learning, a type of self-supervised method, has recently attracted a lot of attention in the field of self-supervised learning and become an essential component for natural language process, computer vision, and other domains [12]. This method does not rely on a single and specific pretext task, it aims to make similar images closer and different images far away from each other in the feature space. Therefore, it avoids learning task specific representations and is able to generalize well in

different downstream tasks such as classification, detection, segmentation, and so on. [13] proposes Bootstrap Your Own Latent (BYOL) consisting of two neural networks named online and target which take positive pairs as the input and learn from each other. A simple but effective data augmentation named multi-crop is proposed by [14] which uses a series of images of different resolutions instead of augmented images with a fixed resolution. Inspired by BYOL, [15] introduces a simple contrastive learning method with a form of self-distillation with no labels (DINO). They observe that the features extracted by a self-supervised Vision Transformer (ViT) trained by DINO deliver useful knowledge for semantic segmentation. Based on DINO and multi-crop, [16] proposes DINO-MC which uses multi-size local crops instead of a series of fixed-size local crops which is proven that the features extracted are more useful than DINO on some remote sensing tasks. There has been some work on the utilization of computer vision methods based on remote sensing imagery to generate footpath or sidewalk maps, but most of them are based on supervised methods. The application of the self-supervised model which is capable of achieving comparable results even requiring less labeled images still lacks exploration in this field. This work applies DINO-MC [16], a self-supervised learning method specialized for remote sensing imagery by considering the variation in feature sizes present in remote sensing imagery, to learn effective representations for footpath segmentation.

B. Footpath Segmentation

Physical site or aerial image surveys can generate accurate and high-quality footpath maps, but they are also a time-consuming and laborious work [17]. With the development of remote sensing technology, many automatic mapping tools have been proposed, like pedestrian Global Positioning System (GPS) trajectories and airborne Light Detection and Ranging (LiDAR). While GPS trajectories of pedestrians can result in lower costs of data collection, it has limited accuracy, in contrast, LiDAR has higher geometric accuracy but also higher cost [3]. Therefore, many existing works [18]–[21] have explored combining LiDAR technology with deep learning models. However, these methods still require specialized equipment and are labor-intensive, and our goal is to propose a generic, low-threshold, practical, and scalable pipeline for certain map generation tasks. In line with our work, a few papers focus on applying computer vision techniques only to extract footpath maps from remote sensing imagery [3], [22], [23]. However, these strategies rely on extra different views of images or a large amount of labeled data support for training, which increase the threshold for other specific applications, like combining horizontal disparity, height above ground, and angle (HHA) features with RGB-D image features to get more accurate maps than using HHA or RGB only [18]. In contrast, our work uses self-supervised learning and a much smaller dataset (1000 images or less) for training, thus considerably reducing labeling requirements. Considering the convenience and accessibility of a large amount of remote sensing image data nowadays, it is worth exploring the use of computer vision methods to automatically generate footpath maps from the remote sensing images only, without the need for large amounts of annotated data. While some work has attempted to explore this as a classification problem [24], we argue that it is important to predict segmentation results for generating fine-grained spatial insights and measurement.

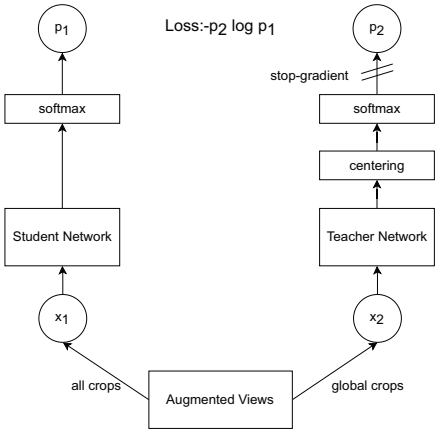

Fig. 3. The structure of DINO-MC used in this work. A self-supervised contrastive model with knowledge distillation.

III. Methodology

Our pipeline consists of two phases, image segmentation (mask generation) and GIS layer generation. To generate masks, we first pre-train a convolutional neural network for feature extraction in a self-supervised manner. Then we use an encoder-decoder model named U-Net to detect and segment the footpath mask for a specific area of remote sensing images. In the second phase, we extract the polygons from the generated masks and stitch them into a whole network according to their latitude and longitude coordinates.

A. Datasets

In this work, we build two remote sensing imagery datasets for the whole pipeline. Both datasets (the self-supervised training set and the footpath segmentation set) are downloaded from MetroMap, a provider of high-resolution aerial imagery that is updated frequently. Our datasets are collected based on XYZ Tiles, which is also known as slippy map tiles. It is a system using Web Mercator coordinates, X and Y represent the index and Z represents the zoom level of the tiles. The collected images are 256×256 pixel, and we request the slippy tiles with $Z = 21$. The training set of the self-supervised representation learning consists of 100,000 unlabeled remote sensing images of Sydney, Australia. We collect approximately 10 million remote sensing images of Melbourne in 2020 for the downstream task of sidewalk segmentation. The ground truth footpath segmentation masks are generated automatically by converting the geographic network into binary masks. We download the footpath geographic network of a small area

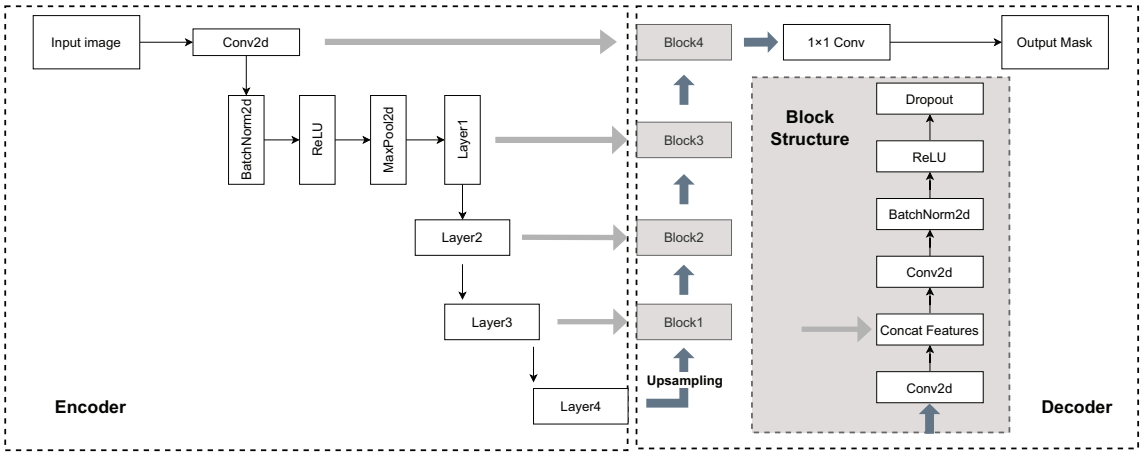

Fig. 4. The structure of the U-Net used in this work consists of two modules. The left part of the figure is the encoder module of the U-Net. In this work, we use the pre-trained Wide ResNet as the encoder to calculate the representations of the inputs. The right part of the figure is the decoder module of the U-Net. It is a custom convolutional neural network used to reconstruct the mask. The decoder module is composed of four blocks, and the structure of each block is also shown in this figure. The block takes both the output of the previous layer and the output of the encoder layer as the input and concatenates them together to process.

of Melbourne and partition this network into a series of small polygons. These polygons are segmented based on latitude and longitude coordinates which correspond to the remote sensing image tiles. The segmented polygons are then converted to binary images and used as ground truth masks for remote sensing images with the coordinates. In this way, we generated footprint masks for a total of 40,363 images. Among the labeled data, 1200 images with good annotations are manually selected, of which 1000 are used as the training set and 200 as the validation set. The remaining 39,163 labeled images are used as the test set.

B. Models

Self-supervised model. The labeling of remote sensing image tasks, especially the segmentation task, is time-consuming, labor-intensive, and requires expertise. Loading pre-trained model weights, rather than initializing with random weights, can accelerate model convergence and improve performance. However, applying the traditional supervised method to pre-train the model requires a large amount of labeled data. So in this work, we choose to use a self-supervised approach named DINO-MC to pre-train the model to learn a general feature representation for remote sensing images. After pre-training, the model is able to have an initial ability for feature extraction, which significantly reduces the amount of training data required for the downstream task. DINO-MC is a contrastive self-supervised method that does not depend on a specific pretext task. Relying on a single pretext task can only learn pretext-specific features, which may lead to poor generalization of the model. DINO-MC utilizes the knowledge distillation architecture instead of a single pretext task, to train the model to learn the relationship between the whole and the parts and capture the essential features that do not change when using various image augmentations. As illustrated in Fig. 3, there are two networks in DINO-MC named teacher and student networks which have identical architecture but different weights. The inputs of the student network include different sizes of crops (global and local crops) of the initial image, while the teacher network is only fed with two global crops of the same size. In addition to the different sizes, the input crops are applied with different and random data augmentation methods, including Gaussian Blur, color jitter, solarization, and flip. The goal of the training process is to make the image representations extracted by two networks as similar as possible in the feature space. Therefore, the model does not require any labeled datasets to learn a generic representation that is invariant to different augmentations. Compared to DINO, DINO-MC employs different sizes of local crops instead of a single size which is proven to improve the performance of the representations learned by the model on different downstream tasks [16].

Since Wide ResNet was proven to achieve better results than ResNet from the results of [16], in this task, we apply Wide ResNet as the backbone (teacher and student network) of DINO-MC.

Segmentation model. We choose U-Net, a U-shape segmentation network, to do the footpath segmentation task. It takes an image as input, outputs a dense prediction that assigns a category to each pixel, i.e., a binary mask showing where the footpath is. It mainly consists of two modules named encoder and decoder. In addition, a special structure of U-Net is the skip connection that connects the shallow features to the decoder directly. The pre-trained backbone model Wide ResNet is used as the encoder to generate the representations for the input images. The output of the first convolution layer and the last four blocks of Wide ResNet are saved and passed to the corresponding layers of the decoder module which is called skip-connection, so the semantics information can

be forwarded to deep layers. The decoder used in U-Net is a custom convolutional neural network for generating the segmentation masks by upsampling the feature maps.

Implementation details. In this section, we present more details about the implementation in the experiments. For self-supervised training, we pre-train the self-supervised model on $100,000$ unlabelled remote sensing images. The optimizer we use to update the weights of the model is adamw. During the training process, the batch size is set to 16 per GPU, and four GPUs are employed in total. Following DINO, we apply the learning rate warmup for the first 10 epochs, during which the learning rate will linearly increase. Then the learning rate starts to decrease following a cosine schedule. Following DINO-MC, we crop the input image into two global crops of the same size 224×224 and six local crops of different sizes which are $184 \times 184, 164 \times 164, 144 \times 144, 124 \times 124, 104 \times 104$, which is called multi-crop. We use the bicubic interpolation to resize images which is the same crop setting as DINO. After cropping, we apply HorizontalFlip, color jittering and GaussianBlur on the generated crops then additionally apply the Solarization on one of the global crops.

In the segmentation task fine-tuning, the implementation of the U-Net refers to the codes of SeCo [25], which is mainly based on the Pytorch Lightning, a deep learning framework. We initialize the Wide ResNet with the pre-trained weights and apply it as the encoder of the U-Net to do feature extraction. During fine-tuning, the batch size is 32 on a single GPU and the learning rate is $6e-5$. The fine-tuning process may result in feature loss, and we need to retain as many useful features learned from pre-training as possible, so the learning rate cannot be set too large. The loss function proved to be the best model in our experiments is dice_bce_loss [26], which combines the BCE (binary cross entropy) and dice coefficient. The parameters of the segmentation model are updated by two different experimental schemes. The first one is to freeze the feature extractor (Wide ResNet) to compute the representations of the input images and only adapt the weights of the decoder network. Another is to update the whole U-Net including both the pre-trained encoder and the custom decoder networks.

C. Evaluation

We perform experiments with our model by applying both end-to-end fine-tuning and the encoder-frozen fine-tuning. The main object of the assessment is the footpath segmentation results. Because of the imbalance in the number of categories (at the pixel level), the pixel accuracy is not able to accurately reflect the performance of the segmentation task. Therefore, we employ both F1-score and mean Intersection over Union (mIoU) evaluation metrics to calculate how well the generated masks match the ground truth masks.

F1-score is the harmonic mean of precision and recall. The calculation of F1-score is shown in Eq.1, 2, and 3, with TP denoting the number of positive examples that are properly predicted, FP denoting the number of positive instances that are wrongly forecasted, and FN denoting the number of negative cases that are incorrectly predicted. mIoU, also referred to as the Jaccard Index, is one of the most widely used assessment measures for segmentation tasks. As shown in Eq.4, IoU is the number of pixels that overlap between the generated segmentation mask and the ground truth mask divided by the number of their union pixels.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3}$$

$$IoU = \frac{overlap\_pixels}{union\_pixels} \tag{4}$$

D. GIS Layer Generation

After the image segmentation phase is completed, we obtain the predicted footpath masks in raster format. As shown in Fig. 2, we first extract the footpath contour from raster images and get the coordinates of the contour. We calculate the coordinates of the contours in the image and translate them into real-world latitude and longitude. The contour represented by real coordinates can then be converted into a polygon. We apply Douglas-Peucker algorithm to simplify the generated geometry by removing some points of the polygons then filter out polygons that are smaller than the threshold value in area. The generated polygons are the geometry objects and can be combined and processed according to their locations. Finally, the generated network is saved in GeoJSON files by an open-source Python tool named GeoPandas. The produced file is able to be loaded and operated as a layer of the subsequent project across multiple GIS software applications.

IV. Experiments and Results

The experiments mainly focus on the footpath segmentation to generate the binary mask for the input remote sensing imagery. We utilize two fine-tuning methods to train our model on different sizes of training sets and the fine-tuned models are evaluated on the validation set quantitatively, and there is no overlap between the training and validation sets. Two baseline models are applied to the footpath segmentation task, and we only fine-tune them on 1000 training images to compare with our model. Then we provide some visualization results of this work.

A. Quantitative Results

Comparing models fine-tuned on different-sized training sets. A machine learning model's performance is thought to be significantly influenced by the dataset size [27]. We create datasets randomly in four different sizes including 100, 400, 500, and 1000 remote sensing images, and the smaller datasets are incorporated into the larger datasets. The validation set consists of 200 images with no overlap with the training set. We load the Wide ResNet pre-trained in DINO-MC as the encoder of U-Net and experimented with two fine-tuning

TABLE I
F1-SCORES WHEN APPLYING TWO FINE-TUNING STRATEGIES ON DIFFERENT SIZES OF DATASETS. WE BUILD FOUR FOOTPATH SEGMENTATION TRAINING SETS OF DIFFERENT SIZES TO EXPLORE THE RELATIONSHIP BETWEEN THE MODEL PERFORMANCE AND THE SIZE OF THE TRAINING SET.

Dataset size (#images)	Decoder	Encoder+Decoder	
	val	val	test
100	57.00	63.11	51.06
400	62.02	71.84	59.03
500	64.33	72.14	60.25
1000	68.35	76.58	63.97

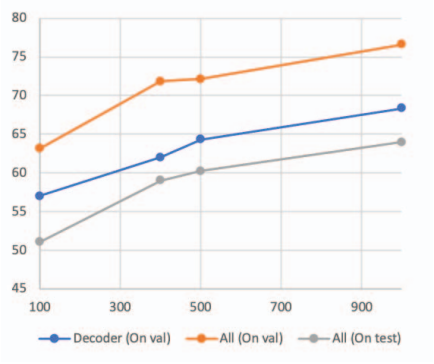

Fig. 5. F1-scores for models fine-tuned on training sets of different sizes.

modes, one is to freeze the encoder and update only the decoder, and the other is to update all parameters of U-Net. The quantitative results are shown in Tab. I. Decoder (val) is updating decoder only and evaluated on the validation set after fine-tuning. Encoder + decoder represents updating all parameters of U-Net and evaluated on the validation set and the test set respectively. The results shown in this table are the F1-score of models pre-trained on different training sets in different fine-tuning modes evaluated on the same validation set (containing a total of 200 images) and the same test set (consisting of 39,163 images in total).

Comparing the F1-scores of the two fine-tuning strategies on the validation set, updating the parameters of both encoder and decoder achieves better results in the quantitative evaluation. When fine-tuning on 100, 400, 500, and 1000 images respectively, updating all parameters achieves higher F1-scores of $6.11, 9.82, 7.81$, and 8.23 than only updating the decoder model. From Fig. 5, the F1-scores achieved by the model become progressively larger as the training data increases. When the number of images increases from 100 to 500, the improvement of the model is obvious: the F1-scores of updating all parameters increase by 9.03 on the validation set and 9.19 on the test set, the F1-score of updating the decoder parameters only increase by 7.33 on the validation set. But when the number of images increases from 500 to 1000, the F1-score of updating all parameters increase by 4.44 on the validation set and 3.72 on the test set, the F1-score of updating the decoder parameters only increase by 4.02 on the validation set. Therefore, the performance of both

TABLE II
COMPARISON BETWEEN BASELINES AND OUR MODEL.

Model	F1-score		mIoU	
	val	test	val	test
Random WRN101	63.19	50.11	46.4	33.49
ImageNet1K WRN101	69.77	55.86	53.7	38.81
DINO-MC WRN101	**76.58**	**63.97**	**62.2**	**47.09**

models improves significantly when the number of images increases from 100 to 500, but the improvement reduces when the number of images increases from 500 to 1000.

Comparing with baseline models. We experiment Wide ResNet initialized with different pre-trained weights as the encoder of U-Net to extract features of the input image. Two baseline models are utilized in this experiment. One is Wide ResNet with random weights without any pre-training, the other is Wide ResNet pre-trained on ImageNet1K in the supervised manner. Tab. II provides the quantitative results of two baseline models and our best performance model. These three models are fine-tuned and evaluated on the same footpath segmentation dataset with 1000 training images, 200 validation images, and 39,163 test images. The first two models listed in the table are the baseline models. Random WRN101 is the Wide ResNet initialized with random weights, while ImageNet1K WRN101 is the Wide ResNet pre-trained on ImageNet1K in a supervised manner. DINO-MC WRN101 is the Wide ResNet pre-trained as the backbone of DINO-MC in a self-supervised manner.

From the table, Wide ResNet pre-trained in DINO-MC achieves better results of F1-score and mIoU metrics on both validation and test sets than other two baseline models. DINO-MC WRN101 achieves 6.81 and 13.39 higher F1-score than ImageNet1k WRN101 and Random WRN101 on validation set, and 8.11 and 13.86 higher F1-score on test set. DINO-MC WRN101 achieves 8.5 and 15.8 higher mIoU than ImageNet1k WRN101 and Random WRN101 on validation set, and 8.28 and 13.6 higher mIoU on test set.

B. Visualization

We use the fully fine-tuned U-Net with Wide ResNet pre-trained in DINO-MC for mask generation which achieves the best quantitative results on both validation and test sets. Fig. 6 shows five instances of the footpath segmentation, where the first column is the input of the model (original remote sensing images), the second column is the ground truth mask, and the last column presents the output mask of the segmentation model. From the visualization results of the generated masks, we can find that our model is capable of detecting the footpath from the remote sensing imagery and restoring occluded and missing sidewalks even in the presence of tree canopy occlusion (see the third row in Fig. 6).

Fig. 7 presents the entire GIS footpath network map with the ground truth footpath layer at the top and the generated footpath map layer at the bottom. We observe that the generated map is able to depict the whole network structure of this area, which is quite similar to the ground truth map. In

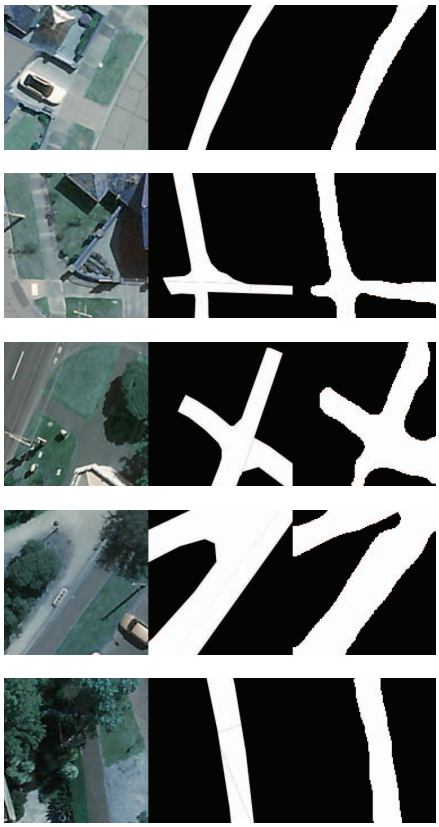

Fig. 6. Examples of the mask output from the best performing fine-tuning model. The model is pre-trained as the backbone of DINO-MC in a self-supervised manner and then fine-tuned on footpath segmentation task training set with 1000 remote sensing images. The first column (from left to right) shows the original remote sensing images, the second row shows the ground truth masks, and the third row shows the generated binary masks.

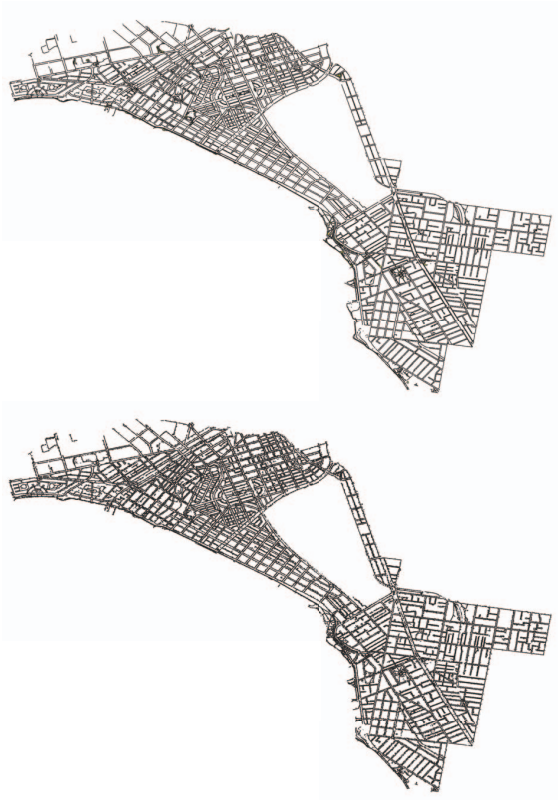

Fig. 7. Visualization of the whole GIS footpath networks. Top: the ground truth footpath network in an area of Melbourne. Bottom: the final generated footpath network using our pipeline.

conclusion, our model can detect the location and shape of the footpath from remote sensing images, but there are prediction errors in the specific width as well as in the edges.

C. Discussion and Error Analysis

From the quantitative results, we find that pre-training on larger datasets can produce better results, but this improvement reduces as the amount of data increases. Updating all the parameters during fine-tuning phase achieves better results than freezing the encoder model, since during this process, the features extracted by the encoder generalize to the specific footpath downstream task which helps the encoder construct better segmentation mask.

We observe that when updating all parameters of U-Net, the improvement resulting from increasing the size of the training set is greater than when updating only the decoder module. One possible reason is that when freezing the encoder module, the fine-tuned model contains only the decoder, which can be seen as training a smaller model than the whole U-Net, and the feature learning ability of the smaller model is to some extent more limited compared to the larger model. Compared to the random initialized Wide ResNet, our self-supervised Wide ResNet gains better performance which proves the effectiveness of the pre-training process again. We even outperform the supervised baseline model showing the large potential of applying self-supervised learning in GIS map generation based on remote sensing imagery.

From the visualization results, our pipeline is able to identify the shape and location of the footpath, and the errors are mainly in the prediction of the specific footpath's precise width prediction and the edge segmentation details. The possible reason could be that the zoom level of the remote sensing imagery tiles we use for segmentation input is large, and each image only covers a small area. Therefore, for each image, the segmentation model can only observe and learn less contextual information. Another possible reason is the problem of occluded segmented objects. Due to the camera angle, lighting problems, and the diversity of occlusions, the model suffers more interference in dealing with the contour of the footpath.

In the future - by combing our outputs with existing data collection processes such as OSM, and Captial Works programs (within Councils) - the output quality will be progressively improved enabling increasingly robust urban analytic

outputs and improving data drived urban design decision-making.

V. Conclusion

This paper has shown the potential of applying a self-supervised model to footpath map generation only using remote sensing imagery. We propose a pipeline for generating a geographic footpath map only based on the corresponding remote sensing images. First, we employ a self-supervised learning model DINO-MC to train the Wide ResNet to learn general feature representations. Then we load the pre-trained Wide ResNet as the encoder of the segmentation model U-Net and fine-tune it on the footpath segmentation task. After training, the best-performing fine-tuned model is applied to the remote sensing imagery of a specific region to obtain the generated masks of raster format. Based on the masks, we extract the contour of the predicted footpath and convert them to polygons, which are saved in the GeoJSON files for the following application. Our approach is highly automated, has a low threshold, and is ready to extend to other datasets or applications. Our model achieves better F1-score and mIoU than the supervised WideResNet baseline model.

Acknowledgment

This research was supported by The University of Melbourne's Research Computing Services and the Petascale Campus Initiative. This research was supported (partially or fully) by the Australian Government through the Australian Research Council's Centre of Excellence for Children and Families over the Life Course (Project ID CE200100025). This grant is supported by National Health and Medical Research Council Grant 1194959 – A Vision of Healthy Urban Design.

References

[1] L. D. Frank, J. F. Sallis, T. L. Conway, J. E. Chapman, B. E. Saelens, and W. Bachman, "Many pathways from land use to health: associations between neighborhood walkability and active transportation, body mass index, and air quality," *Journal of the American planning Association*, vol. 72, no. 1, pp. 75–87, 2006.

[2] R. H. Lo, "Walkability: what is it?" *Journal of Urbanism*, vol. 2, no. 2, pp. 145–166, 2009.

[3] M. Hosseini, A. Sevtsuk, F. Miranda, R. M. Cesar Jr, and C. T. Silva, "Mapping the walk: A scalable computer vision approach for generating sidewalk network datasets from aerial imagery," *Computers, Environment and Urban Systems*, vol. 101, p. 101950, 2023.

[4] M. White, X. Huang, N. Langenheim, T. Yang, R. Schofield, M. Young, S. Livesley, S. Seneviratne, and M. Stevenson, "Why are people still not walking? the need for a micro-scaled multi-criteria spatio-temporal design approach to improve walk-quality." *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 10, 2022.

[5] V. GM, B. Pereira, and S. Little, "Urban footpath image dataset to assess pedestrian mobility," in *Proceedings of the 1st International Workshop on Multimedia Computing for Urban Data*, 2021, pp. 23–30.

[6] S. Fotios and J. Uttley, "Illuminance required to detect a pavement obstacle of critical size," *Lighting Research & Technology*, vol. 50, no. 3, pp. 390–404, 2018.

[7] N. Gosala, K. Petek, P. L. Drews-Jr, W. Burgard, and A. Valada, "Skyeye: Self-supervised bird's-eye-view semantic mapping using monocular frontal view images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 901–14 910.

[8] S. Seneviratne, K. A. Nice, J. S. Wijnands, M. Stevenson, and J. Thompson, "Self-supervision. remote sensing and abstraction: Representation learning across 3 million locations," in *2021 Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 2021, pp. 01–08.

[9] Y. Wang, C. M. Albrecht, N. A. A. Braham, L. Mou, and X. X. Zhu, "Self-supervised learning in remote sensing: A review," *arXiv preprint arXiv:2206.13188*, 2022.

[10] T. Senlet and A. Elgammal, "Satellite image based precise robot localization on sidewalks," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 2647–2653.

[11] H. Thisanke, C. Deshan, K. Chamith, S. Seneviratne, R. Vidanaarachchi, and D. Herath, "Semantic segmentation using vision transformers: A survey," *arXiv preprint arXiv:2305.03273*, 2023.

[12] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, Dec 2020. [Online]. Available: http://dx.doi.org/10.3390/technologies9010002

[13] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar *et al.*, "Bootstrap your own latent-a new approach to self-supervised learning," *Advances in neural information processing systems*, vol. 33, pp. 21 271–21 284, 2020.

[14] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in neural information processing systems*, vol. 33, pp. 9912–9924, 2020.

[15] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.

[16] X. Wanyan, S. Seneviratne, S. Shen, and M. Kirley, "Dino-mc: Self-supervised contrastive learning for remote sensing imagery with multi-sized local crops," *arXiv preprint arXiv:2303.06670*, 2023.

[17] F. R. Proulx, Y. Zhang, and O. Grembek, "Database for active transportation infrastructure and volume," *Transportation research record*, vol. 2527, no. 1, pp. 99–106, 2015.

[18] J. Schlosser, C. K. Chow, and Z. Kira, "Fusing lidar and images for pedestrian detection using convolutional neural networks," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 2198–2205.

[19] D. Matti, H. K. Ekenel, and J.-P. Thiran, "Combining lidar space clustering and convolutional neural networks for pedestrian detection," in *2017 14th IEEE international conference on advanced video and signal based surveillance (AVSS)*. IEEE, 2017, pp. 1–6.

[20] J. Alfred Daniel, C. Chandru Vignesh, B. A. Muthu, R. Senthil Kumar, C. Sivaparthipan, and C. E. M. Marin, "Fully convolutional neural networks for lidar–camera fusion for pedestrian detection in autonomous vehicle," *Multimedia Tools and Applications*, pp. 1–24, 2023.

[21] Q. Hou and C. Ai, "A network-level sidewalk inventory method using mobile lidar and deep learning," *Transportation research part C: emerging technologies*, vol. 119, p. 102772, 2020.

[22] J. Luo, G. Wu, Z. Wei, K. Boriboonsomsin, and M. Barth, "Developing an aerial-image-based approach for creating digital sidewalk inventories," *Transportation research record*, vol. 2673, no. 8, pp. 499–507, 2019.

[23] H. Ning, X. Ye, Z. Chen, T. Liu, and T. Cao, "Sidewalk extraction using aerial and street view images," *Environment and Planning B: Urban Analytics and City Science*, vol. 49, no. 1, pp. 7–22, 2022.

[24] S. Seneviratne, J. S. Wijnands, K. Nice, H. Zhao, B. Godic, S. Mavoa, R. Vidanaarachchi, M. Stevenson, L. Garcia, R. F. Hunter *et al.*, "Urban feature analysis from aerial remote sensing imagery using self-supervised and semi-supervised computer vision," *arXiv preprint arXiv:2208.08047*, 2022.

[25] O. Manas, A. Lacoste, X. Giró-i Nieto, D. Vazquez, and P. Rodriguez, "Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9414–9423.

[26] L. Zhou, C. Zhang, and M. Wu, "D-linknet: Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 192–1924.

[27] A. Althnian, D. AlSaeed, H. Al-Baity, A. Samha, A. B. Dris, N. Alzakari, A. Abou Elwafa, and H. Kurdi, "Impact of dataset size on classification performance: an empirical evaluation in the medical domain," *Applied Sciences*, vol. 11, no. 2, p. 796, 2021.