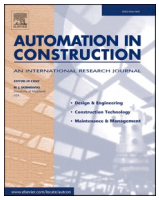
Automation in Construction 154 (2023) 104983

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon



Transformer-based automated segmentation of recycling materials for semantic understanding in construction

Xin Wang [a], Wei Han [a], Sicheng Mo [b], Ting Cai [c], Yijing Gong [d], Yin Li [e], Zhenhua Zhu [a,*]

[a] Department of Civil and Environmental Engineering, University of Wisconsin-Madison, 1415 Engineering Drive, Madison, WI 53706, USA
[b] Department of Computer Science, University of California, Los Angeles, 404 Westwood Plaza, Engineering VI, Los Angeles, CA 90095, USA
[c] Department of Computer Sciences, University of Wisconsin-Madison, 1210 W. Dayton Street, Madison, WI 53706, USA
[d] Department of Animal and Diary Sciences, University of Wisconsin-Madison, 1675 Observatory Drive, Madison, WI 53706, USA
[e] Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, 1300 University Ave, Madison, WI 53706, USA

ARTICLE INFO

Keywords:
Construction image segmentation
Systematic evaluation
Transformer-based architectures
Ensemble learning
Model averaging

ABSTRACT

Construction sites are incorporating cameras to gather imagery data for project management. While transformer-based deep models show promise in recognizing construction objects and understanding the environment, their use in construction images is largely unexplored. This paper presents a systematic evaluation of three state-of-the-art transformer-based models for automatic segmentation and recognition of construction images. Further, a two-stage model ensembling strategy based on model averaging and probability weighting is introduced and implemented for performance improvement. A dataset containing five classes of recycling materials on construction sites is created as a benchmark to compare their performance. The comparison results indicate the ensemble model could achieve encouraging results with a mIoU of 82.36% and mPA of 90.30%, which demonstrate superior segmentation performance on construction images.

1. Introduction

It has been common to establish a video camera network on a construction site to acquire imagery data due to the rapid development of digital camera technology [1]. Availability of these data provides opportunities for promoting automation [2], improving productivity [1], solving safety issues [3], etc. in construction. They create essential visual documentation that will serve the project team during construction, such as onsite monitoring [1,4], progress tracking [5]. Moreover, the learnings from processing site images (e.g., the cues for identifying potential hazards [6], the features for recognizing equipment activities [7]) can be transferred easily to benefit future projects.

Techniques for automatic segmentation and recognition of construction objects have provided a powerful tool to analyze these image data. They are capable of segmenting and recognizing regions of various objects from an input construction image, and thus offering novel means for the visual understanding of construction environment. For example, these segmentation techniques can enable onsite robots to conduct comprehensive analyses of the surrounding environmental conditions for performing safe navigation [8]. Also, they offer a convenient way for aiding human workers to rapidly inspect infrastructure health, such as assessing surface defects [9], and identifying damaged components [10].

So far, there are many research studies proposed for image semantic segmentation tasks. They either relied on classical machine learning classifiers (e.g., Support Vector Machine (SVM) [11], Random Forest [12]), or deep neural networks (e.g., convolutional neural networks (CNNs) [13] using encoder-decoders [14]). The performance of these methods was evaluated on several public datasets, such as COCO [15] and ADE20K [16]. The results demonstrated superior performance using deep neural networks and have thus validated the effectiveness of these deep networks. Among deep learning methods, recently proposed transformer-based architectures that leverage self-attention mechanism can model long-range dependencies in an image [17], which refer to the relationships between pixels or image regions that are separated by a significant distance from each other. Transformers have demonstrated impressive segmentation performance in various applications, such as tasks in medical image analysis [18] and autonomous driving [19].

Although the performance of transformer-based segmentation is promising, its applications to construction images remain largely

* Corresponding author.
E-mail addresses: xwang2463@wisc.edu (X. Wang), whan59@wisc.edu (W. Han), smo3@ucla.edu (S. Mo), tcai35@wisc.edu (T. Cai), gong44@wisc.edu (Y. Gong), yin.li@wisc.edu (Y. Li), zzhu286@wisc.edu (Z. Zhu).

https://doi.org/10.1016/j.autcon.2023.104983
Received 25 January 2023; Received in revised form 24 May 2023; Accepted 10 June 2023
Available online 17 June 2023

unexplored. This motivates us to study the feasibility of using transformer-based models for construction image segmentation tasks. This paper presents a systematic evaluation of transformer-based architectures and their ensemble model on construction image segmentation tasks. It starts with the literature review of existing image semantic segmentation methods and their applications in construction. Then, three state-of-the-art transformer-based architectures (Swin Transformer, Twins Transformer, and K-Net) are selected and their ensemble model is implemented based on a two-stage model ensembling strategy. A novel dataset containing 5 classes of recycling materials on construction sites is utilized and employed as a benchmark. The models are evaluated in terms of mean Intersection over Union (mIoU) and mean pixel accuracy (mPA), widely considered for semantic segmentation tasks. The evaluation results demonstrate superior semantic segmentation performance of transformer-based architectures on construction images.

The contributions of the paper are summarized as follows. First, we evaluated systemically existing transformer-based segmentation models on construction image samples. Second, we proposed a novel two-stage model ensembling framework built upon the performance analysis of existing transformer-based segmentation models. The framework takes advantage of the strength of existing transformer-based segmentation models and demonstrates significantly improved overall segmentation performance. To our knowledge, none of the existing research studies conducted such systematic evaluations and explored the adaptation and performance of transformer-based segmentation models in the context of construction image segmentation. Also, none of the existing research studies proposed the model ensembling framework as ours to overcome the weakness of individual transformer-based segmentation models when segmenting construction images with different types of materials.

2. Related work

2.1. Applications of semantic segmentation techniques in construction

Image semantic segmentation techniques have been applied to improve the construction industry in many aspects, such as safety, automation, and inspection. Considerable research focused on improving construction and building safety by vision-based techniques. For example, Yu et al. [20] located construction workers by Fast R-CNN and detected openings by DeepLabv3+ for the real-time monitoring of the construction site, aiming to prevent workers from falling. Wang et al. [21] proposed a novel method based on YOLOv3, SENet and thresholding segmentation algorithm to detect tower crane rust defects for improving construction safety. Bang et al. [22] developed a proactive proximity-monitoring method applied on Unmanned Aerial Vehicle (UAV) collected videos to prevent struck-by accidents on construction sites. The method segmented construction objects by Mask R-CNN and predicted their future trajectories by convolutional long short-term memory (LSTM) for avoiding dangerous situations. Zhou et al. [23] implemented a Mask R-CNN based segmentation framework to automatically detect and estimate the fire loads of buildings.

There has been a recent surge of interest in developing construction robots for improving the automation of construction projects. To introduce robots into construction sites, scene understanding plays a crucial role since it allows the on-site robots to recognize the surrounding environment and maneuver by themselves. Asadi et al. [24] proposed a deep neural network architecture for real-time pixel-wise semantic segmentation to make camera-equipped UAVs understand their surrounding environments. Atkinson et al. [25] presented a two-stage transfer learning method based on Mask R-CNN to allow robots autonomously inspect the underfloor voids. Wang et al. [26] developed a synthetic robotic system integrated with a semantic segmentation model (DeepLabv3+) for visual understanding on sites.

Besides, the segmentation techniques offer a convenient way to rapidly inspect infrastructure quality and health. For example, Pi et al. [10] evaluated the performance of Mask R-CNN and Pyramid Scene Parsing Network (PSPNet) on a fully annotated dataset named Volan2019 for detecting and segmenting critical objects in the aerial footage of disaster sites. Wang and Su [27] presented a deep learning architecture for real-time crack segmentation on pavement images. Wang et al. [28] developed a deep learning-based image segmentation model to evaluate the conditions of unreinforced masonry buildings from street view images. With this evaluation, the model could understand whether the buildings were vulnerable to strong ground motions. Xie et al. [9] proposed a model which combined sparse-sensing-based encoder and superpixel-based decoder for concrete crack segmentation.

2.2. Semantic segmentation

Semantic segmentation aims to predict the labels of each pixel of the input image from a given label set. Classical machine learning techniques, such as SVM [11], K-means Clustering [29], and Random Forest [12], were used to solve the problem of image segmentation. For example, Dhanachandra et al. [29] segmented images by a modified k-clustering algorithm, which used subtractive clusters to generate the initial centroids. Kang and Nguyen [12] presented a Random Forest framework that learned the flexible filters using an iterative optimization algorithm and segmented input images using the learned representations.

The advent of deep learning models such as CNNs [13], LSTM [30], encoder-decoders [14] and generative adversarial networks (GANs) [31] has boosted the performance of semantic segmentation. For instance, Long et al. [13] defined a novel fully convolutional network architecture that combined semantic information from a deep, coarse layer with appearance information from a shallow, fine layer to produce accurate and detailed segmentations. Badrinarayanan et al. [14] designed a deep neural network architecture for semantic pixel-wise segmentation termed SegNet. This core trainable segmentation engine consisted of an encoder network, and a corresponding decoder network followed by a pixel-wise classification layer. Moreover, several recent works [32–34] combined semantic segmentation with weak supervision, and showed the ability of learning from weaker forms of labels such as classes, tags, bounding boxes, scribbles, and point annotations.

Currently, CNNs are the basic building blocks of most methods proposed for image segmentation. However, they lack the ability to model long-range dependencies present in an image, which can arise when there are complex spatial structures or patterns that require modeling interactions between distant regions. In many Natural Language Processing (NLP) applications, transformers have shown the ability to encode long-range dependencies due to the self-attention mechanism, which finds the dependency between given sequential input [17]. Following their popularity in NLP settings, transformers have been adopted to computer vision applications very recently [35,36]. With regard to image segmentation problems, various transformer-based architectures such as Swin Transformer [37], CSWin Transformer [38], Twins Transformer [39], K-Net [40], Masked-attention Mask Transformer (Mask2Former) [41], etc. have been proposed to achieve a higher accuracy and/or save more training efforts. They have exhibited excellent segmentation performance in various fields, such as segmentation tasks in medical image analysis [18], remotely sensed urban images [42], autonomous driving [19], etc.

2.3. Comparison study

Along with large amount of transformer architectures emerges, many comparison studies have also been conducted to examine the performance of those methods with previous state-of-the-art architectures under the scope of semantic segmentation. Liu et al. [37] who proposed the architecture of Swin Transformer, also compared its performance with other state-of-the-art deep learning architectures. They found that the UperNet with Swin backbone increased the mIoU by 5.3% on the

ADE20K dataset compared to the previous transformer-based method: UperNet with DeiT backbone. It was also 4.4% mIoU higher than all methods with ResNet-101 backbone, a cutting edge convolutional-based backbone. In the work of Chu et al. [39], the comparison results showed that their proposed transformer-based architecture: Twins Transformer clearly outperformed PVT and ResNet on ADE20K and obtained slightly better performance compared to Swin Transformer. Dong et al. [38] presented CSWin Transformer which achieved an impressive 55.7% mIoU on the ADE20K segmentation task, surpassing the Swin Transformer by 2.2% mIoU with the same complexity. When compared to convolutional-based backbone: Res-101, the increase came to 6.6% mIoU with even less parameters. Cheng et al. [41] compared their proposed architecture: Mask2Former to both ResNet and Swin backbones. The results indicated that Mask2Former outperformed Mask R-CNN with 8× less training epochs on COCO segmentation task and achieved 3.5% mIoU higher than Swin architecture.

Several findings have been noted from existing comparison studies. First, transformer-based architectures have gradually become mainstream in segmentation tasks, replacing the dominance place of other deep neural networks due to the following reasons. Transformers enable long-term dependencies across image regions compared to CNNs. Also, they require minimal inductive biases, which means that transformers do not have any built-in assumptions or preconceptions about the data they process compared to CNNs and RNNs. This practice is beneficial especially when the data amount is large. In addition, transformers have an edge over RNNs in computational requirement since the transformer blocks support parallel processing of sequence elements compared to recurrent structures [43]. Second, large-scale general-purpose labeled image datasets such as COCO and ADE20K play a crucial role in existing studies. However, less attention has been paid to creating subject-specific datasets, except for the medical domain, where MSD [44] has been built.

3. Research gap, objective and scope

As illustrated in the literature review, the performance of transformer-based architectures is promising and impressive in various semantic segmentation tasks. However, their applications to construction images remain largely unexplored in the following aspects. First, there lacks an evaluation and comparison of state-of-the-art transformer-based architectures on construction image segmentation tasks. Second, most of the previous studies focused on solving the segmentation problems by a single transformer-based model [18,19,36]. Since ensemble learning has been proven effective to improve the predictive performance of a single model [45], it is necessary to implement a transformer-based ensemble model for better segmentation results.

The main objective of this study is to explore the above aspects. To this end, this paper presents a systematic evaluation of state-of-the-art transformer-based architectures on construction semantic segmentation tasks. Three architectures including Swin Transformer, Twins Transformer, K-Net are chosen here since they exhibited superior segmentation performance among various transformer-based techniques. Further, their ensemble model based on model averaging and probability weighting is implemented to achieve better segmentation results. A novel benchmark dataset containing 5 classes of recycling materials on construction sites is created. The recycling materials include rebar, bricks (full or broken), PVC pipes, plastic wires, and debris. They are selected as the study objects since the segmentation results of construction recycling materials can provide important information for waste management. The segmentation models could be expanded to work for other construction objects without the loss of generality. Finally, the performance of all the models is measured in terms of mIoU and mPA to provide an in-depth analysis of their benefits and limitations.

4. Systematic evaluation

4.1. Selection of state-of-the-art transformer-based architectures

Based on the findings from existing image segmentation studies [36–38], transformer-based architectures illustrated an excellent segmentation performance due to their ability to enable long-term dependencies and model highly representative features. Thus, in this study, the selection of semantic segmentation methods for testing is focused on state-of-the-art transformer-based architectures. Specifically, three architectures, namely, Swin Transformer [37], Twins Transformer [39], and K-Net [40] are considered, since they illustrated better segmentation performance in terms of accuracy and training efforts than other transformer-based architectures, such as SEgmentation TRansformer (SETR), Pyramid Vision Transformer, Mask Transformer, Panoptic SegFormer, etc.

In this study, Swin Transformer refers to an UperNet [46] based segmentation framework which utilizes Swin as the backbone. The Swin is a hierarchical transformer architecture, which computes representations with shifted windows. The shifted windowing scheme brings greater efficiency by limiting self-attention computation to non-overlapping local windows while also allowing for cross-window connection. Twins Transformer is an UperNet based framework with Twins serving as the backbone. The Twins is a type of vision transformer that utilizes a spatially separable attention mechanism (SSAM). The SSAM is composed of two types of attention operations to capture the short-distance information and global information, separately. K-Net refers to a unified segmentation framework which introduces a group of learnable kernels to generate the masks for either potential instances or stuff classes. The Swin is served as the backbone in K-Net. Table 1 summarizes the complexity of the above architectures. More details of these architectures could be found in the following references [37, 39, 40].

4.2. Evaluation metrics

The segmentation performance is evaluated in terms of mIoU and mPA on a test set. The IoU is the most commonly accepted metric for image semantic segmentation. It measures the number of pixels common between the ground truth mask and prediction mask divided by the total number of pixels present across both masks, which is defined as Eq. (1).

$$IoU = \frac{\text{Ground truth mask} \cap \text{Prediction mask}}{\text{Ground truth mask} \cup \text{Prediction mask}} \tag{1}$$

The IoU score is calculated for each class separately and then averaged over all classes to provide a global mIoU score of our semantic segmentation prediction. Besides, PA provides an alternative metric to evaluate semantic segmentation results. It reports the percentage of pixels in the image which are correctly classified, defined as Eq. (2).

$$PA_i = \frac{\#TP_i + \#TN_i}{\#TP_i + \#TN_i + \#FP_i + \#FN_i} \tag{2}$$

where TP_i represents pixels that are correctly predicted to belong to the given class i, TN_i means pixels that are correctly identified as not belonging to class i, FP_i refers to pixels classified incorrectly as class i and FN_i represents pixels classified incorrectly as not class i. The mPA takes the average value of PA across all the classes to provide a global evaluation.

Table 1
Complexity summary of the selected architectures.

Architectures	Crop Size	# Params (M)	FLOPs (G)	Memory (GB)
Swin Transformer	512 × 512	121	1188	8.52
Twins Transformer	512 × 512	133	297	8.41
K-Net	512 × 512	208	–	13.50

5. Two-stage model ensembling

Typically, a comparison study ends with selecting a single model that achieves the best performance and discarding the remaining models. This may reduce out-of-distribution performance [47], which measures the model's ability to make accurate predictions on examples that are outside of its training distribution. Further, performance improvement can be typically achieved by combining multiple models. Ensemble learning is a machine learning technique that combines the predictions from two or more models. It has been proven effective to make more accurate predictions than any single contributing model [45,48]. Moreover, the ensemble model helps reduce the spread or dispersion of the predictions. These benefits of ensemble learning motivate us to implement a transformer-based ensemble model to improve segmentation results in this study.

A two-stage model ensembling framework is designed as shown in Fig. 1. The two stages are model averaging and probability weighting, separately. Specifically, the raw image is first input into the model soups to get their initial predictions. Here, the model soup of each transformer-based architecture is formed by averaging the weights of different trained models of the same architecture after training. Then, the probability weighting is applied to process the outputs from the model soups to obtain the final prediction result.

The idea of model averaging is to average different trained models' weights for each transformer-based architecture along a single training trajectory. The averaged model is termed as model soup. It has previously been shown to improve the performance of models in both non-transfer [49] and transfer [48] learning settings. The procedure of model averaging is conducted as follows: $f(x,\theta)$ is considered as a neural network with input data x and parameters $\theta \in R^d$, where d is the parameter dimension. Along one single training trajectory, there are multiple checkpoints produced, which are periodic snapshots of the model parameters. Assume that n checkpoints $f(x,\theta_k), k = 1,2,\ldots,n$ are extracted from the training process and sorted in descending order of validation subset accuracy. A model list is used to store the checkpoints needed for the model soup. The flowchart of generating the model list is shown in Fig. 2. The model list is initialized with the checkpoint $f(x,\theta_1)$ which has the highest validation accuracy. Then, from $k = 2$ to $k = n$, the performance between the model that averages the weights of $f(x,\theta_k)$ and the checkpoints in the model list and the one which averages all the checkpoint weights in the model list is compared. If the performance of the previous model is better, $f(x,\theta_k)$ will be added into the model list; otherwise, $f(x,\theta_k)$ will be disregarded. The final model list is generated when all the n checkpoints have repeated the above process. Therefore, the model soup will be obtained by averaging the weights of all the checkpoints in the model list.

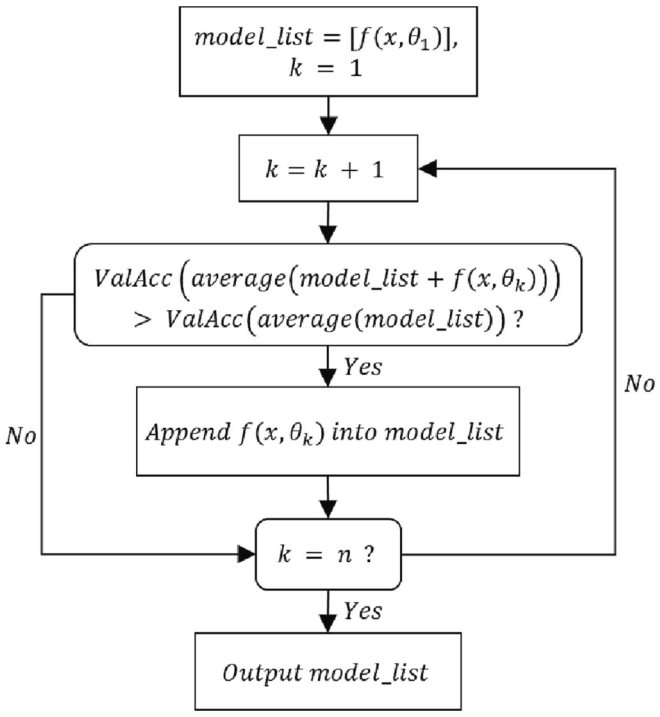


Fig. 2. Flowchart of generating the model list.

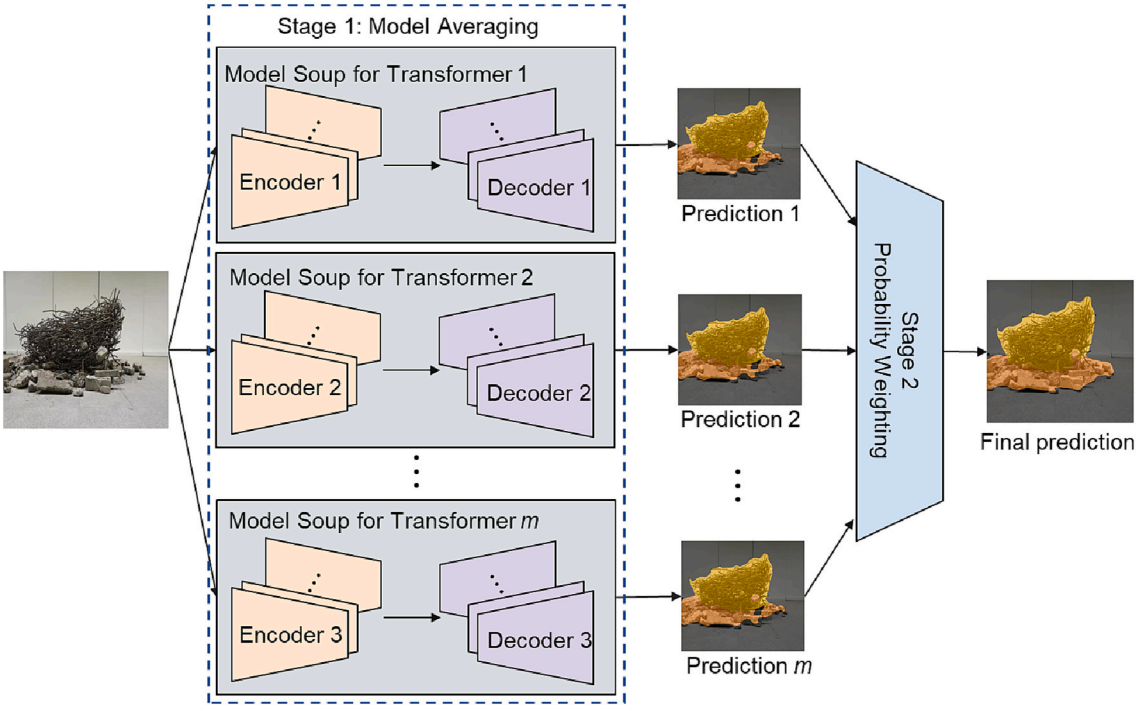


Fig. 1. Two-stage model ensembling framework.

The purpose of probability weighting is to post process the outputs from multiple model soups to obtain the final prediction probabilities. It is achieved by assigning weights to each model soup. The probability weighting approach is most suitable for cases where the performance of the base models is comparable [45]. In this study, the weighting of the model soups is performed on their outputs via the following equation:

$$P_i = \frac{\sum_j w_j O_i^j}{\sum_i \left(\sum_j w_j O_i^j \right)} \tag{3}$$

where P_i is the final probability outcome of the i-th class, w_j is the assigned weight for the j-th model soup, O_i^j is the output of the i-th class for the j-th model soup. The assigned weights are proportional to the strengths of the model soups and determined by grid search [50], which is an exhaustive searching through a manually specified subset of the weight space. In this study, the weight space for each transformer-based architecture starts from 0 to 10 with the step value of 1. After probability weighting, the class with the highest probability in P_i will be chosen as the final predication.

6. Implementation and results

6.1. Datasets

To evaluate the segmentation performance, a novel dataset containing images of recycling materials on construction sites is created and employed as the benchmark. The dataset includes five classes of recycling materials in total, which are rebar, bricks, PVC pipes, plastic wires, and debris. The recycling materials are selected as the segmentation objects since timely and accurate identification of construction recycling materials can provide yardstick information for their subsequent management. It is of significant value for managing the recycling material schemes on construction sites (e.g., determining admissibility and chargeable levy) [51]. Further, the ability to recognize specific material types and positions makes it possible to replace human workers with intelligent robots for sorting recycling materials automatically [52].

The images coming from different data sources are collected to establish the dataset. These data sources include VIMS-IAARC competition committee [53], the self-collected database, and Google Image, which provided 1109, 444 and 366 images, separately. To augment the diversity of the dataset, the self-collected database also contains synthetic images. These synthetic images were generated manually by matting the foreground objects (e.g., plastic wires) and then putting them onto the relevant background images. Examples of the synthetic images could be found in Fig. 3. In total, 37 synthetic images are included in the dataset, which contain about 1.20×10^5 pixels. These synthetic images could save the efforts to collect real-word images which are difficult to get and minimize the bias in the dataset.

In total, the dataset contains 1919 images which are distributed in 5 classes. The details of the dataset are listed in Table 2. The numbers of images for rebar, bricks, PVC pipes, plastic wires, and debris are 541, 324, 490, 456 and 400, respectively. Among these 5 classes, the minimum and maximum number of pixels are 9.52×10^7 (bricks) and 1.68×10^8 (debris), respectively. The average resolution of an image is about 620 (height) × 770 (width). Examples of the collected data could be found in Fig. 4. Further, the pixel-level labeling is conducted manually for all the collected images via VGG Image Annotator (VIA) [54] as shown in Fig. 5.

While public datasets like COCO provide a wide range of labeled images for object segmentation and recognition tasks, they may not be suitable for specific domains such as construction due to the absence of any construction-related objects. Construction sites exhibit unique challenges for automated image segmentation, including complex and dynamic environments, varying lighting conditions, and diverse object types. Compared to other public datasets, the dataset we considered contains not only voluminous objects (e.g., bricks), but also long and narrow objects (e.g., rebar, wires) as well as fragmental objects (e.g., debris). Models trained on construction domain dataset can achieve higher accuracy in the segmentation and recognition of construction objects. Moreover, they are likely to perform better on new construction sites because they are trained on data that is more representative of the domain.

6.2. Performance evaluation

6.2.1. Implementation

The transformer-based networks and their ensemble model are implemented on an Ubuntu Linux 64-bit operating system. The Python 3.8 environment with the support of the Pytorch [55] platform provides the critical algorithms, functions, and tools required for the networks. The hardware configuration includes an AMD Ryzen 5800X CPU (Central Processing Unit) @ 3.80 GHz, a 64 GB memory, and an NVIDIA GeForce RTX 3090 @ 24.0 GB GPU (Graphic Processing Unit).

6.2.2. Training

In order to train and test image semantic segmentation performance, the dataset is randomly split into the training subset (88%), validation subset (1%) and testing subset (11%). Specifically, the training, validation, and testing subset includes 1696, 23 and 200 images, respectively. The training subset is used for the training of the network parameters for the transformer-based architectures. 23 images are utilized as a small validation subset to provide evaluations for creating the model soups while the remaining 200 images are employed to test the final segmentation performance of the architectures.

The number of parameters for transformer-based architectures is huge, which typically requires more training data to prevent underfitting. Here, the transfer learning strategy is adopted. All the three transformer-based architectures are pretrained firstly using the ImageNet [56] and ADE20K [16], which are large image datasets publicly available. However, they do not include any recycling materials related

Fig. 3. Examples of synthetic images.

Table 2
Dataset configurations.

Classes	Rebar	Bricks	PVC pipes	Plastic wires	Debris	Total
# Images	541	324	490	456	400	1919
# Pixels	1.25×10^8	9.52×10^7	1.15×10^8	1.62×10^8	1.68×10^8	6.65×10^8

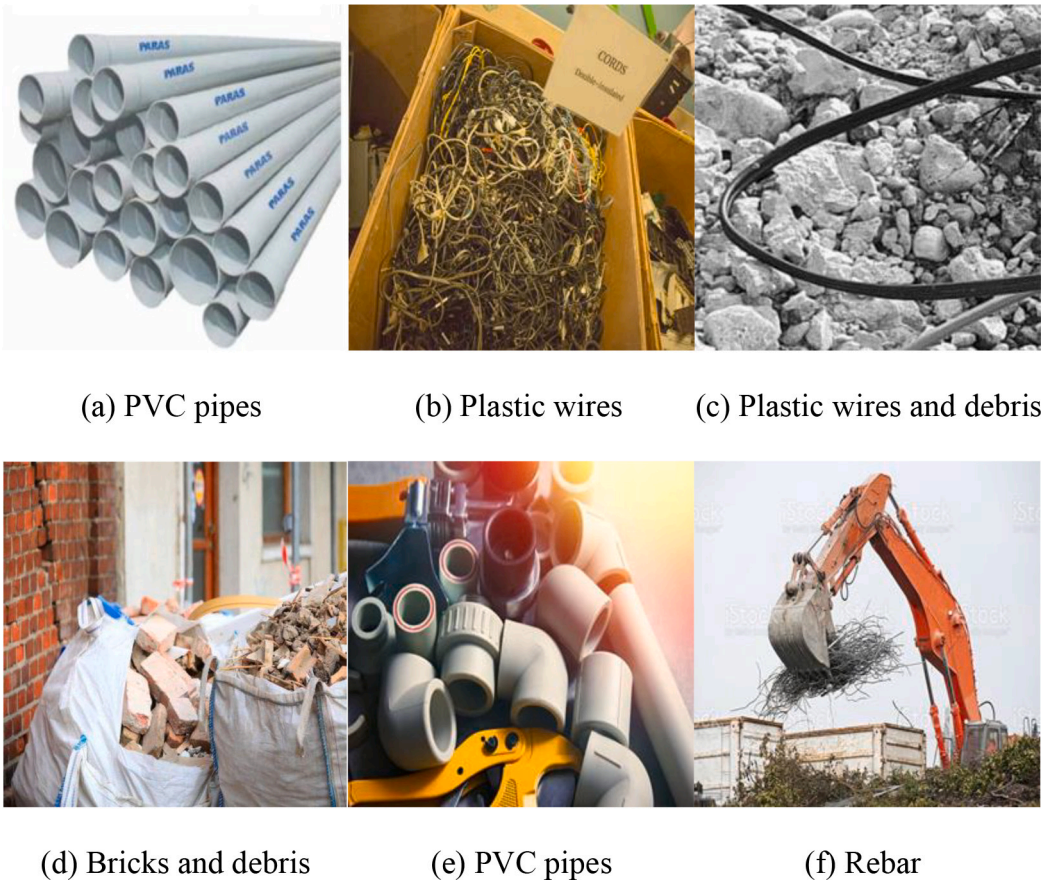


Fig. 4. Example images in the dataset.

Fig. 5. Example of manual labeling.

to construction. The dataset collected in this study is used to fine-tune all the architectures to increase the segmentation performance in construction and meanwhile shorten the training durations required.

Table 3 summarizes the parameters set for the training. The specific training process is conducted following the suggestions from the references [37, 39, 40]. The learning rate and the batch size are firstly initialized. When the training loss is steady, the learning rate is reduced

Table 3
Specific parameter settings of transformer-based architectures.

Architecture	Initial Learning rate	Weight decay	Batch size	Learning rate decay strategy
Swin Transformer	2.4×10^{-5}	1×10^{-2}	8	Poly
Twins Transformer	2.4×10^{-5}	1×10^{-2}	6	Poly
K-Net	3.0×10^{-5}	5×10^{-3}	4	Step

with different decay strategies. AdamW [47] is employed as the optimizer where the value of weight decay is shown in Table 3. The cross-entropy loss is employed as the loss function. To better train the networks, several engineering efforts are employed to increase the diversity of the data and improve the model generality. First, all training images are going through several data augmentation procedures (e.g., horizontal flipping, photometric distortion) and randomly cropped with a spatial size of 512×512 as the inputs. Second, the labeling smoothing technique [57] with the factor of 0.8 is employed to penalize over-confident predictions and improve the model generalization.

Fig. 6 shows the loss reduction along with the training progress. The training process is terminated when there is no significant improvement for the validation performance. During the training process, the

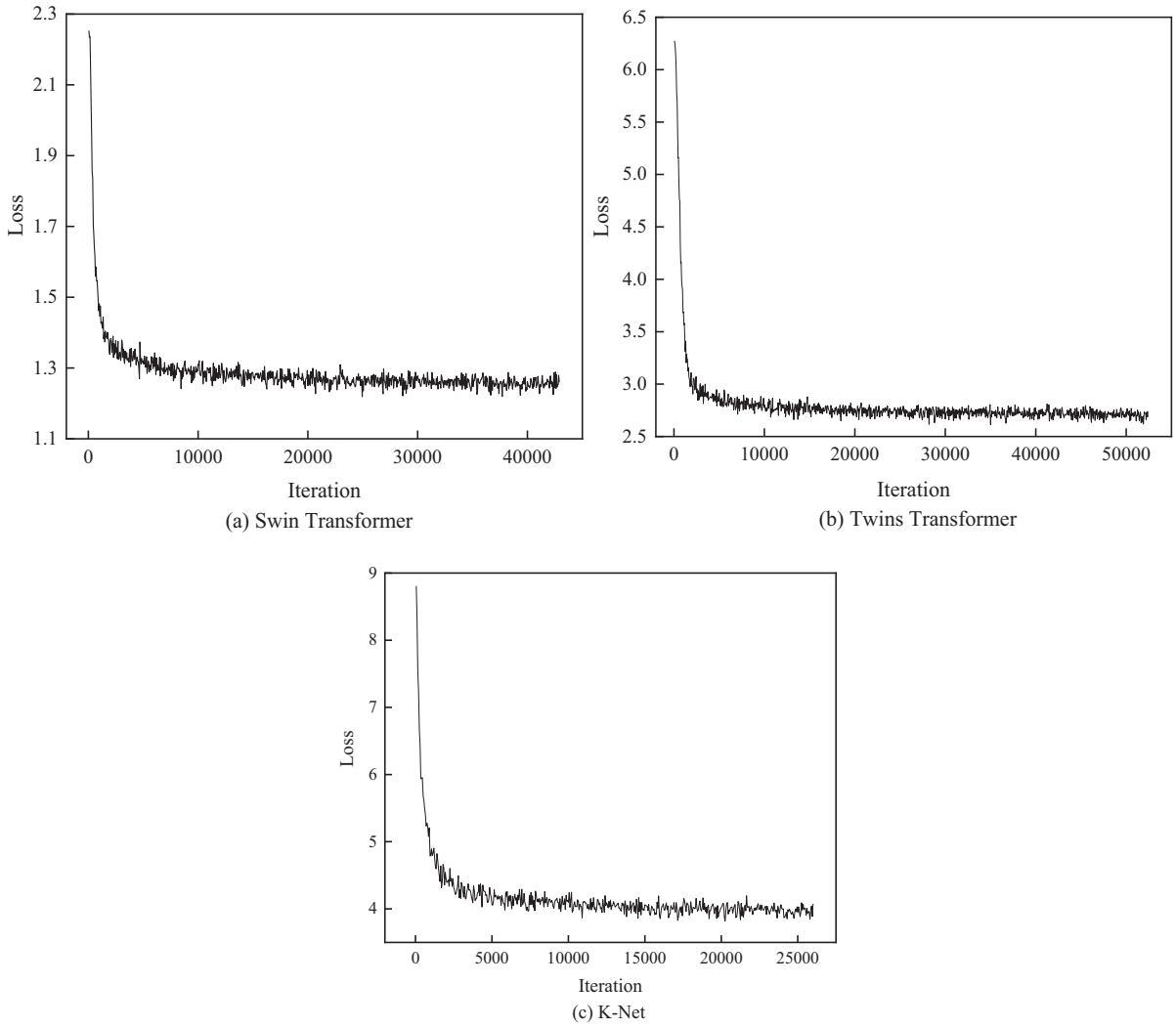


Fig. 6. Training loss for different transformer-based architectures.

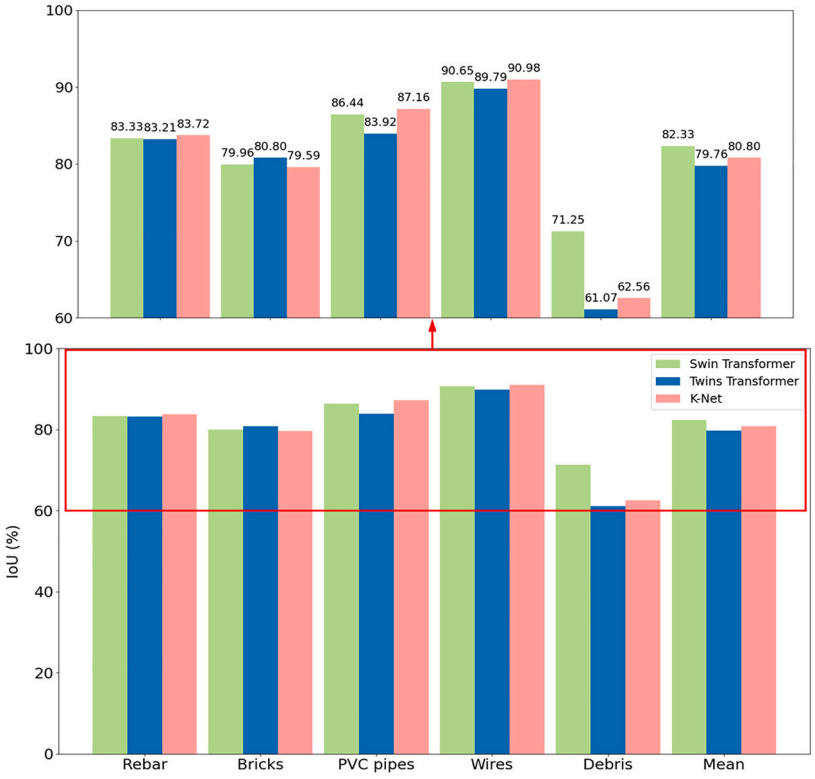
checkpoints are saved every 1000 iterations, which will be used for model averaging. Taking Swin Transformer as an example, the training loss tends to be stable when the iteration reaches about 6000. The training is completed after 42,900 iterations and a total of 42 checkpoints are saved along the process. The trained models are publicly available on https://drive.google.com/drive/folders/1IrEPBp0oaBjKq5taZ3vZcFrPVMBCyrDc?usp=share_link.

6.2.3. Experimental results

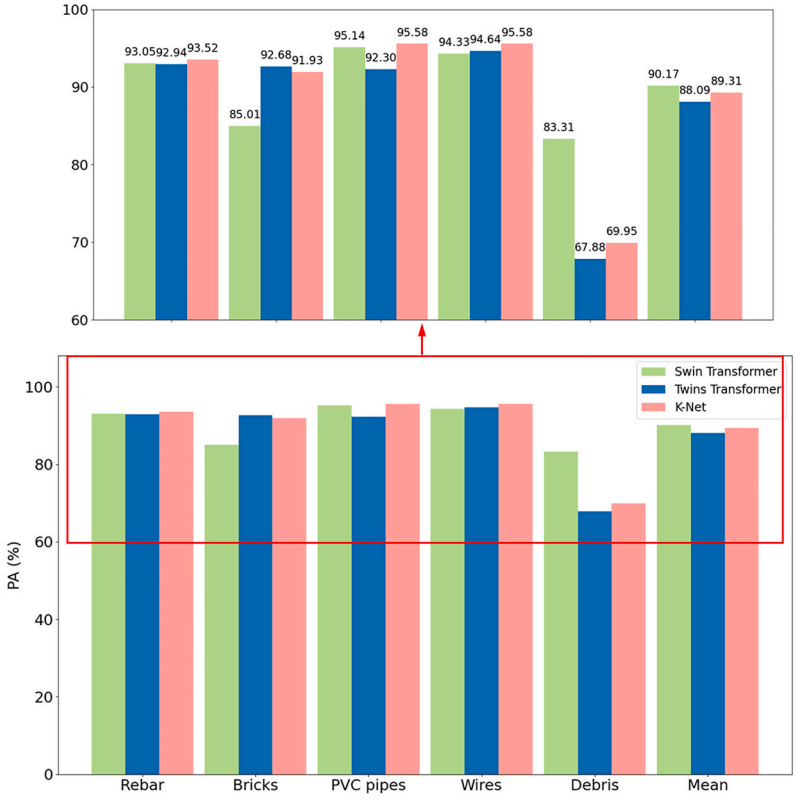
Fig. 7 shows the systemic evaluation results of the transformer-based architectures in terms of IoU and PA. The mIoU of Swin Transformer, Twins Transformer and K-Net are 82.33%, 79.76% and 80.80%, separately, while the mPA of these three architectures are 90.17%, 88.09% and 89.31%, respectively. The results indicate that Swin Transformer achieves better overall segmentation performance compared to Twins Transformer and K-Net. These three architectures obtain comparable performance on the segmentation of rebar, bricks, PVC pipes and wires. However, Swin Transformer outperforms the other two architectures by at least 8.69% for IoU and 13.36% for PA when segmenting the debris. This relatively large superiority in segmenting debris makes Swin Transformer achieve the best overall performance.

Further, the ensemble model with Twins Transformer and K-Net as base models is implemented as an example to demonstrate the effectiveness of the two-stage model ensembling framework. The numbers of checkpoints which are preserved for generating the model soups of Twins Transformer and K-Net are 2 and 4, respectively. The grid search results for ensembling Twins Transformer and K-Net are shown in Fig. 8. It indicates that the best ensemble result could be achieved with the ratio of Twins Transformer to K-Net close to 1:2 (e.g., 4 for Twins Transformer and 9 for K-Net). Fig. 9 shows the segmentation performance of the ensemble model. It is found the ensemble model achieves the mIoU and mPA of 82.36% and 90.30%, respectively. Compared to the best results achieved by the individual models, the mIoU for segmenting rebar, bricks, PVC pipes, wires and debris increases by 0.17%, 0.32%, 2.67%, 0.34% and 3.06%, separately. The mPA of these five materials is improved by 0.46%, 0.79%, 1.15%, 1.50% and 0.31%, respectively. As reported in the literature, the mIoU of semantic segmentation techniques on different construction tasks ranged from 64.67% to 86.29% [9,22,26,27]. The mIoU of this study is 82.36%, which suggests that our segmentation model achieves a satisfactory level of accuracy in segmenting the target objects.

Fig. 10 shows the segmentation results of the example images using the ensemble model. It can be seen that different kinds of recycling materials could be successfully and precisely identified and segmented by the ensemble model. Taking the top right image as an example, there are two kinds of materials including plastic wires and debris. The ensemble model could segment not only the long and narrow wires but also the voluminous debris which occupied most of the pixels in the image. This result validates the model's capability of handling multi-scale objects in the same image. Also, for the bottom right image, the



(a) IoU



(b) PA

Fig. 7. Systemic evaluation and comparison results.

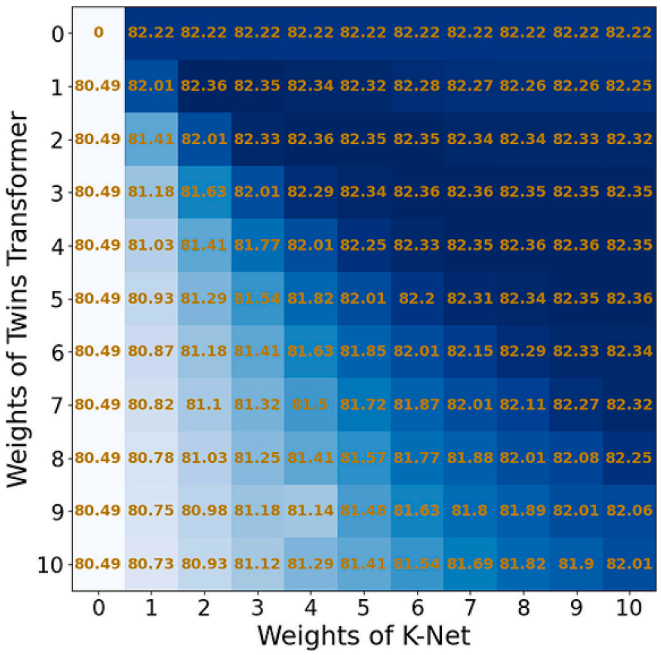


Fig. 8. Grid search results for the probability weighting process in terms of mIoU (%).

rebar could be distinguished from the grass below although they look similar to each other.

Fig. 11 presents the comparison among the best single models for Twins Transformer and K-Net, their model soups and the ensemble model. The model soup of Twins Transformer outperforms its best single model by 0.73% for mIoU and 0.45% for mPA. For K-Net, the mIoU and mPA of its model soup increases by 1.42% and 0.97%, respectively. The final ensemble model obtains an increase of 1.56% for mIoU and 0.99% for mPA, separately, compared to the best single model among these two architectures. This performance is generally considered as a major advance in the field of computer vision. A paired *t*-test is conducted for the mIoU between the ensemble model and the best single model. The results yield with a *P*-value of 0.002, which indicate that there is significant performance gain for the ensemble model. Further, the literature reported that the mIoU improvement by the ensemble learning could be in the range of [0.14%, 1.66%] [58], [0.50%, 1.20%] [59] or [0.25%, 2.07%] [60], depending on the complexity of the dataset and base models. In this study, the performance gain of the ensemble model validates the effectiveness of the two-stage model ensembling strategy for improving the segmentation performance.

7. Discussion

The segmentation results show that the segmentation performance of the transformer-based architectures varies across different classes of recycling materials. As shown in Figs. 7 and 9, the classes of bricks and debris have relatively low IoU and PA. This is partly because that the appearance and characteristics of these two classes are similar to each other, especially for broken bricks and debris. During the manual labeling process, the relatively large and bulky broken bricks are annotated as bricks while the terribly fragmented bricks are annotated as debris. For example, in Fig. 12 (a), the left part of broken bricks belongs to bricks while the right part should be debris. However, this rule is difficult for the networks to learn precisely. Taking Fig. 12 (b) as an example, the terribly fragmented bricks in this image should belong to the class of debris based on our labeling rule while the networks falsely segment them as bricks. To solve this problem, one possible way could be considering a cost sensitive loss function to make the networks pay more attention to the segmentation performance of bricks and debris.

The ensemble model could enhance the segmentation performance and reduce the variance of individual models. The individual models may make some mistakes. Taking Fig. 13 as an example, K-Net does not segment the bricks well in the circled areas. Compared to K-Net, the ensemble model performs better in those areas, which means it could overcome the weaknesses of individual models in a way. Further, the ensemble model can leverage the strengths of different models to improve the overall performance. As shown in Fig. 7, when comparing Twins Transformer and K-Net, Twins Transformer achieves better results on bricks, while K-Net performs better for segmenting the other four types of materials. By combining these two models that perform well on different types of materials, the IoU and PA of the ensemble model are

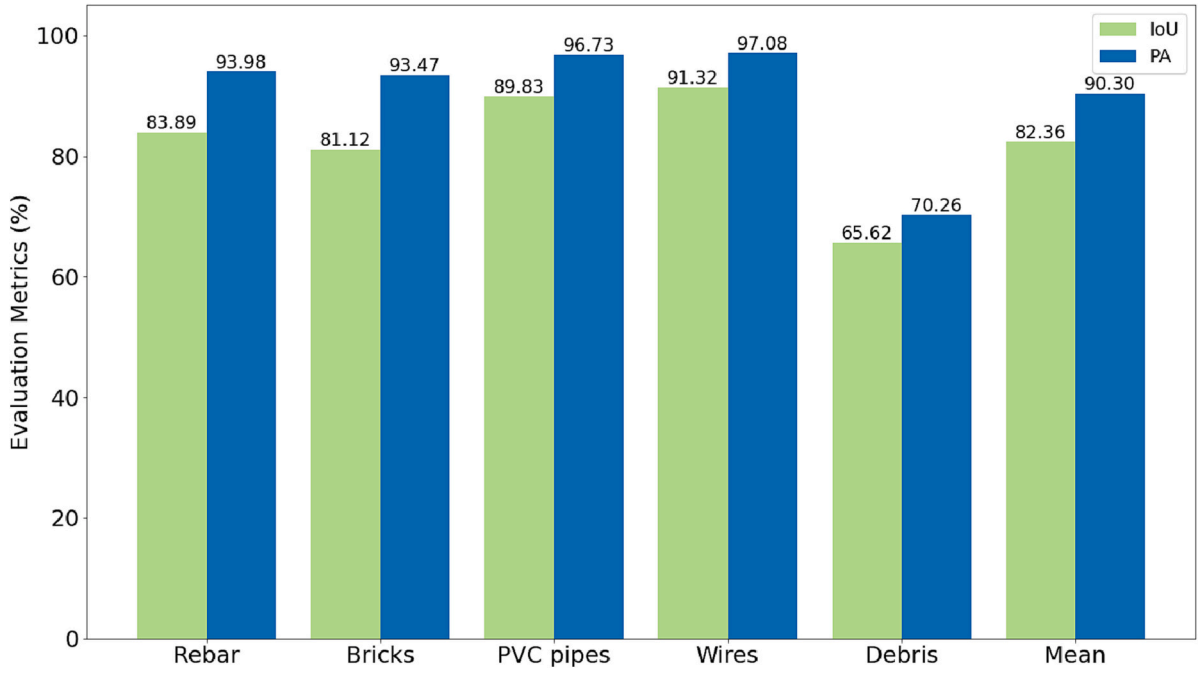


Fig. 9. Quantitative results of the ensemble model.

Fig. 10. Test results of the example images.

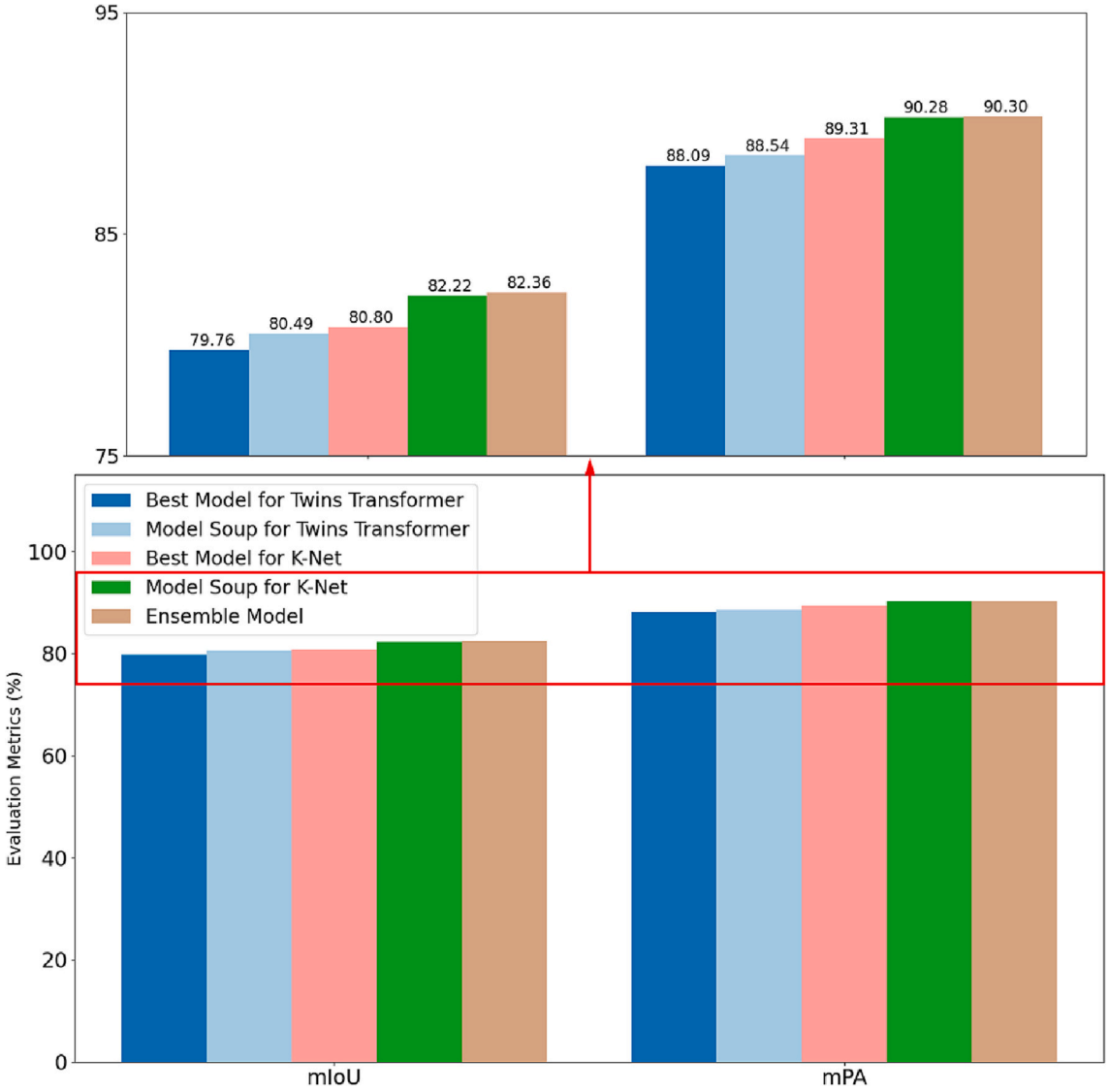


Fig. 11. Comparison among best single models, model soups and final ensemble model.

(a) A labeling example (b) A false prediction example

Fig. 12. Difference between broken bricks and debris.

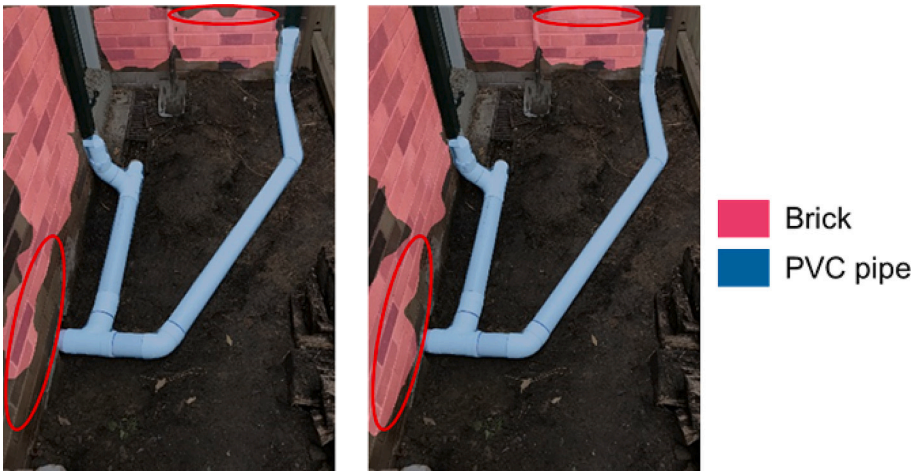


Fig. 13. Comparison example between K-Net (left) and the ensemble model (right).

higher than both Twins Transformer and K-Net for all these five materials (Fig. 9), suggesting that the ensemble model can achieve higher accuracy and better generalize to new, unseen data.

Besides, the performance gain obtained by the two-stage model ensembling strategy may be related to the best segmentation performance of base models. Table 4 summarizes the performance of the ensemble models with different base models. The performance gain is 1.56% when the mIoU of the best single model is 80.80%. When the mIoU of the best single model achieves 82.33%, the performance gain for the ensemble model would drop to 0.41%. It could be seen that the performance gain of the ensemble model would be lowered with the increase of the best single model performance. This phenomenon was also found in other research work [61]. It might indicate that there is generally less improvement room for the single model with a higher performance.

Table 4
Comparison of different ensemble models in terms of mIoU (%).

Base models	Twins Transformer + K-Net	Swin Transformer + Twins Transformer + K-Net
Best single model	80.80	82.33
Ensemble model	82.36	82.74
Performance gain	1.56	0.41

Further, the performance trend of different transformer-based architectures on general image segmentation dataset (e.g., ADE20K) and construction image dataset (ours) is compared. On ADE20K dataset, K-Net outperforms Swin Transformer and Twins Transformer. In contrast, Swin Transformer achieves the best overall segmentation performance on our created construction dataset. The next two are K-Net and Twins Transformer, separately. The result indicates that the transformer-based architectures may have different performance rankings in the construction domain. This practice suggests the necessity of our created dataset, which provides a new benchmark to evaluate the state-of-art transformer-based architectures in the construction field.

Our segmentation models have the potential for developing an automatic visual surveillance mobile robotic platform on construction sites. For example, the models could be integrated with the mobile robotic systems (e.g., drones, unmanned vehicles) to automatically identify and segment the recycling materials on sites. The mobile robotic systems could reach out to onsite locations safely and quickly, especially for those hard-to-reach or unsafe ones. In this case, our segmentation models could identify whether there exists construction waste that need to be recycled and roughly the amount of recycling materials at these locations. These results would provide valuable information for the subsequent management of the construction waste, such as determining the recycling schedule and cost.

8. Conclusions and future work

Construction sites are increasingly equipped with cameras to acquire imagery data for promoting automation, solving safety issues, etc. Automated segmentation techniques are critical for processing the acquired images to provide the visual understanding of the construction environment. Compared with other semantic segmentation techniques, transformer-based architectures have gradually become mainstream in segmentation tasks due to their ability to understand long-range dependencies. This paper presents a systematic evaluation of three state-of-the-art transformer-based architectures on construction image segmentation tasks. Further, their ensemble model based on model averaging and probability weighting is implemented for performance improvement. A dataset containing construction recycling materials is created and employed as a benchmark to compare their performance. The results indicate that the ensemble model could achieve a mIoU of 82.36% and mPA of 90.30%, which demonstrate superior segmentation performance on construction images.

Although the learned models showed promising results on our dataset, there are still several aspects that need to be further considered. First, the transformer-based models have only been evaluated on construction images containing the recycling materials in this study. Considering that construction sites are complex and cluttered with tools, materials, workers, etc., more classes of construction objects could be included into our dataset to make our models more robust to accommodate such complicated characteristics of the environment. Second, there is a tradeoff between the segmentation accuracy and efficiency for the probability weighting process. Although the probability weighting technique increases the segmentation accuracy by combing the predictions of different models, it would inevitably decrease the segmentation efficiency since all models need to be executed to output the final results. This tradeoff should be carefully considered by the user requirement, especially for the real-time applications.

Future work will focus on two aspects. First, more construction objects will be included into our dataset to make the training and testing of transformer-based networks more robust. Second, a cost sensitive loss function will be considered to improve the segmentation performance of the specific classes.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgement

This paper is based in part upon the work supported by the Wisconsin Alumni Research Foundation (WARF) under Project No. AAJ4872 and the M.A. Mortenson Company. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of WARF or Mortenson. The authors would like to express our thanks to Visualization, Information Modeling, and Simulation (VIMS) of American Society of Civil Engineers (ASCE) and International Association for Automation and Robotics in Construction (IAARC) for organizing the VIMS-IAARC Joint Datathon 2022 Competition and providing a part of the dataset.

References

[1] B. Zhang, Z. Zhu, A. Hammad, W. Aly, Automatic matching of construction onsite resources under camera views, Autom. Constr. 91 (2018) 206–215, https://doi.org/10.1016/J.AUTCON.2018.03.011.
[2] X. Wang, Z. Zhu, Vision-based hand signal recognition in construction: a feasibility study, Autom. Constr. 125 (2021), 103625, https://doi.org/10.1016/j.autcon.2021.103625.
[3] X. Yan, H. Zhang, H. Li, Estimating worker-centric 3D spatial crowdedness for construction safety management using a single 2D camera, J. Comput. Civ. Eng. 33 (2019) 04019030, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000844.
[4] A. Kazemian, X. Yuan, O. Davtalab, B. Khoshnevis, Computer vision for real-time extrusion quality monitoring and control in robotic construction, Autom. Constr. 101 (2019) 92–98, https://doi.org/10.1016/J.AUTCON.2019.01.022.
[5] B. Ekanayake, J.K.W. Wong, A.A.F. Fini, P. Smith, Computer vision-based interior construction progress monitoring: a literature review and future research directions, Autom. Constr. 127 (2021), 103705, https://doi.org/10.1016/J.AUTCON.2021.103705.
[6] B. Zhong, H. Li, H. Luo, J. Zhou, W. Fang, X. Xing, Ontology-based semantic modeling of knowledge in construction: classification and identification of hazards implied in images, J. Constr. Eng. Manag. 146 (2020) 04020013, https://doi.org/10.1061/(ASCE)CO.1943-7862.0001767.
[7] C. Chen, Z. Zhu, A. Hammad, Automated excavators activity recognition and productivity analysis from construction site surveillance videos, Autom. Constr. 110 (2020), 103045, https://doi.org/10.1016/j.autcon.2019.103045.
[8] K. Asadi, H. Ramshankar, H. Pullagurla, A. Bhandare, S. Shanbhag, P. Mehta, S. Kundu, K. Han, E. Lobaton, T. Wu, Vision-based integrated mobile robotic system for real-time applications in construction, Autom. Constr. 96 (2018) 470–482, https://doi.org/10.1016/J.AUTCON.2018.10.009.
[9] X. Xie, J. Cai, H. Wang, Q. Wang, J. Xu, Y. Zhou, B. Zhou, Sparse-sensing and superpixel-based segmentation model for concrete cracks, Comput. Civ. Infrastruct. Eng. 37 (2022) 1769–1784, https://doi.org/10.1111/MICE.12903.
[10] Y. Pi, N.D. Nath, A.H. Behzadan, Detection and semantic segmentation of disaster damage in UAV footage, J. Comput. Civ. Eng. 35 (2020) 04020063, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000947.
[11] X. Wang, S. Wang, Y. Zhu, X. Meng, Image segmentation based on support vector machine, in: Proc. 2nd Int. Conf. Comput. Sci. Netw. Technol, IEEE, 2012, pp. 202–206, https://doi.org/10.1109/ICCSNT.2012.6525921.
[12] B. Kang, T.Q. Nguyen, Random Forest with learned representations for semantic segmentation, IEEE Trans. Image Process. 28 (2019) 3542–3555, https://doi.org/10.1109/TIP.2019.2905081.
[13] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit, 2015, pp. 3431–3440, https://doi.org/10.1109/CVPR.2015.7298965.
[14] V. Badrinarayanan, A. Kendall, R. Cipolla, SegNet: a deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2017) 2481–2495, https://doi.org/10.1109/TPAMI.2016.2644615.
[15] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, C.O.C.O. Microsoft, Common objects in context, in: Eur. Conf. Comput. Vis, Springer, 2014, pp. 740–755, https://doi.org/10.1007/978-3-319-10602-1_48.
[16] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ADE20K dataset, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit, 2017, pp. 633–641, https://doi.org/10.1109/CVPR.2017.544.
[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Adv. Neural Inf. Process. Syst, 2017, pp. 5998–6008, https://doi.org/10.48550/arXiv.1706.03762.
[18] J.M.J. Valanarasu, P. Oza, I. Hacihaliloglu, V.M. Patel, Medical transformer: gated axial-attention for medical image segmentation, in: Med. Image Comput. Comput. Assist. Interv, Springer, 2021, pp. 36–46, https://doi.org/10.1007/978-3-030-87193-2_4.
[19] B. Zhou, P. Krähenbühl, Cross-view transformers for real-time map-view semantic segmentation, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit, 2022, pp. 13760–13769, https://doi.org/10.48550/arXiv.2205.02833.
[20] W. Der Yu, H.C. Liao, W.T. Hsiao, H.K. Chang, C.K. Tsai, C.C. Lin, Automatic safety monitoring of construction Hazard working zone: A semantic segmentation based deep learning approach, in: Proc. Int. Conf. Autom. Logist, ACM, 2020, pp. 54–59, https://doi.org/10.1145/3412953.3412969.
[21] C. Wang, G. Chen, M. Huang, J. Lin, Rust defect detection and segmentation method for tower crane, in: Cross Strait Radio Sci. Wirel. Technol. Conf, IEEE, 2020, pp. 1–3, https://doi.org/10.1109/CSRSWTC50769.2020.9372457.
[22] S. Bang, Y. Hong, H. Kim, Proactive proximity monitoring with instance segmentation and unmanned aerial vehicle-acquired video-frame prediction, Comput. Civ. Infrastruct. Eng. 36 (2021) 800–816, https://doi.org/10.1111/mice.12672.
[23] Y.C. Zhou, Z.Z. Hu, K.X. Yan, J.R. Lin, Deep learning-based instance segmentation for indoor fire load recognition, IEEE Access. 9 (2021) 148771–148782, https://doi.org/10.1109/ACCESS.2021.3124831.
[24] K. Asadi, P. Chen, K. Han, T. Wu, E. Lobaton, Real-time scene segmentation using a light deep neural network architecture for autonomous robot navigation on construction sites, in: Comput. Civ. Eng. 2019 Data, Sensing, Anal, ASCE, 2019, pp. 320–327, https://doi.org/10.1061/9780784482438.041.
[25] G.A. Atkinson, W. Zhang, M.F. Hansen, M.L. Holloway, A.A. Napier, Image segmentation of underfloor scenes using a mask regions convolutional neural network with two-stage transfer learning, Autom. Constr. 113 (2020), 103118, https://doi.org/10.1016/j.autcon.2020.103118.

[26] Z. Wang, Y. Zhang, K.M. Mosalam, Y. Gao, S.L. Huang, Deep semantic segmentation for visual understanding on construction sites, Comput. Civ. Infrastruct. Eng. 37 (2022) 145–162, https://doi.org/10.1111/mice.12701.
[27] W. Wang, C. Su, Deep learning-based real-time crack segmentation for pavement images, KSCE J. Civ. Eng. 25 (2021) 4495–4506, https://doi.org/10.1007/s12205-021-0474-2.
[28] C. Wang, S.E. Antos, L.M. Triveno, Automatic detection of unreinforced masonry buildings from street view images using deep learning-based image segmentation, Autom. Constr. 132 (2021), 103968, https://doi.org/10.1016/j.autcon.2021.103968.
[29] N. Dhanachandra, K. Manglem, Y.J. Chanu, Image segmentation using K-means clustering algorithm and subtractive clustering algorithm, in: Procedia Comput. Sci, Elsevier, 2015, pp. 764–771, https://doi.org/10.1016/j.procs.2015.06.090.
[30] F. Visin, A. Romero, K. Cho, M. Matteucci, M. Ciccone, K. Kastner, Y. Bengio, A. Courville, ReSeg: a recurrent neural network-based model for semantic segmentation, in: IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work, 2016, pp. 41–48, https://doi.org/10.1109/CVPRW.2016.60.
[31] N. Souly, C. Spampinato, M. Shah, Semi supervised semantic segmentation using generative adversarial network, in: Proc. IEEE Int. Conf. Comput. Vis, 2017, pp. 5688–5696, https://doi.org/10.1109/ICCV.2017.606.
[32] Z. Ren, Z. Yu, X. Yang, M.Y. Liu, A.G. Schwing, J. Kautz, UFO2 : a unified framework towards omni-supervised object detection, in: Eur. Conf. Comput. Vis, Springer, 2020, pp. 288–313, https://doi.org/10.1007/978-3-030-58529-7_18.
[33] S.Y. Pan, C.Y. Lu, S.P. Lee, W.H. Peng, Weakly-supervised image semantic segmentation using graph convolutional networks, in: IEEE Int. Conf. Multimed. Expo, IEEE, 2021, pp. 1–6, https://doi.org/10.1109/ICME51207.2021.9428116.
[34] Y. Liu, Y.H. Wu, P. Wen, Y. Shi, Y. Qiu, M.M. Cheng, Leveraging instance-, image-and dataset-level information for weakly supervised instance segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 44 (2022) 1415–1428, https://doi.org/10.1109/TPAMI.2020.3023152.
[35] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin, Emerging properties in self-supervised vision transformers, in: Proc. IEEE/CVF Int. Conf. Comput. Vis, 2021, pp. 9650–9660, https://doi.org/10.48550/arXiv.2104.14294.
[36] R. Ranftl, A. Bochkovskiy, V. Koltun, Vision transformers for dense prediction, in: Proc. IEEE/CVF Int. Conf. Comput. Vis, 2021, pp. 12179–12188, https://doi.org/10.48550/arXiv.2103.13413.
[37] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proc. IEEE/CVF Int. Conf. Comput. Vis, 2021, pp. 10012–10022, https://doi.org/10.48550/arXiv.2103.14030.
[38] X. Dong, J. Bao, D. Chen, W. Zhang, N. Yu, L. Yuan, D. Chen, B. Guo, CSWin transformer: a general vision transformer backbone with cross-shaped windows, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit, 2022, pp. 12124–12134, https://doi.org/10.48550/arxiv.2107.00652.
[39] X. Chu, Z. Tian, Y. Wang, B. Zhang, H. Ren, X. Wei, H. Xia, C. Shen, Twins: revisiting the design of spatial attention in vision transformers, in: Adv. Neural Inf. Process. Syst, 2021, pp. 9355–9366, https://doi.org/10.48550/arXiv.2104.13840.
[40] W. Zhang, J. Pang, K. Chen, C.C. Loy, K-Net: towards unified image segmentation, in: Adv. Neural Inf. Process. Syst, 2021, pp. 10326–10338, https://doi.org/10.48550/arxiv.2106.14855.
[41] B. Cheng, I. Misra, A.G. Schwing, A. Kirillov, R. Girdhar, Masked-attention mask transformer for universal image segmentation, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit, 2022, pp. 1290–1299, https://doi.org/10.48550/arxiv.2112.01527.
[42] L. Wang, R. Li, C. Zhang, S. Fang, C. Duan, X. Meng, P.M. Atkinson, UNetFormer: a UNet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery, ISPRS J. Photogramm. Remote Sens. 190 (2022) 196–214, https://doi.org/10.1016/J.ISPRSJPRS.2022.06.008.
[43] S. Khan, M. Naseer, M. Hayat, S.W. Zamir, F.S. Khan, M. Shah, Transformers in vision: a survey, ACM Comput. Surv. 54 (2022) 1–41, https://doi.org/10.1145/3505244.
[44] A.L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. van Ginneken, A. Kopp-Schneider, B.A. Landman, G. Litjens, B. Menze, O. Ronneberger, R. M. Summers, P. Bilic, P.F. Christ, R.K.G. Do, M. Gollub, J. Golia-Pernicka, S. H. Heckers, W.R. Jarnagin, M.K. McHugo, S. Napel, E. Vorontsov, L. Maier-Hein, M.J. Cardoso, A large annotated medical image dataset for the development and evaluation of segmentation algorithms, ArXiv Prepr. ArXiv. 12 (2019), https://doi.org/10.48550/arxiv.1902.09063, 1902.09063.
[45] O. Sagi, L. Rokach, Ensemble learning: a survey, Wiley Interdiscip, Rev. Data Min. Knowl. Discov. 8 (2018), e1249, https://doi.org/10.1002/WIDM.1249.
[46] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, J. Sun, Unified perceptual parsing for scene understanding, in: Proc. Eur. Conf. Comput. Vis, 2018, pp. 418–434, https://doi.org/10.48550/arXiv.1807.10221.
[47] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, ArXiv Prepr, ArXiv. (2017) 1711.05101, https://doi.org/10.48550/arXiv.1711.05101.
[48] M. Wortsman, G. Ilharco, S.Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A.S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith, L. Schmidt, Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time, in: Int. Conf. Mach. Learn, PMLR, 2022, pp. 23965–23998, https://doi.org/10.48550/arXiv.2203.05482.
[49] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, A.G. Wilson, Averaging weights leads to wider optima and better generalization, ArXiv Prepr. ArXiv. (2018) 1803.05407, https://doi.org/10.48550/arXiv.1803.05407.
[50] D. Chicco, Ten quick tips for machine learning in computational biology, BioData Min. 10 (2017) 10–35, https://doi.org/10.1186/s13040-017-0155-3.
[51] J. Chen, W. Lu, F. Xue, "Looking beneath the surface": a visual-physical feature hybrid approach for unattended gauging of construction waste composition, J. Environ. Manag. 286 (2021), 112233, https://doi.org/10.1016/j.jenvman.2021.112233.
[52] W. Lu, J. Chen, F. Xue, Using computer vision to recognize composition of construction waste mixtures: a semantic segmentation approach, Resour. Conserv. Recycl. 178 (2022), 106022, https://doi.org/10.1016/j.resconrec.2021.106022.
[53] International Association for Automation and Robotics in Construction, in: The 3rd Annual VIMS / 1st VIMS-IAARC Joint Datathon Competition, Linkedin, 2022. https://www.linkedin.com/feed/update/urn:li:activity:6918671751778373632/ (accessed June 15, 2022).
[54] A. Dutta, A. Zisserman, The VIA annotation software for images, audio and video, in: Proc. 27th ACM Int. Conf. Multimed, Association for Computing Machinery, Inc, 2019, pp. 2276–2279, https://doi.org/10.1145/3343031.3350535.
[55] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, in: Adv. Neural Inf. Process. Syst, 2019, pp. 8026–8037, https://doi.org/10.48550/arXiv.1912.01703.
[56] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, Li Fei-Fei, ImageNet: A large-scale hierarchical image database, in: IEEE Conf. Comput. Vis. Pattern Recognit, 2009, pp. 248–255, https://doi.org/10.1109/cvpr.2009.5206848.
[57] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the Inception Architecture for Computer Vision, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit, 2016, pp. 2818–2826, https://doi.org/10.1109/CVPR.2016.308.
[58] Y.W. Kim, Y.C. Byun, A.V.N. Krishna, Portrait segmentation using ensemble of heterogeneous deep-learning models, Entropy. 23 (2021) 197, https://doi.org/10.3390/e23020197.
[59] R. Ali, R.C. Hardie, H.K. Ragb, Ensemble lung segmentation system using deep neural networks, in: IEEE Appl. Imag. Pattern Recognit. Work, 2020, pp. 1–5, https://doi.org/10.1109/AIPR50011.2020.9425311.
[60] T. Dang, A.V. Luong, A.W.C. Liew, J. McCall, T.T. Nguyen, Ensemble of deep learning models with surrogate-based optimization for medical image segmentation, in: IEEE Congr. Evol. Comput, IEEE, 2022, pp. 1–8, https://doi.org/10.1109/CEC55065.2022.9870389.
[61] A. Renda, M. Barsacchi, A. Bechini, F. Marcelloni, Comparing ensemble strategies for deep learning: an application to facial expression recognition, Expert Syst. Appl. 136 (2019) 1–11, https://doi.org/10.1016/j.eswa.2019.06.025.