ISLE: A Framework for Image Level Semantic Segmentation Ensemble

First Author[1][0000−1111−2222−3333], Second Author[2,3][1111−2222−3333−4444], and Third Author[3][2222−−3333−4444−5555]

No Institute Given

Abstract. One key bottleneck of employing state-of-the-art semantic segmentation networks in the real world is the availability of training labels. Conventional semantic segmentation networks require massive pixel-wise annotated labels to reach state-of-the-art prediction quality. Hence, several works focus on semantic segmentation networks trained with only image-level annotations. However, when scrutinizing the results of state-of-the-art in more detail, we notice that they are remarkably close to each other on average prediction quality, different approaches perform better in different classes while providing low quality in others. To address this problem, we propose a novel framework, ISLE, which employs an ensemble of the "pseudo-labels" for a given set of different semantic segmentation techniques on a class-wise level. Pseudo-labels are the pixel-wise predictions of the image-level semantic segmentation frameworks used to train the final segmentation model. Our pseudo-labels seamlessly combine the strong points of multiple segmentation techniques approaches to reach superior prediction quality. We reach up to 2.4% improvement over ISLE's individual components. An exhaustive analysis was performed to demonstrate ISLE's effectiveness over state-of-the-art frameworks for image-level semantic segmentation.

Keywords: Semantic Segmentation, Weakly Supervised, Ensemble, Deep Learning, Class Activation Maps

1 Introduction

Generating high-quality semantic segmentation predictions using only models trained on image-level annotations would enable a new level of applicability. The progress of fully supervised semantic segmentation networks has already helped provide many useful tools and applications. For example, in autonomous and self-driving vehicles [1,2], remote sensing [3,4], facial recognition [5,6], agriculture [7,8], and in the medical field [9,10], etc. The downside of those fully supervised semantic segmentation networks (FSSS) is that they require copious amounts of pixel-wise annotated images. Generating such a training set is very tedious and time-consuming work. For instance, one image of the Cityscapes dataset, which contains street scenes from cities that require many complex objects to be annotated, takes more than an hour of manual user-driven labour [11].

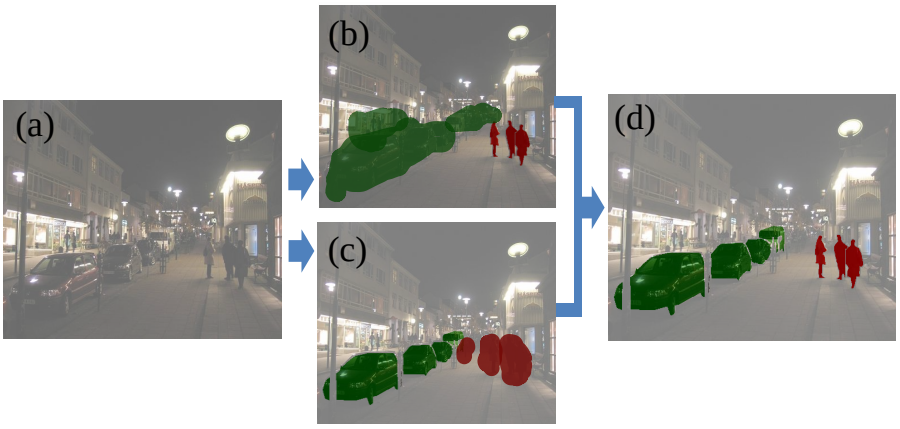

Fig. 1. The idea of our Framework: (a) Input image, (b) prediction of a method that is good with person segmentation and bad with cars, (c) prediction of a method good with car segmentation and bad with persons, (d) prediction of ENLIST, combining the strengths of the two methods.

Furthermore, medical imaging and molecular biology fields require the knowledge of highly qualified individuals capable of interpreting and annotating the images.

Therefore, to reduce the time and resources required for generating pixel-wise masks, a wide range of research works focus on developing approaches that focus on weaker kinds of supervision. In this work, we will focus on weak supervision in the form of image-level labels. Image-level labels give the least amount of supervision for semantic segmentation but are the easiest to acquire.

Several works already focus on image-level semantic segmentation techniques, and they consistently reach better and better high scores. Most works are based on Class Activation Maps (CAMs) [12]. CAMs localize the object by training a DNN model with classification loss and then reusing the learned weights to highlight the image areas responsible for its classification decision. Most image-level segmentation approaches aim to improve the CAM baseline by adding additional regularizations to the classification loss or refining the CAM mask afterward. As more methods emerge for improving CAM quality, state-of-the-art is usually compiled of regularizations, after-the-fact refinements, or combinations of both. However, when analyzing different image-level segmentation techniques on a class-by-class basis, we observed that the differences between those them vary significantly on specific classes, although those methods generate predictions that reach comparable scores on average.

Therefore, we are proposing our ISLE framework. In our framework, we combine the pseudo-labels of multiple image-level segmentation techniques based on the respective class scores to generate a superset of pseudo-labels, combining the upsides of multiple different approaches. Fig. 1 visualizes the gains possible of ISLE in comparison to its components. The detection of edges of objects

is a major weakness in Image-level based semantic segmentation and many state-of-the-art methods tried to address this problem. We noticed that they achieve their goal to some degree. Namely, in certain objects, a specific method achieves remarkable results, but also scores under average with different objects. Hence, our ISLE framework addresses the problem of insufficient object border detection, by applying state-of-the-art only in scenarios, where their individual approach is suited best. We perform extensive experiments on the PASCAL VOC2012 dataset [13] to prove the effectiveness of the proposed framework in various experimental settings and compare them with a range of state-of-the-art techniques to illustrate the benefits of our approach. The **key contribution** of this work are:

1. Our novel ISLE framework improves the prediction quality of segmentation masks by combining state-of-the-art pseudo-labels on a class-by-class basis.
2. Our ISLE framework is not limited by the number or approach of any image-level guided segmentation frameworks to combine their pseudo-labels. Since the ISLE is only used for generating pseudo-labels, it will not add more computations for inference predictions.
3. We present detailed ablation studies and analysis comparing the results of ISLE to state-of-the-art methods on the VOC2012 dataset to evaluate our method's efficacy and the improvements achieved using our framework.
4. The complete framework will be made open-source and accessible online once published at `https://anonymous.4open.science/r/ISLE-1C41/README.md`.

2 Related Work

In this section, we provide a discussion of the current state-of-the-art in semantic segmentation using image-level supervision.

AffinityNet [14] trains a second network to learn pixel similarities, which generates a transition matrix combined with the CAM iteratively to refine its activation coverage. PuzzleCAM [15] introduces a loss, that subdivides the input image into multiple parts, forcing the network to predict image segments that contain the non-discriminative parts of an object. CLIMS [16] trained the network by matching text labels to the correct image. Hence, the network maximizes and minimizes the distance between correct and wrong pairs, respectively, instead of just giving a binary classification result. PMM [17] used Coefficient of Variation Smoothing to smooth the CAMs, Proportional Pseudo-mask Generation that introduces a new metric, which highlights the importance of each class on each location, in contrast to the scores trained from the binary classifier. Furthermore, they employed Pretended Under-fitting, which improves training with noisy labels, and Cyclic Pseudo-mask, which iteratively trains the final segmentation network with its predictions. DRS [18] aims to improve the image's activation area to less discriminative areas. [18] et al. achieve this by suppressing the attention on discriminative regions, thus guiding the attention to adjacent regions to generate a complete attention map of the target object.

Table 1 lists all the twenty classes of the VOC2012 dataset and shows, which state-of-the-art method achieves the best result on specific classes. We observe that PMM has the best result in only two of the twenty classes, DRS in three, CLIMS has the best result in seven, and PuzzleCAM in eight classes.

Table 1. Highest score per VOC2012 class on each component of ISLE.

Class	PMM	DRS	CLIMS	Puzzle
Bus	✓	✗	✗	✗
Car	✓	✗	✗	✗
Bottle	✗	✓	✗	✗
Chair	✗	✓	✗	✗
Train	✗	✓	✗	✗
Bike	✗	✗	✓	✗
Boat	✗	✗	✓	✗
Table	✗	✗	✓	✗
Motor	✗	✗	✓	✗
Person	✗	✗	✓	✗
Sofa	✗	✗	✓	✗
TV	✗	✗	✓	✗
Aero	✗	✗	✗	✓
Bird	✗	✗	✗	✓
Cat	✗	✗	✗	✓
Cow	✗	✗	✗	✓
Dog	✗	✗	✗	✓
Horse	✗	✗	✗	✓
Plant	✗	✗	✗	✓
Sheep	✗	✗	✗	✓

3 Our Framework

Our framework aims to combine the strengths of different methods by just using their predictions for classes where they are performing the best, while not considering predictions of classes where they perform worse compared to other methods. The prerequisite of our framework is a list of candidate state-of-the-art approaches for image-level semantic segmentation. Fig. 2 presents an overview of the ISLE framework. We start with collecting the pseudo labels of our candidate methods. In the next step, we can employ several refinement methods to improve the pseudo-label quality beforehand. In our case, we used AffinityNet [14] and a dense Conditional Random Field (dCRF) [19] for the candidates if the provided pseudo labels did not already undergo refinement methods. Then we can combine the pseudo-labels on a class-wise basis, where we only copy the predictions of classes of the candidate labels to our ensemble if the candidate has a high score in that class. Finally, we use the generated pseudo labels to train an FSSS network. Our proposed version uses the four state-of-the-art methods introduced in the

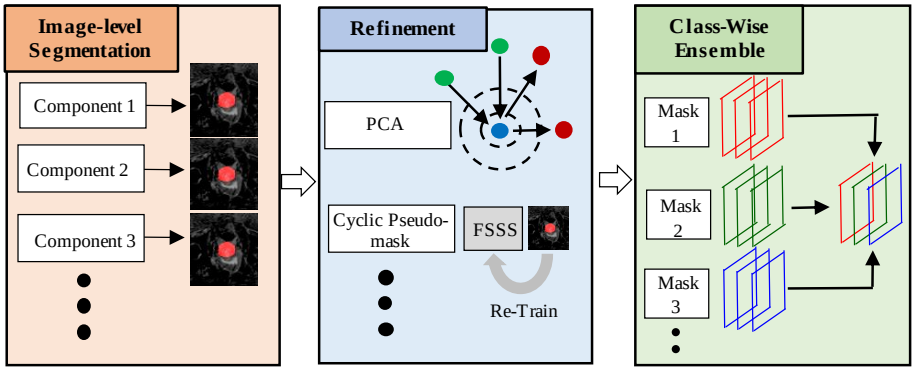

Fig. 2. Overview of the ISLE framework. The first stage is the collection of Image-level semantic segmentation; In the second stage, we can use a number of refinement methods to improve the mask quality; In the third stage, we combine the refinement masks on a class-wise basis to generate the pseudo-labels that reach the best prediction for each class; In the final stage, we are training an FSSS with the pseudo labels.

previous section, and for all of them except CLIMS, we also refine their baseline with AffinityNet. AffinityNet uses a random walk, in combination with pixel affinities. Furthermore, as a widespread practice, we use dCRF to the AffinityNet predictions to improve their quality. The refinement of the pseudo labels is not limited to AffinityNet, any combination of additional refinement methods can be employed within our framework.

In the next step, we will evaluate the different candidate pseudo label sets on a class-wise basis and determine which candidate is used for which class for the ensemble. We reshape the pseudo labels ps to the dimensions $ps_i \in \mathbb{R}^{H \times W \times C}$, if they were not already provided in that shape, where i is the specific image, H is the image height, W is the image width and C is the number of classes. We perform the combination by copying for all images the slides $ps^x_{i,c} \in \mathbb{R}^{H \times W \times 1}$ of a specific class c by the candidate network x, if x has the high score on c. We do this for every class c in C and then simply concatenate all slides to get a complete prediction, which contains only the best predictions state-of-the-art can muster on a class-wise level:

$$ps^{x_1}_{i,c_1} \times ps^{x_2}_{i,c_2} \times \ldots \times ps^{x_N}_{i,c_N} = ps^*_i,$$

Where N is the number of classes in the used dataset and ps^*_i is the ensemble prediction of image i.

Algorithm 1 ISLE 1.Step (Optional)

Input: N training-pseudo-segmentations of components: ps
Output: N refined training-pseudo-segmentations of components: ps^*

Let F() be a combination of refinement methods (F() = {AffinityNet(), Cycle(),..}.
for n **in** range(N) **do**
    $ps_n^*(i) = F(ps_n(i))$
    $ps_n(i) = ps_n^*(i)$
end for

Note that the same method can achieve the best score in multiple classes and therefore sometimes $x_i = x_j$ for $i \neq j$. Furthermore, we assessed a naive version, in which we ranked every class by its number of instances in the training set and then used the complete CAM of candidate x from the whole image if x has the high score on the highest ranked class x present on the image. The naive ISLE performed worse than our final version.

We excluded the *background* class from our ensemble since the background is the inverse of all classes combined. Note that we can perform this class selection method since we already assign the correct class labels to each prediction instead of using a classification network for the assignment, as conventional for image level based semantic segmentation methods. Therefore, it is necessary to train a fully supervised semantic segmentation network with those pseudo-labels. Nevertheless, the FSSS training guarantees that collecting multiple pseudo-label sets is a one-time effort per dataset. Fig. 2 illustrates an overview of the process. Further details can be seen in the Pseudocodes 1, 2, 3.

Algorithm 2 ISLE 2.Step

Input: N (refined) training-pseudo-segmentations of components: ps
Output: List of best scoring methods for each class: best

Let C be the number of classes in the dataset.
Let $mIoU_c()$ be the evaluation algorithm on class c.

for c **in** range(C) **do**
    $top_c = 0$
    for n **in** range(N) **do**
        for images i **do**
            $score+ = mIoU_c(ps_n(i))$
        end for
        $score = score/\#images$
        if $score > top_c$ **then**
            $top_c = score$
            $best(c) = n$
        end if
    end for
end for

Algorithm 3 ISLE 3.Step

Input: N refined training-pseudo-segmentations of components: ps^* and list of best scoring methods for each class: $best$

Output: Semantic Segmentation network trained on weakly-supervised predictions: SSN

Let ae be the final ensemble.
Define $ae = 0$ for all classes and all images in the dataset.

for c **in** range(C) **do**
    $n = best(c)$
    for images i **do**
        $ae_c(i) = ps_{n,c}(i)$
    end for
end for

4.Step:

for Epochs x **do**
    for images i **do**
        $pred = SSN(i)$
    end for
    $Backpropagation(SSN)$
end for

4 Experiments

First, we will discuss our experimental setup. We completed the experiments on a CentOS 7.9 Operating System executing on an Intel Core i7-8700 CPU with 16GB RAM and 2 Nvidia GeForce GTX 1080 Ti GPUs. The CLIMS pseudo-labels were used as provided by the official GitHub, and we performed AffinityNet with a ResNet50 backbone and dCRF on the pseudo-labels provided on the DRS and PMM GitHub. All DeepLabV3+ results were generated using a ResNet50 backbone. The mean Intersection-over-Union (mIoU) ratio is the evaluation metric for all experiments. We used the PASCAL VOC2012 semantic segmentation benchmark for evaluating our framework. It comprises twenty-one classes, including a background class, and most images include multiple objects. Following the conventional experimentation protocol for semantic segmentation, we use the $10,528$ augmented images and image-level labels, for training. Our model is evaluated on the validation set with $1,464$ images and the test set of $1,456$ images to ensure a constant comparison with the state-of-the-art.

For all experiments, the mean Intersection-over-Union (mIoU) ratio is used as the evaluation metric.

4.1 Semantic Segmentation Performance on VOC2012

Next, we compare our ensemble with its components consisting of recent works using image-level supervision Table 2. We trained all pseudo-labels with the

Table 2. Comparison of ISLE mIoU scores with state-of-the-art techniques on the VOC2012 val and test datasets. All methods were trained with DeepLabV3+ with a ResNet50 backbone for comparability. Adding more or different methods to the ensemble is possible, as those work orthogonal to the ISLE.

Method	Val	Test
PuzzleCAM	62.4	62.9
PMM	64.0	64.1
DRS	64.5	64.5
CLIMS	65.0	65.4
ISLE-2 (Ours)	66.6	67.1
ISLE (Ours)	**67.4**	**67.8**

Table 3. Semantic segmentation mIoU performance on the first 11 classes of the VOC2012 training dataset for the final pseudo-labels.

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
PuzzleCAM	88.6	79.2	43.7	89.0	61.8	72.1	83.3	76.0	92.0	29.5	86.0
CLIMS	89.8	71.2	45.4	81.7	70.2	67.6	84.0	75.7	90.0	20.3	84.2
PMM	89.5	76.8	43.9	88.1	65.8	76.0	84.2	78.0	91.2	30.6	84.3
DRS	90.1	78.9	45.3	85.8	68.4	80.8	83.8	77.4	90.6	31.5	84.0
ISLE-2 (Ours)	90.5	**79.6**	**45.4**	**89.0**	70.0	72.1	84.0	76.1	**92.2**	30.9	**86.1**
ISLE (Ours)	**91.0**	**79.6**	**45.4**	**89.0**	69.7	**81.2**	**84.2**	**78.0**	**92.2**	**31.6**	**86.1**

same DeepLabV3+ model for comparability using a ResNet50 backbone. We notice that the ensemble outperforms its component by a margin of at least 2%, although the individual components do not show this amount of variance between them. ISLE-2 is the ensemble of just PuzzleCAM and CLIMS, and ISLE is the ensemble of all four methods. DRS is the best performing of its component, and the ISLE reaches a 2% higher mIoU score.

Here, we present a more comprehensive analysis by providing a class-wise mIoU breakdown for all classes in the VOC2012 training dataset and a discussion. On the one hand, we see in Table 2 that the difference in the average mIoU score between our four component methods the relatively small, with the lowest scoring PuzzleCAM reaching 69.7% and the highest scoring DRS at 71.3%. On the other hand, the ensemble of all four methods reaches 74.1%, and the ensemble of just PuzzleCAM and CLIMS 73.6% achieves a significant gain compared to its components. Although, we also notice that the gain from adding more image-level segmentation pseudo-labels to the ensemble shrinks over time and needs to be considered when choosing the component for ISLE.

Let us take a closer look at the performance of the individual components on a class-wise basis. PMM reaches the best score only in two classes and an average 0.77% improvement in those classes. Although only reaching the highest average score in three classes, DRS provides an average 6.04% improvement in those three classes. CLIMS is the best in seven classes and achieves an average improvement of 6.04% as well. Whereas PuzzleCAM is the lowest average scoring method but reaches high scores in eight classes but improves them only by 2.62% on average.

Table 4. Semantic segmentation mIoU performance on the remaining 10 classes of the VOC2012 training dataset for the final pseudo-labels.

Method	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU
PuzzleCAM	44.0	91.6	83.1	80.1	42.8	68.9	92.6	53.4	64.8	42.6	69.7
CLIMS	57.8	86.9	80.9	80.8	72.7	48.4	90.3	56.5	68.1	58.4	70.5
PMM	48.5	89.3	82.0	79.0	61.4	66.5	89.9	54.4	66.4	38.6	70.7
DRS	41.2	88.7	80.0	79.8	65.4	62.6	89.9	55.0	77.0	41.3	71.3
ISLE-2	**52.5**	**92.0**	**86.0**	**80.9**	71.8	68.5	**92.7**	56.2	68.1	54.1	73.3
ISLE	51.1	91.9	85.9	80.8	72.3	68.5	**92.7**	**56.9**	**77.1**	52.8	**74.2**

Therefore, we conclude that CLIMS and PuzzleCAM contribute the most and PMM the least to the ensemble. Hence, we also evaluated the combination of only CLIMS and PuzzleCAM to see how much improvement we gain while combining the minimum amount of pseudo-label sets. We called this version ISLE-2. We notice that most high scores translated to the ensemble, with only minor losses in some classes, most probably due to overlap with other classes.

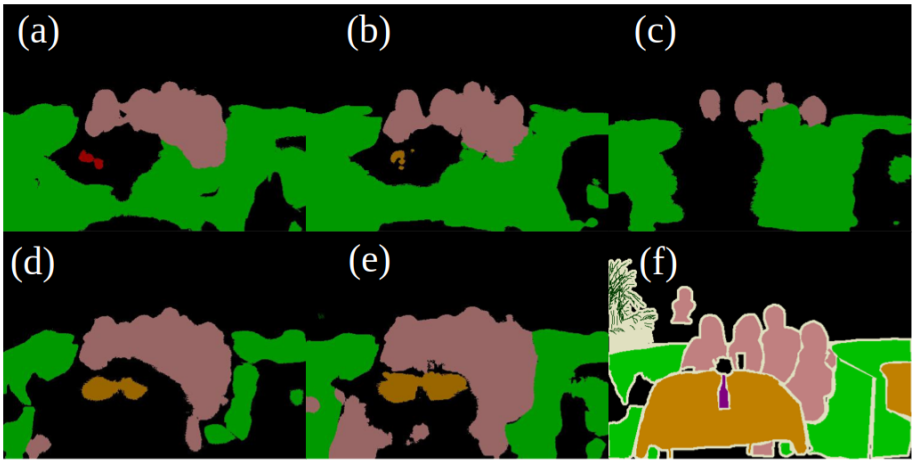

Fig. 3. Pseudo labels from (a) DRS, (b) PMM, (c) PuzzleCAM, (d) CLIMS, (e) ISLE (Ours), (f) Ground truth .

Fig. 3 presents one example of our experimental results. (a) shows the pseudo labels of DRS, (b) of PMM, (c) of PuzzleCAM, (d) of CLIMS, (e) of our ISLE, and (f) the Ground truth. We observe that DRS is not recognizing the table in the image but over-detects the couch. Like PMM, although PMM has a few pixels detected as table but also detects fewer person pixels. PuzzleCAM struggles even more with couch over-prediction and person under-prediction. The best result of the not combined images stems from CLIMs, which only correct

couch predictions and the best person detection. Finally, our Autoensemble expands person detection but also includes some over-predictions, for example in the bottom left. Furthermore, Autoensemble has the be table detection and successfully expands the couch detection.

4.2 Complexity Analysis

In this section, we will provide a complexity analysis of our code. This will help to estimate to additional complexity that comes when combining multiple methods.

Step. 1

Let $\{Comp_1, Comp_2, ..., Comp_N\}$ be the list of Components. Each component takes as input a specific image i from the list of all images I and gives as output class activation maps $CAM_n^{i,c}$ for all classes c in the dataset.

$$Comp_n(i) = \sum_{c=0}^{C} CAM_n^{i,c}, 1 \leq n \leq N$$

As the components are not further defined by the framework, we can only summarize their complexity as follows:

$$O(Step1) = \sum_{n=0}^{N} O(Comp_n^{training}(i)) \times O(Comp_n^{inference}(i)) \times epochs_n \times 2 \times I$$

Step. 2

Let $\{Ref_1, Ref_2, ..., Ref_M\}$ be the list of all applied refinements. We assume that all refinements are applied to all components for ease of notation. Then Step. 2 is defined as:

$$\widetilde{CAM_n^i} = Ref_1(CAM_n^i) \otimes Ref_2(CAM_n^i) \otimes ... \otimes Ref_M(CAM_n^i)$$

For any n with $1 \leq n \leq N$ Again, we need to define the complexity of each refinement method as $O(Ref_m())$ as the refinements are not further defined by the framework:

$$O(Step2) = \sum_{m=0}^{M} O(Ref_m^{training}(i)) \times O(Ref_m^{inference}(i)) \times epochs_n \times 2 \times I \times N$$

Step. 3

The merging of pseudo-labels is done after a class-wise evaluation for each $\widetilde{CAM_n^i}$ to determine which Component after refinement has the high score for each class c with $1 \leq c \leq C$:

$$AE(i) = \sum_{c=1}^{C} AE^c(i) = \sum_{c=1}^{C} best(\widetilde{CAM_n^{c,i}})$$

For all i in I The refinement step and Class-Wise Ensemble are just linearly dependent on the number of Components:

$$O(Step3) = O(eval) + O(merger) = I \times N \times C + I \times C$$

Step. 4

The training of the DeepLabV3+ model is not different from any other WSSS pipeline:

$$O(Step.4) = O(\texttt{DeepLabV3+}^{training} \times epochs \times images)$$

Nonetheless, for the final deployment of ISLE, only the forward pass of DeepLabV3+ is necessary, independent of the number of components ad refinements used:

$$O(Deployment) = O(\texttt{DeepLabV3+}^{inference} \times images)$$

5 Conclusion

In this paper, we have proposed our ISLE framework, which combines the pseudo-labels of several image-level segmentation techniques on a class-wise basis to leverage the strong points of its different components. The combined pseudo labels reach at least 2% higher mIoU scores than its components. Most of those gains stem from bigger variances within particular classes, as we observed that different approaches have different strengths and weaknesses. The ISLE framework combines any number of pseudo-labels to boost the quality of the pseudo-labels for final training. We showed that the predictions generated by the model trained with the pseudo labels of ISLE achieve state-of-the-art performance on the VOC2012 dataset showing its effectiveness. Our framework is open-source to ensure reproducible research and accessibility. The source code will be published at `https://anonymous.4open.science/r/ISLE-1C41/README.md`.

References

1. J. Ren, H. Gaber, and S. S. Al Jabar, "Applying deep learning to autonomous vehicles: A survey," in *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pp. 247–252, IEEE, 2021.
2. D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, and K. Dietmayer, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1341–1360, 2020.
3. F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "Resunet-a: A deep learning framework for semantic segmentation of remotely sensed data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 162, pp. 94–114, 2020.

4. R. Kemker, C. Salvaggio, and C. Kanan, "Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning," *ISPRS journal of photogrammetry and remote sensing*, vol. 145, pp. 60–77, 2018.
5. T. Meenpal, A. Balakrishnan, and A. Verma, "Facial mask detection using semantic segmentation," in *2019 4th International Conference on Computing, Communications and Security (ICCCS)*, pp. 1–5, IEEE, 2019.
6. K. Khan, M. Mauro, and R. Leonardi, "Multi-class semantic segmentation of faces," in *2015 IEEE International Conference on Image Processing (ICIP)*, pp. 827–831, IEEE, 2015.
7. A. Milioto, P. Lottes, and C. Stachniss, "Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns," in *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 2229–2235, IEEE, 2018.
8. R. Barth, J. IJsselmuiden, J. Hemming, and E. J. Van Henten, "Data synthesis methods for semantic segmentation in agriculture: A capsicum annuum dataset," *Computers and electronics in agriculture*, vol. 144, pp. 284–296, 2018.
9. A. Rehman, S. Naz, M. I. Razzak, F. Akram, and M. Imran, "A deep learning-based framework for automatic brain tumors classification using transfer learning," *Circuits, Systems, and Signal Processing*, vol. 39, no. 2, pp. 757–775, 2020.
10. Z. Zhao, S. Voros, Y. Weng, F. Chang, and R. Li, "Tracking-by-detection of surgical instruments in minimally invasive surgery via the convolutional neural network deep learning-based method," *Computer Assisted Surgery*, vol. 22, no. sup1, pp. 26–35, 2017.
11. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.
12. B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.
13. M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.
14. J. Ahn and S. Kwak, "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4981–4990, 2018.
15. S. Jo and I.-J. Yu, "Puzzle-cam: Improved localization via matching partial and full features," in *2021 IEEE International Conference on Image Processing (ICIP)*, pp. 639–643, IEEE, 2021.
16. J. Xie, X. Hou, K. Ye, and L. Shen, "Clims: Cross language image matching for weakly supervised semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4483–4492, 2022.
17. Y. Li, Z. Kuang, L. Liu, Y. Chen, and W. Zhang, "Pseudo-mask matters in weakly-supervised semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6964–6973, 2021.
18. B. Kim, S. Han, and J. Kim, "Discriminative region suppression for weakly-supervised semantic segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 1754–1761, 2021.
19. Z.-H. Yuan, T. Lu, Y. Wu, *et al.*, "Deep-dense conditional random fields for object co-segmentation.," in *IJCAI*, vol. 1, p. 2, 2017.