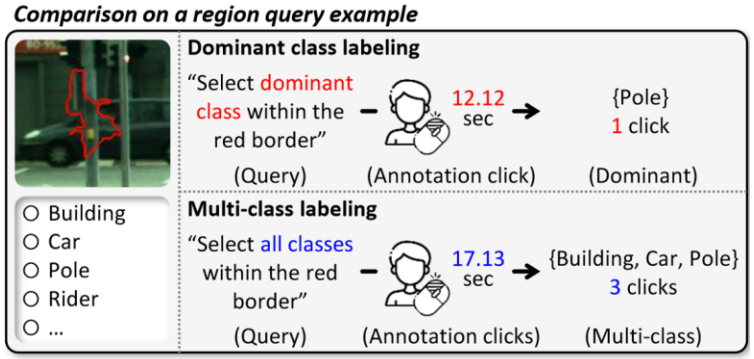
Abstract.
This paper proposes a new active learning method for semantic segmentation. The core of our method lies in a new annotation query design. It samples informative local image regions (e.g., superpixels), and for each of such regions, asks an oracle for a multi-hot vector indicating all classes existing in the region.
문제 : multi-class labeling strategy의 학습시 class ambiguity 문제
원인 : 각 픽셀마다 partial label 부여해서
Ours : *We thus propose a new algorithm for learning semantic segmentation while disambiguating the partial labels in two stages.*
We thus propose a new algorithm for learning semantic segmentation while disambiguating the partial labels in two stages. In the first stage, it trains a segmentation model directly with the partial labels through two new loss functions motivated by partial label learning and multiple instance learning. In the second stage, it disambiguates the partial labels by generating pixel-wise pseudo labels, which are used for supervised learning of the model.

- dominant class labeling vs multi-class labeling : multi-class labeling 이 더 좋아요~



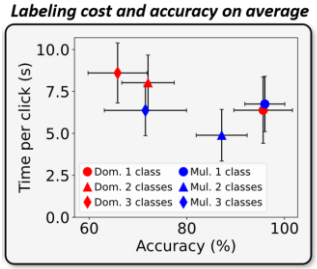
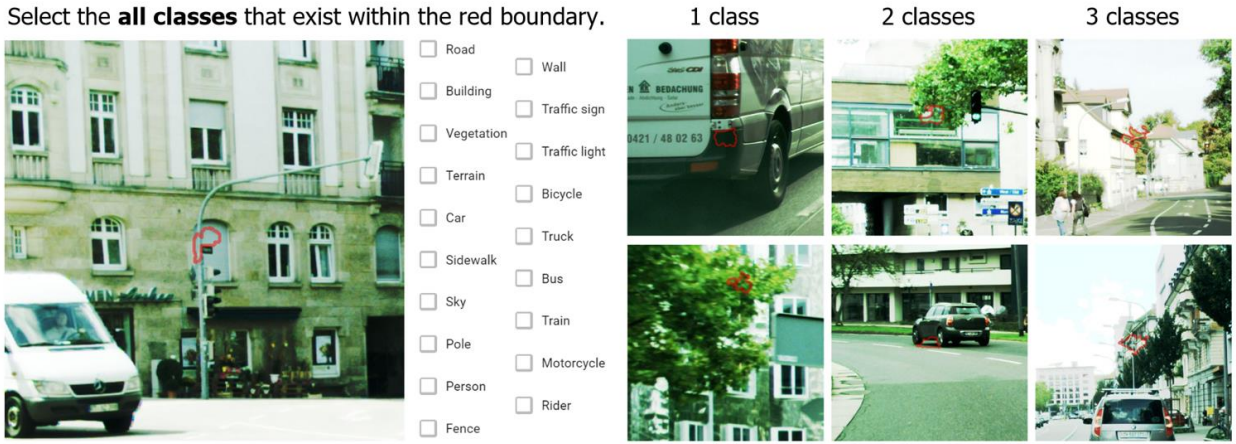


Figure 1: Dominant class labeling [9] versus our multi-class labeling. (*left*) Given a local region as query, an oracle is asked to select the most dominant class by a single click in dominant class labeling, and all existing classes by potentially more than one click in multi-class labeling. As shown here, multi-class labeling often takes less *annotation time per click* because, to determine the dominant one, the oracle has to infer every class in the region after all and sometimes should very carefully investigate the region when the classes occupy areas of similar sizes. (*right*) We conducted a user study to compare the two strategies in terms of actual labeling cost and accuracy versus the number of classes in region queries; the results are summarized in the right plot with one standard deviation. Multi-class labeling took less time per click on average due to the above reason. Furthermore, it resulted in more accurate labels by annotating non-dominant classes ignored in dominant class labeling additionally. Details of this user study are given in Appendix A.

As shown in Fig. 10(b), each image patch was a 360-pixel square mostly centered on a local region. Using ground-truth segmentation mask, we divided regions into three groups based on the number of classes (from 1 to 3) present in each region. Twenty regions were then randomly selected from each group for each survey, excluding those containing pixels irrelevant to the original 19 classes, referred to as the 'undefined' class.



(a) Questionnair for multi-class labeling (b) Example queries with different number of classes

Figure 10: Questionnaire and local region examples used in the user study. (a) Questionnaire of multi-class labeling survey, consisting of instruction, image patch along with local region marked with red boundary, and class options allowing multiple selections. (b) Examples of local regions used in the user study according to the number of classes present in each region.

As shown in Table 3, multi-class labeling demonstrates comparable efficiency to dominant class labeling for regions with a single class. Moreover, when it comes to regions with multiple classes, multi-class labeling requires less annotation time per click compared to the dominant class labeling.

Table 3: The result of user study showing the labeling time (second) and accuracy (%) of dominant class labeling and multi-class labeling according to the number of classes within each region.

Query	# of classes	Total time (s)	Total clicks	Time per click (s)	Accuracy (%)
Dominant	1	127.6±39.4	20.0±0.0	6.38±1.97	95.63±6.00
	2	160.5±33.0	20.0±0.0	8.02±1.65	72.05±5.36
	3	172.1±35.8	20.0±0.0	8.60±1.79	65.83±6.07
	average	153.4±41.1	20.0±0.0	7.67±2.05	77.84±14.24
Multi-class	1	145.5±41.8	21.6±2.1	6.75±1.65	95.97±4.10
	2	191.6±65.8	39.1±3.7	4.89±1.54	87.14±5.21
	3	295.8±65.3	49.0±8.6	6.37±1.51	71.52±8.42
	average	211.0±86.3	36.5±12.5	6.01±1.75	84.88±11.76

multi-class labeling strategy

Introduction.

In this context, we first introduce a new query design for region-based AL of semantic segmentation. The essence of our proposal is to ask the oracle for a multi-hot vector that indicates all classes existing in the given region. This multi-class labeling strategy enables to prevent annotation errors for local regions capturing multiple classes, and works the same as dominant class labeling (and thus inherits its advantages) for single-class region queries. Moreover, our user study revealed that multi-class labeling demands less annotation time per click and results in more accurate labels compared with dominant class labeling as demonstrated in Fig. 1. However, such region-wise multi-class labels introduce a new challenge in training, known as the class ambiguity issue, since they assign partial labels [16, 29] (i.e., a set of candidate classes) to individual pixels of the selected regions.

To address the ambiguity issue, we propose a new AL method tailored to learning semantic segmentation with partial labels. Fig. 2 illustrates the overall pipeline of the proposed method. Given a set of local regions and their multi-class labels, our method trains a segmentation network in two stages. In the first stage, the network is trained directly with the region-wise multi-class labels. To this end, we propose two new loss functions for the label disambiguation based on the notions of partial-label learning [16, 29] and multiple instance learning [19], respectively. In the second stage, our method disambiguates the partial labels through pseudo segmentation labels, which are used to train the segmentation network in the supervised learning fashion. To be specific, it finds a set of class prototype features from each local region using the model of the first stage, and employs the prototypes as a region-adaptive classifier to predict pixel-wise pseudo labels within the region. In addition, we propose to propagate the pseudo labels to neighboring local regions to increase the amount of supervision given per query; this strategy benefits by multi-class labeling that enables to propagate pseudo labels of multiple classes, leading to larger expansion of pseudo labels. Last but not least, we introduce an acquisition function designed to maximize the advantage of multi-class labels in the region-based AL. Our acquisition function considers both uncertainty [32,62] and class balance [7, 66, 67] of sampled regions so that local regions where the model finds difficult and containing underrepresented classes are chosen more frequently.

In short, the main contribution of this paper is five-fold:

- *We introduce a new query design for region-based AL in semantic segmentation, which asks the oracle for a multi-hot vector indicating all classes existing within a particular region.*
- *We propose a novel AL framework that includes two new loss functions effectively utilizing the supervision of multi-class labels and a method for generating pseudo segmentation labels from the multi-class labels, resulting in enhanced supervision.*
- *To maximize the advantage of multi-class labels, we design an acquisition function that considers multiple classes of a local region when examining its uncertainty and class balance.*
- *The effectiveness of multi-class labeling was demonstrated through extensive experiments and user study in real-world annotation scenarios.*
- *The proposed framework achieved the state of the art on both two public benchmarks, Cityscapes and PASCAL VOC 2012, with a significant reduction in annotation cost.*

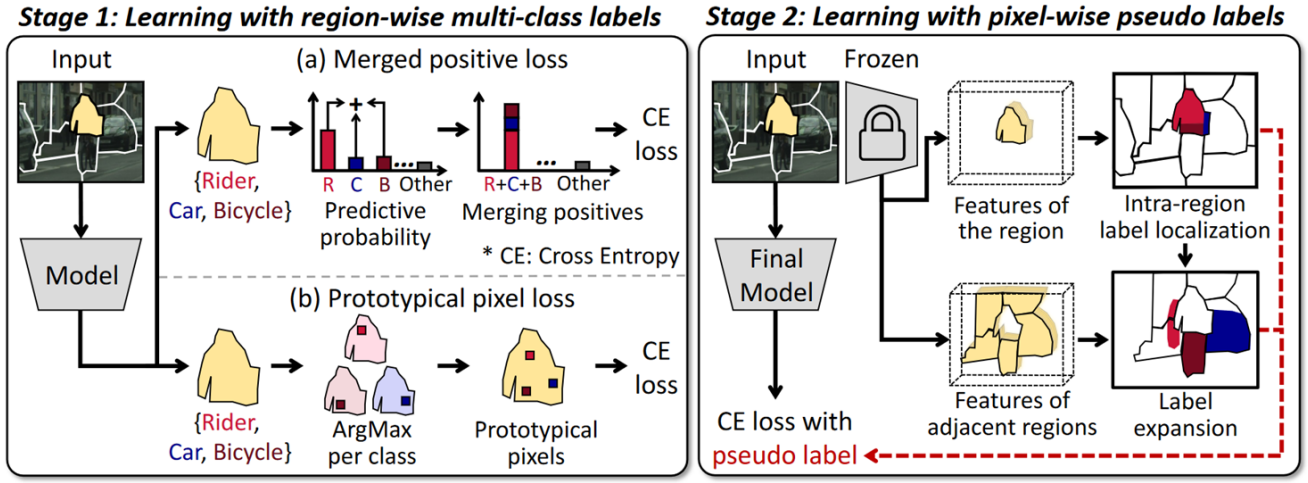


Figure 2: Our two-stage training algorithm using partial labels. (*left*) In the first stage, a model is trained using region-wise multi-class labels through two losses: the merged positive loss that encourages the model to predict any of the annotated classes for each pixel of the region, and the prototypical pixel loss that ensures at least one pixel in the region corresponds to each annotated class. (*right*) The second stage disambiguates the region-wise multi-class labels by generating pixel-wise pseudo labels, which are then used for training the final model. To this end, it first assigns pseudo class labels to individual pixels within the region (*i.e.*, intra-region label localization), and then propagates the pseudo labels to adjacent regions (*i.e.*, label expansion).

2 Related Work

Active learning (AL).
For structured prediction tasks like semantic segmentation, queries should be more carefully designed to optimize cost-effectiveness of annotation.
Active learning for semantic segmentation.
Our paper proposes a new cost-effective region query that allows more accurate and faster annotation, and a new training algorithm taking full advantage of the query design.
Partial label learning.
Our two-stage training algorithm addresses these issues by disambiguating partial labels by pseudo labeling.

3 Proposed Method

We consider an AL process with R rounds. At each round, local regions of a batch are selected using an acquisition function, and then a multi-hot vector (i.e., multi-class label) is assigned to each of them by an oracle. Given the labeled regions, our training algorithm operates in two stages as illustrated in Fig. 2. In the first stage, a segmentation model is trained directly with the region-wise multi-class labels by two loss functions specialized to handle the ambiguity of the labels (Sec. 3.2). In the second stage, the ambiguity is mitigated by generating pixel-wise pseudo labels and using them for training the model further (Sec. 3.3).

3.1 Acquisition of region-wise multi-class labels

$\mathcal{I}$: unlabeled image set

$S(I)$: a set of non-overlapping regions of the image $I \in \mathcal{I}$

Such a nonoverlapping partition can be obtained by a superpixel algorithm as in [9].

$\mathcal{S} := \bigcup_{I \in \mathcal{I}} S(I)$: the set of all the partitions for $\mathcal{I}$

$\mathcal{B}_t \subset \mathcal{S}$: a batch of regions for each round t : queried to acquire a multi-class label $Y \subset \{1, 2, \cdots, C\}$

$|Y| \geq 1$

C : the number of classes

θ_t : model is trained using the labeled regions $\mathcal{D} := \bigcup_t \mathcal{D}_t$

$\mathcal{D}_t := \{(s, Y) : s \in \mathcal{B}_t\}$: pairs of region and associated multi-class label

θ_t : model includes feature extractor and classifier

$f_t(\cdot)$: with weight matrix $[\mathbf{w}_{t,1}, \mathbf{w}_{t,2}, \cdots, \mathbf{w}_{t,C}] \in \mathbb{R}^{d \times C}$

$$P_{\theta_t}(y = c|x) = \text{softmax}\left(\frac{f_t(x)^\top \mathbf{w}_{t,c}}{\tau \|f_t(x)\| \, \|\mathbf{w}_{t,c}\|}\right), \quad (1)$$

: predictive probability of pixel x being class c

τ : temperature

Acquisition function.
acquisition function for selecting a batch of regions $\mathcal{B}_t \subset \mathcal{S}$ at round t

1) uncertainty measure : best-versus second-best (BvSB)

$$u_{\theta_t}(x) := \frac{P_{\theta_t}(y = c_{\text{sb}}|x)}{P_{\theta_t}(y = c_{\text{b}}|x)}, \quad (2)$$

c_{b} and c_{sb} : the classes with the largest and second-largest predictive probabilities for x under θ_t

2) class-balanced sampling

$$P_{\theta_t}(y = c) = \frac{1}{|X|} \sum_{x \in X} P_{\theta_t}(y = c|x) \ , \tag{3}$$

$$X := \{x : \exists s \in \mathcal{S},\ x \in s\}$$

Our acquisition function, favoring uncertain regions of rare classes, is defined as

$$a(s; \theta_t) := \frac{1}{|s|} \sum_{x \in s} \frac{u_{\theta_t}(x)}{\left(1 + \nu\, P_{\theta_t}(c_b)\right)^2} \ , \tag{4}$$

ν : hyperparameter regulating the class balancing effect

Distinct from an existing acquisition function [9] that considers the dominant class only, our function considers classes of all pixels in a region and thus better aligns with multi-class labeling.

3.2 Stage 1: Learning with region-wise multi-class labels

$$\mathcal{D}_{\mathrm{s}} := \left\{(s, \{c\}) : \exists (s, Y) \in \mathcal{D},\ |Y| = 1,\ c \in Y\right\}. \tag{5}$$

: the set of local regions equipped with single-class labels

$$\mathcal{L}_{\mathrm{CE}} = \hat{\mathbb{E}}_{(s,\{c\})\sim\mathcal{D}_{\mathrm{s}}}\left[\frac{1}{|s|}\sum_{x\in s} -\log P_\theta(y = c|x)\right]. \tag{6}$$

: pixel-wise CE loss

$\mathcal{D}_{\mathrm{m}} := \mathcal{D} - \mathcal{D}_{\mathrm{s}}$: regions labeled with multiple classes

Regions labeled with multiple classes cannot be used for training using the pixel-wise CE loss, since a multi-class label lacks precise correspondence between each pixel and class candidates, making it a weak label [16, 19, 29]. To effectively utilize the supervision of Dm, we introduce two loss functions.

Merged positive loss.

$$\mathcal{L}_{\mathrm{MP}} := \hat{\mathbb{E}}_{(s,Y)\sim\mathcal{D}_{\mathrm{m}}}\left[\frac{1}{|s|}\sum_{x\in s} -\log \sum_{c\in Y} P_\theta(y = c|x)\right]. \tag{7}$$

: merged positive loss

This loss encourages to predict any class from the candidate set since the predictive probability of every candidate class is considered as positive.

Prototypical pixel loss.
Learning with regions assigned with multi-class labels can be considered as an example of multiple instance learning (MIL) [19], where each region is a bag, each pixel in the region is an instance, and at least one pixel in the region must be positive for each candidate class.

prototypical pixel : the pixel with the most confident prediction for each candidate class within the region

$$x^*_{s,c} := \max_{x \in s} P_\theta(y = c|x) \ , \quad (8)$$: prototypical pixel $c \in Y$ and $(s, Y) \in \mathcal{D}_\mathrm{m}$

The segmentation model is trained by applying the CE loss to each prototypical pixel with the assumption that the class associated with it is true.

$$\mathcal{L}_\mathrm{PP} := \hat{\mathbb{E}}_{(s,Y)\sim\mathcal{D}_\mathrm{m}} \left[\frac{1}{|Y|} \sum_{c \in Y} -\log P_\theta(y = c|x^*_{s,c}) \right] . \quad (9)$$: prototypical pixel loss

As reported in the literature of MIL [19], although the prototypical pixels may not always match the ground truth, it is expected that training with numerous prototypical pixels from diverse regions enables the model to grasp the underlying concept of each class.
Moreover, this loss mitigates the class imbalance issue as it ensures that every candidate class equally contributes to training via a single prototypical pixel in a region, leading to a balanced class representation.

$$\mathcal{L} = \lambda_\mathrm{CE}\, \mathcal{L}_\mathrm{CE} + \lambda_\mathrm{MP}\, \mathcal{L}_\mathrm{MP} + \mathcal{L}_\mathrm{PP} \ , \quad (10)$$: total training loss of the first stage

λ_CE and λ_MP : balancing hyperparameters

3.3 Stage 2: Learning with pixel-wise pseudo labels

In the second stage, we disambiguate the partial labels by generating and exploiting pixel-wise one hot labels.
The pseudo label generation process comprises two steps: intra-region label localization and label expansion
1) intra-region label localization : assigns pseudo class labels to individual pixels within each labeled region
2) label expansion : spreads the pseudo labels to unlabeled regions adjacent to the labeled one

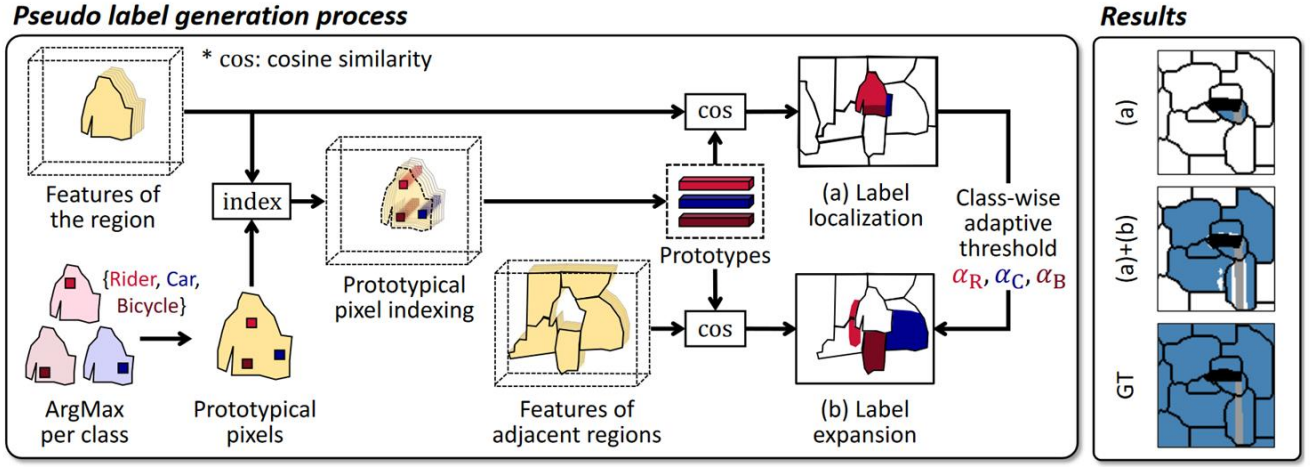


Figure 3: The pseudo label generation process (*left*) and its qualitative results (*right*). In each of the labeled regions, the feature vector located at the prototypical pixel of an annotated class is considered the prototype of the class, and the set of such prototypes is used as a region-adaptive classifier for pixel-wise pseudo labeling within the region (label localization). The pseudo labels of the region are propagated to adjacent unlabeled regions similarly (label expansion), but for conservative propagation, only relevant pixels that are close to at least one prototype will be assigned pseudo labels.

Intra-region label localization.

For each of the labeled regions, we define a prototype for each annotated class as the feature vector located at the prototypical pixel of the class, which is estimated by Eq. (8) using the model of the first stage.
The set of such prototypes is then used as a region-adaptive classifier, which is dedicated to pixel-level classification within the region.
To be specific, we assign each pixel the class of its nearest prototype in a feature space

$$\hat{y}(x) := \arg\max_{c \in Y} \cos\big(f_\theta(x), f_\theta(x^*_{s,c})\big) \ , \qquad (11)$$

: assigned pseudo label for $x \in s \quad (s, Y) \in \mathcal{D}$

$x^*_{s,c}$: prototypical pixel of class c

$\cos(f, f') = \frac{f^\top f'}{\|f\| \, \|f'\|}$: cosine similarity between two feature vectors f and f ′

Label expansion.

class composition Y of a region $(s, Y) \in \mathcal{D}$ may provide clues about classes of its adjacent regions $s' \in \mathrm{N}(s) \qquad \mathrm{N}(s) \cap \mathcal{D} = \emptyset$

$\mathrm{N}(\cdot)$: a set of unlabeled regions adjacent to s

$$\alpha_c(s) = \mathrm{med}\Big(\big\{\cos\big(f_\theta(x), f_\theta(x^*_{s,c})\big) : x \in s, \ \hat{y}(x) = c\big\}\Big) \ , \qquad (12)$$

: prototype-adaptive threshold for class $c \in Y$ in $(s, Y) \in \mathcal{D}$

$\mathrm{med}(\cdot)$: median value of a set

$x^*_{s,c}$: prototypical pixel of class c (Eq. (8)) $\hat{y}(x)$: pseudo label of x (Eq. (11))

We propagate pseudo labels of the labeled region $s \text{ in } \mathcal{D}$ to pixels of an adjacent region $\{x : \exists s' \in \mathrm{N}(s),\ x \in s'\}$ by

$$\hat{y}(x) := \underset{c \in \hat{Y}(x)}{\arg\max}\ \cos\big(f_\theta(x), f_\theta(x^*_{s,c})\big) \quad \text{only if } |\hat{Y}(x)| \geq 1\ , \tag{13}$$

$$\hat{Y}(x) := \big\{c : \cos\big(f_\theta(x), f_\theta(x^*_{s,c})\big) > \alpha_c(s),\ c \in Y\big\};\ x \text{ is a relevant pixel if } |\hat{Y}(x)| \geq 1$$

The segmentation model is then further trained using the pixel-wise CE loss with pseudo segmentation labels generated by both of the label localization and expansion steps.