MetaAT: Active Testing for Label-Efficient Evaluation of Dense Recognition Tasks

Sanbao Su[1⋆], Xin Li[2], Thang Doan[2], Sima Behpour[2], Wenbin He[2], Liang Gou[2], Fei Miao[1], and Liu Ren[2]

[1] University of Connecticut
{sanbao.su,fei.miao}@uconn.edu
[2] Bosch Research North America & Bosch Center for Artificial Intelligence (BCAI)
{xin.li9,thang.doan,sima.behpour,wenbin.he2,liang.gou,liu.ren}@us.bosch.com

Abstract. In this study, we investigate the task of active testing for label-efficient evaluation, which aims to estimate a model's performance on an unlabeled test dataset with a limited annotation budget. Previous approaches relied on deep ensemble models to identify highly informative instances for labeling, but fell short in dense recognition tasks like segmentation and object detection due to their high computational costs. In this work, we present MetaAT, a simple yet effective approach that adapts a Vision Transformer as a Meta Model for active testing. Specifically, we introduce a region loss estimation head to identify challenging regions for more accurate and informative instance acquisition. Importantly, the design of MetaAT allows it to handle annotation granularity at the region level, significantly reducing annotation costs in dense recognition tasks. As a result, our approach demonstrates consistent and substantial performance improvements over five popular benchmarks compared with state-of-the-art methods. Notably, on the CityScapes dataset, MetaAT achieves a 1.36% error rate in performance estimation using only 0.07% of annotations, marking a 10× improvement over existing state-of-the-art methods. To the best of our knowledge, MetaAT represents the first framework for active testing of dense recognition tasks.

1 Introduction

The success of modern machine learning is largely driven by extensive annotated datasets [18]. However, accurate and detailed instance annotations remain slow and expensive [9, 25, 26]. In efforts to improve efficiency and reduce costs, substantial research in active learning (AL) has been carried out to optimize the selection of training data [2]. However, the selection of test data has been mostly overlooked. In this work, we focus on the practical task of active testing (AT) for label-efficient evaluation (Figure 1a). The core objective is to select instances for labeling from an unlabeled test dataset, enabling accurate estimation of model performance across the entire dataset within a limited annotation budget. This task is crucial in the lifecycle of machine learning, as it promises to significantly reduce the time and costs associated with label annotation [18].

⋆ Work done while interning at Bosch Research North America.

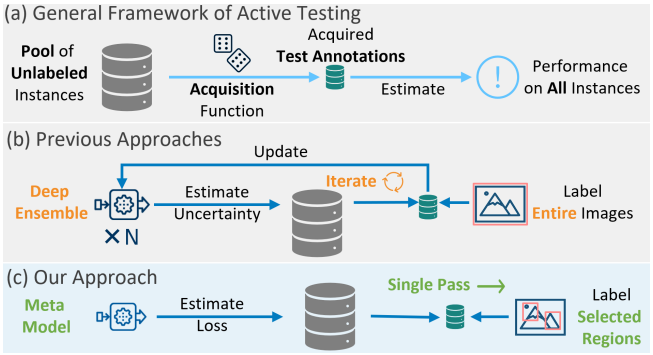

Fig. 1: Overview of active testing framework and comparison between previous approaches and our approach. (a) Active Testing selects a subset of test instances to be labeled by the acquisition function, to estimate test performance on the entire test dataset. (b) Previous approaches are typically designed for classification tasks, requiring multiple iterations of ensemble model training and entire image labeling. (c) Our approach can be used for dense recognition tasks. With a meta model, it enables a single-pass evaluation that requires only selected regions of the image to be labeled.

The state-of-the-art approaches in AT [23, 24] have focused mainly on image classification models, using deep ensemble models to identify informative (high-loss) instances for labeling (Figure 1b). For instance, the Active Surrogate Estimator (ASE) [23] employs a weighted epistemic uncertainty score, estimated by ensemble models, to efficiently pinpoint informative instances. A key feature of these approaches is the iterative updating of ensemble models with newly acquired labels, which helps reduce overconfidence and enhances the prediction for unseen test data. While these methods demonstrate strong performance in evaluating image classification model, their effectiveness is limited in dense recognition tasks, such as segmentation and object detection, due to inherent challenges in both instance and label acquisition stages.

First, regarding instance acquisition, deep ensemble models are often impractical due to high computational costs [37, 40] and the challenge of achieving sufficient diversity within the ensemble models for dense recognition tasks [1]. Additionally, updating these models with few newly labeled instances per iteration may provide insufficient information for retraining [16]. Second, from a label acquisition perspective, iterative processes increase the communication overhead between researchers and annotators. Moreover, previous approaches that involve labeling entire images are inefficient for tasks where only specific regions, such as borders and areas of ambiguity, carry the majority of the test error [35].

To overcome these limitations, we propose a meta-model-based active testing method for label-efficient evaluation, MetaAT, which is carefully designed for evaluating dense recognition tasks. The core idea is to develop a **meta model** that can be used to identify highly informative images or regions by estimating the loss over the unlabeled test dataset **in a single pass**.

Specifically, we adapt a vision transformer (ViT) [12] as our meta model for its strength in addressing long-range dependencies. This capability is crucial for linking small, critical regions like object boundaries that often lead to errors in dense recognition tasks [7,36], thus resulting in more accurate test error estimation. Furthermore, MetaAT offers several additional key advantages compared to previous approaches in both the instance and label acquisition stages (Figure 1c). First, it significantly reduces the additional training cost of deep ensemble models. Importantly, the consistent performance of our model eliminates the need for iterative retraining, thereby greatly reducing the communication overhead mentioned before. Second, the inherent characteristics of the ViT, which processes images in small patches, allow us to estimate image level and region level losses simultaneously. This approach enables us to select only informative regions in the image for labeling, thus largely reducing annotation costs.

In summary, the main contributions of this paper are threefold.

1. We propose the meta-model-based approach for active testing for label-efficient evaluation. To the best of our knowledge, MetaAT is a pioneering work designed to efficiently evaluate dense recognition tasks, including both segmentation and object detection.
2. We demonstrate MetaAT's remarkable flexibility across various levels of annotation granularity for testing datasets, ranging from labeling entire regions in an image to annotating only a few portions of these regions. The latter significantly lowers the costs without compromising performance, representing a substantial leap in label-efficient evaluation.
3. We extensively benchmark our method using various models and datasets, where our method consistently outperforms current state-of-the-art methods.

2 Related Work

Active Learning actively selects the highest-value training instances for labeling during the training process and thus optimizes the balance between annotation costs and model performance. In general, AL strategies can be categorized into two main lines: The first line involves methods [4,38,42] that employ a score function, such as loss or uncertainty, to identify and select informative instances. The second line, represented by works [5, 34], focuses on selecting a diverse set of instances that represent the overall distribution of the dataset. Notably, some studies [22, 32] have attempted to address both informativeness and diversity.

Active Testing represents a pioneering paradigm in label-efficient model evaluation, where the objective is to estimate the model's performance on the entire unlabeled test dataset with a limited annotation budget. The distinction between active learning and active testing highlights key challenges in directly applying active learning methods to active testing tasks. Active learning typically aims to select the most challenging or uncertain instances, focusing on the ranking of all instances. In contrast, active testing requires accurate value estimation. For instance, LLAL [42] introduced a loss prediction approach for active learning. They developed a ranking-based loss function that prioritizes the relative value

of the loss of two instances, rather than estimating their true loss value. However, this approach can lead to a shift in the loss distribution across all instances, resulting in a highly biased estimation of performance. Conversely, AT focuses on precisely estimating the loss value, providing a more accurate understanding of the loss distribution for all instances.

Of particular relevance are two current state-of-the-art approaches: Active Testing Surrogate (ATS) [24] and Active Surrogate Estimator (ASE) [23]. These methods leverage deep ensemble capabilities to identify informative (high-loss) test instances for labeling. ATS, for instance, computes the cross-entropy loss between the predicted distribution of deep ensemble models and the target model output to select informative instances. On the other hand, ASE expands this approach by using deep ensemble models to assess the epistemic uncertainty of unlabeled instances. It introduces an eXpected WEighted Disagreement (XWED) score that combines estimated loss and epistemic uncertainty for more effective instance acquisition. Both methods improve their ensemble models by iteratively updating with information from newly labeled test instances during evaluation.

Although their methods achieve great performance in assessing classification models, they are not suitable for more dense recognition tasks such as segmentation and object detection. As discussed in Section 1, the main reason is the difficulty of training ensemble models in those tasks, both in terms of computational cost and performance. In addition, they are designed for labeling the entire instance, which can be unnecessary, as usually only small regions from each image contribute most to the test error in dense recognition tasks.

3 Methodology

An overview of our approach is shown in Figure 2. In this section, first, we formally define the active testing task. Next, we present the framework of MetaAT. Finally, we provide a detailed explanation of two key components of MetaAT: the Vision Transformer-based Meta Model and the Subsample Risk Estimator.

3.1 Active Testing

We start with a **target model** that we wish to evaluate, $f : X \to Y$, which maps inputs $x \in X$ to its corresponding labels $y \in Y$. We make no assumption about this target model; the only requirement is that we can obtain its prediction for any given input. Our goal is to estimate some model evaluation statistics with a limited annotation budget. For generality, we can estimate the expected loss of the model predictions commonly as the **risk** [39]: $R = \mathbb{E}[\mathcal{L}_f(f(x), y)]$, where $\mathcal{L}_f$ is the loss function of f, and the expectation is over the true test distribution. Because our objective is to estimate the risk of the target model f, the model itself remains unchanged during the evaluation process. If we had all labels for the entire test dataset, denoted as $\mathcal{D}_{test}$, we could simply compute the risk as follows:

$$\hat{R}_{test} = \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}_f(f(x_n), y_n), (x_n, y_n) \in \mathcal{D}_{test}, \tag{1}$$

which is an unbiased estimate of the true risk where N is the number of instances in $\mathcal{D}_{test}$. However, evaluating $\hat{R}_{test}$ in this manner is prohibitively expensive, as the cost associated with label acquisition makes it impractical to label all instances in $\mathcal{D}_{test}$. Instead, AT methods strategically select a subset of instances $\mathcal{D}_{test}^{observed} \subseteq \mathcal{D}_{test}$ to be labeled. Typically $M = |\mathcal{D}_{test}^{observed}| \ll |\mathcal{D}_{test}| = N$.

A naive approach to reduce annotation cost would be randomly sampling (RS) M instances in the test dataset for labeling and then calculating the subsample empirical risk:

$$\hat{R}_{iid} = \frac{1}{M} \sum_{m=1}^{M} \mathcal{L}_f(f(x_m), y_m), (x_m, y_m) \in \mathcal{D}_{test}^{observed}. \tag{2}$$

Although this Monte Carlo estimation is unbiased since selected instances are independently and identically distributed (i.i.d.), its *variance* can be unacceptably high under the setting of $M \ll N$ [17].

3.2 Overview of MetaAT

Our MetaAT integrates two key components: the **Meta Model** and the **Subsample Risk Estimator**. The meta model is designed to reduce the high variance associated with limited labels in AT, while the subsample risk estimator counters potential biases from the meta model. Together, they create an estimator with both low variance and minimal bias.

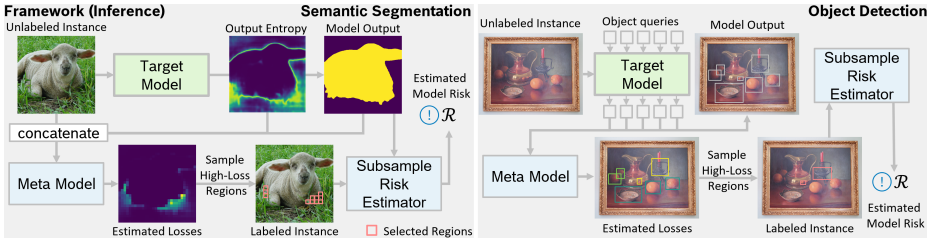

Fig. 2: The framework of our MetaAT.

Figure 2 displays the MetaAT workflow. Given an unlabeled test dataset, our vision transformer-based meta model accurately predicts the losses for all instances. It leverages the output of the target model, either independently for object detection or in combination with input images for semantic segmentation. This allows the identification of highly informative (high-loss) instances, significantly cutting down the variance from the RS mentioned before. However, directly selecting highly informative instances for labeling and computing the risk $\hat{R}$ could introduce high bias, as these instances may be treated as "hard cases", potentially leading to an overestimation of the risk $\hat{R}$ for the entire test dataset. To mitigate this, our subsample risk estimator doesn't simply average

Algorithm 1: MetaAT: Active Testing for Label-efficient Evaluation

Data: labeled dataset $\mathcal{D}_{train}(X_{train}, Y_{train})$, unlabeled dataset $\mathcal{D}_{test}(X_{test})$ with S images, Target Model f with its loss function $\mathcal{L}_f$, number of training steps T, our meta model v_θ with the loss function $\mathcal{L}_{ViT}$, batch size B, annotation budget M, region feature r

Result: the estimated risk $\hat{R}_{MetaAT}$

`/* Train` v_θ `on` $\mathcal{D}_{train}$ `and` f `*/`

for $t = 1$ *to* T **do**

  Sample a mini-batch $\{x_b, y_b\}_{b=1}^{B}$ from $\mathcal{D}_{train}$

  `/* For segmentation,` $r = [x, f(x), entropy(f(x))]$`; For object detection,` $r =$ `query features from` $f(x)$ `*/`

  Compute overall loss of v_θ $\mathcal{L}_{ViT}(v_\theta(r(x_b, f)), \mathcal{L}_f(f(x_b), y_b))$ as Eq. 4

  Update θ of v_θ to minimize $\mathcal{L}_{ViT}$

end

`/* Estimate losses for all instances in` $\mathcal{D}_{test}$`. For image level,` $N = S$ `and each instance is one image; For region level,` $N =$ `number of considered regions in all images and each instance is one region in the image */`

for $n = 1$ *to* S **do**

  Append $v_\theta(r(x_s, f))$ into q, where $x_s \in X_{test}$

end

$N \leftarrow |q|$

`/* Select instances from` $\mathcal{D}_{test}$ `*/`

for $m = 1$ *to* M **do**

  Select instance i_m with probabilities defined by the distribution q ($i_m \in [1, N]$)

  Add instance i_m to $\mathcal{D}_{test}^{observed}$ and remove q_{i_m} from q to avoid repeated selection

end

`/* Compute the risk` $\hat{R}_{MetaAT}$ `*/`

Label all instances in $\mathcal{D}_{test}^{observed}$

Compute $\hat{R}_{MetaAT}$ with the subsample risk estimator on $\mathcal{D}_{test}^{observed}$ as Eq. 5

all losses equally as outlined in Eq. 2; it computes a weighted average based on the loss distribution predicted by the meta model. The specifics of the meta model and the subsample risk estimator are detailed in subsequent subsections.

3.3 Meta Model

Backbone. As discussed in the previous subsection, our meta model aims to predict losses for unseen test data (without labels). In dense recognition tasks, these losses often stem from small, uncertain regions like object boundaries [13]. To address this, we have adapted the Vision Transformer (ViT) as our meta model due to its proficiency in handling long-range dependencies in images. This capability allows meta model to effectively relate those regions to the broader image context, thus improve the accuracy of error prediction. Furthermore, ViT's unique approach to processing images in discrete patches enables the concurrent estimation of losses at both the image and regional levels, serving as mutual regularization during training and enhancing the overall loss estimation.

Inputs. Under the AT setting, it is assumed that there are no labels at the beginning of the evaluation. As shown on the left side of Figure 3, our meta model estimates the segmentation loss based on (1) the input image, (2) the target model's output, and (3) the output entropy, which measures uncertainty [8]. Specifically, the RGB channels of the input image, the class predictive distribution, and its corresponding entropy value are concatenated together. We then split these combined inputs into E flattened patches, with each patch corresponding to a distinct region within the image. These patches undergo a trainable linear projection, mapping them to tokens of D dimensions, which are then added with position embeddings as the input for the transformer encoder. Note that a class token is added at position 0 to serve as a representation of the entire image, which can be used to predict the whole image loss later. For object detection, we employ object query features extracted by the target model, as detailed in [7]. These query features, which serve as input for the meta model, represent the learned representations of potential objects in corresponding regions [43].
Outputs. Labeling the entire image may not be necessary and can be resource-intensive in the context of segmentation and object detection [30, 35]. To facilitate region level selection and annotation, we have modified the original ViT model by introducing a trainable Multilayer Perceptron (MLP) head as our **region loss estimation head** after the transformer encoder. This head estimates the region level loss for all split patches or object queries within the input image. With this region level estimation, instead of selecting highly informative images to label, we can now focus on selecting highly informative regions to label, thus significantly reducing the annotation cost. On the other hand, for training the meta model, as shown in Figure 3, we generate a ground truth loss map using a segmentation or object detection loss function from the target model. This map is then split and arranged in the same order as the input.

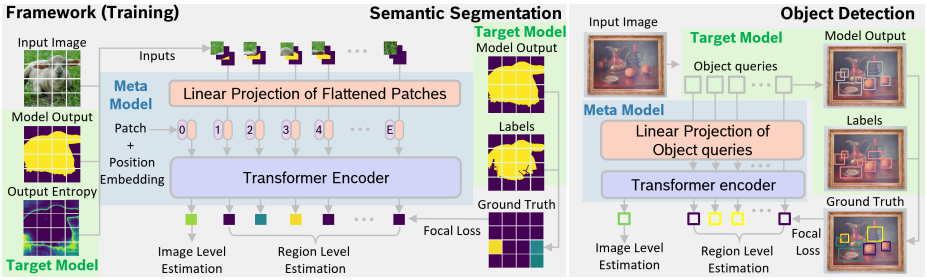

Fig. 3: The training process for the meta-model: "labels" refer to the labels used in the target model (segmentation/object detection), whereas "ground truth" pertains to the actual loss values that the meta-model aims to predict.

Loss Function. As mentioned, small and challenging regions disproportionately impact test errors in segmentation and object detection, leading to a highly imbalanced loss distribution at the region level. This imbalance presents the chal-

lenge for training the meta model using regression on original loss values. The issue arises because regression techniques are particularly sensitive to skewed distributions and can be overly influenced [41]. To address this issue, we apply the **equal-width binning** [18] technique to convert the ground truth loss into discrete classes, denoted as c. During training, the transformer encoder generates one class $\hat{c}_0$ for the entire image and E classes $\{\hat{c}_e\}_{e=1}^{E}$ for all regions. These classes can then be converted back to numerical values during the inference. To further mitigate the highly imbalanced issue, we employ the **focal loss function** [27], which focuses on hard examples by dynamically reweighing the loss during the training. Additionally, we train the image head and the region head simultaneously as a multitask learning approach. Thus, they can serve as each other's regularizers. The final loss function is formulated as follows:

$$\mathcal{L}_{ViT} = \mathcal{L}_{\text{image}} + \mathcal{L}_{\text{region}} = FL(c_0, \hat{p}_0) + \frac{1}{E}\sum_{e=1}^{E} FL(c_e, \hat{p}_e) \tag{3}$$

where FL is the focal loss function, c_0 and $\hat{p}_0$ are the ground truth and predicted class distributions for the entire image, while c_e and $\hat{p}_e$ are those for each region.

3.4 Subsample Risk Estimator

After estimating the losses for all unlabeled test instances, the pivotal task involves sampling from $\mathcal{D}_{test}^{observed}$ for labeling, followed by computing the risk $\hat{R}$. To avoid bias introduced by selecting instances with high loss, we employ LURE [15] as our Subsample Risk Estimator

$$\hat{R}_{MetaAT} = \frac{1}{M}\sum_{m=1}^{M} v_m \mathcal{L}_f(f(x_m), y_m), (x_m, y_m) \in \mathcal{D}_{test}^{observed}, \tag{4}$$

$$v_m = 1 + \frac{N-M}{N-m}\left(\frac{1}{N-m+1} q_{i_m} - 1\right), i_m \in [1, N]. \tag{5}$$

where q_{i_m} is the predicted loss of given images or regions. As proved in [15], LURE effectively mitigates selection bias through corrective weighting with v_m. Furthermore, its capability extends to variance reduction, given its foundation on importance sampling, a technique explicitly crafted to diminish variance [31].

4 Experiments

4.1 Experimental Setups

Target Model and Dataset. We assess the effectiveness of our MetaAT approach through comprehensive experiments on both semantic segmentation and object detection tasks. In semantic segmentation, we evaluate various classical target models, including UNet [33], PSPNet [44], SEGNet [3], and FCN [29], operating on diverse datasets such as Pascal VOC [14], CityScapes [10], ADE20K [45],

and COCO [6]. For the object detection task, we utilize target models, including DETR [7], Deformable-DETR [46] and DINO [43], and estimate their risks on the COCO Detection dataset [28]. In both cases, we measure the ground-truth risk using the loss function inherent to each specific target model.

ViT-based Meta Model. For segmantic segmentation, we use the ViT-B_32 and ViT-B_16 models [12] as our backbones for different levels of granularity. The former splits the entire input image into 225 non-overlapping regions, each with a size of 32x32 pixels, while the latter divides it into 900 non-overlapping 16x16-pixel regions, resulting in a more fine-grained region level annotation. For object detection, we only use the ViT-B_32 as our backbone and each patch corresponds to one object query from the target model. At the beginning of training, we initialize our ViT-based meta-model with pre-trained weights from ImageNet [11]. For the initial weights of the region loss estimation head, we randomly sample from a uniform distribution [20]. All parameters are set to be trainable during the training process.

Evalution Metric. We compute the absolute error rate (ER) to the true risk of the entire test dataset as the measurement of AT, which is $ER = |\hat{R} - \hat{R}_{test}|/\hat{R}_{test} \times 100\%$, where $\hat{R}$ is the estimated risk and $\hat{R}_{test}$ is the ground truth risk on the entire labeled test dataset as shown in Eq. 1. To assess the robustness and reliability of all methods, we report the mean and standard deviation of multiple runs with different random seeds for all experiments.

Baseline. To validate the efficiency of our MetaAT approach, we conduct the comparison with three baseline methods which are introduced in Sections 2 and 3: Random Sampling (RS), Active Testing Surrogate (ATS) [24] and Active Surrogate Estimator (ASE) [23]. The original ATS and ASE methods were designed for whole-image annotation. For region-level tasks, we adapted these by dividing the images into smaller, equal sections. Since ATS and ASE can't handle the classification and regression losses in object detection simultaneously, we only compare our MetaAT method with RS in this task.

4.2 Semantic Segmentation

We first present the results of our MetaAT and three baseline methods for the semantic segmentation task, applied to four target models and four datasets, considering both image and region level annotation settings. For region level, we explore two patch sizes: 32x32 and 16x16 pixels, resulting in 225 and 900 patches per single image, named **region-32** and **region-16** for simplicity.

Performance with Respect to Annotation Budget. Figure 4 presents the main results of the average Error Rate (ER) calculated across all four target models. The budget range for each annotation level was chosen to keep the Error Rate (ER) under 10% for most methods at the highest budget point. Specifically, we use annotation budget ranges of $\leq 5\%, \leq 0.2\%$, and $\leq 0.08\%$ for image, region-32, and region-16 level experiments, respectively. Overall, our MetaAT consistently outperforms other competitors in risk estimation under all settings. Notably, it achieves a reduction of up to 87% in ER compared to the suboptimal baseline, specifically on CityScapes at the region-16 level experiment.

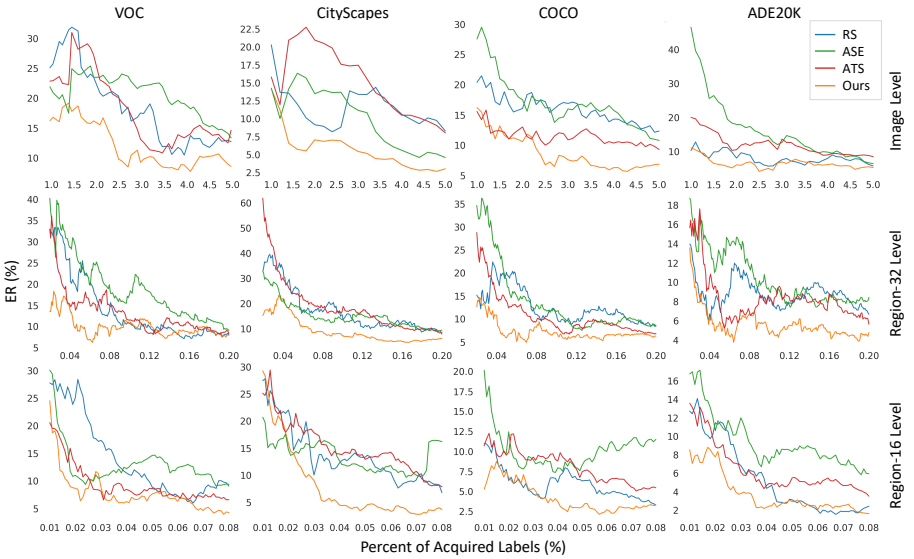

Fig. 4: Performance comparison between MetaAT and three competitors with respect to various annotation budgets for the semantic segmentation task. The results represent the average Error Rate (ER) computed across four distinct target models. In every scenario, our MetaAT surpasses the performance of all other competitors.

Further analysis shows that region level annotation is much more efficient than image level annotation. For instance, in the Pascal VOC dataset, achieving an average ER below 10% requires at least 2.5% of ground truth labels for image level annotation. However, for region-32 and region-16 level annotations, less than 0.05% and 0.02% of ground truth labels are needed to attain similar performance. This indicates a saving of about 99% in annotation costs.

Numerical Comparison of Performance. To facilitate direct quantitative comparison, Table 1 presents the average ER across the annotation budget range shown in Figure 4, equivalent to calculating the area under the ER curve. We also report the corresponding standard deviation from multiple runs with random seeds. It is evident that our approach achieves the best performance among all other methods in most settings, in terms of both ER and variance. Specifically, at the image level, region-32 level, and region-16 level, our MetaAT achieves a reduction of 39%, 37%, and 39% in ER, and 33%, 40%, and 59% in standard deviation, respectively, compared to the suboptimal approach.

Among the few scenarios where our approach is not the optimal choice, there is only one setting in which RS has a lower mean ER than us (ADE20K dataset with FCN target model). However, the difference (7.85 ± 3.81 vs 8.08 ± 2.09) is not statistically significant, with RS exhibiting almost double the variance. It's noteworthy that while ATS and ASE may occasionally outperform our method, their advantage largely stems from the iterative training of additional deep en-

Image Level (Budgets $\leq 5\%$)

Dataset	Model	RS	ASE	ATS	Ours	Dataset	Model	RS	ASE	ATS	Ours
VOC	PSPNet	16.48 ± 7.87	19.30 ± 5.46	16.42 ± 3.22	**10.57 ± 2.57**	CityScapes	PSPNet	8.09 ± 2.97	**4.39 ± 1.77**	12.39 ± 5.05	4.88 ± 0.92
	UNet	11.60 ± 3.74	11.62 ± 2.67	7.60 ± 2.95	**6.21 ± 0.79**		UNet	24.59 ± 6.47	7.77 ± 1.86	16.17 ± 10.39	**6.89 ± 2.75**
	SEGNet	25.39 ± 5.31	27.20 ± 13.47	24.48 ± 16.01	**17.52 ± 2.09**		SEGNet	6.96 ± 0.42	20.91 ± 9.54	16.97 ± 10.37	**5.58 ± 1.97**
	FCN	20.04 ± 6.49	24.08 ± 18.63	23.84 ± 11.11	**13.70 ± 4.46**		FCN	6.48 ± 0.72	7.05 ± 1.64	15.10 ± 13.03	**5.35 ± 2.06**
COCO	PSPNet	22.67 ± 3.50	**10.29 ± 1.01**	22.69 ± 7.47	14.26 ± 6.17	ADE20K	PSPNet	7.64 ± 1.59	9.45 ± 1.24	17.24 ± 6.79	**5.85 ± 2.48**
	UNet	8.55 ± 1.00	5.78 ± 3.41	5.49 ± 1.45	**5.26 ± 1.58**		UNet	7.33 ± 1.68	7.45 ± 2.06	5.44 ± 2.40	**3.68 ± 0.70**
	SEGNet	14.12 ± 0.24	34.49 ± 15.50	5.92 ± 2.01	**5.92 ± 0.75**		SEGNet	9.37 ± 1.44	24.21 ± 20.79	10.00 ± 7.21	**8.18 ± 0.49**
	FCN	18.54 ± 3.07	17.58 ± 8.21	11.77 ± 2.06	**9.20 ± 2.57**		FCN	**7.85 ± 3.81**	23.63 ± 18.99	15.14 ± 7.93	8.08 ± 2.09
Average		RS	13.48 ± 3.89	ASE	15.95 ± 10.56			ATS	14.17 ± 8.07	Ours	**8.19 ± 2.59**

Region-32 Level (Budgets $\leq 0.2\%$)

Dataset	Model	RS	ASE	ATS	Ours	Dataset	Model	RS	ASE	ATS	Ours
VOC	PSPNet	18.42 ± 6.75	21.51 ± 19.13	20.18 ± 5.52	**11.72 ± 1.52**	CityScapes	PSPNet	15.40 ± 5.67	9.98 ± 3.10	13.90 ± 1.61	**9.89 ± 2.24**
	UNet	6.00 ± 3.69	**5.02 ± 0.92**	5.13 ± 1.41	5.36 ± 2.37		UNet	19.48 ± 12.18	11.74 ± 8.19	18.40 ± 13.96	**8.70 ± 1.87**
	SEGNet	19.75 ± 10.24	32.34 ± 7.59	18.00 ± 7.69	**13.14 ± 5.14**		SEGNet	17.93 ± 4.16	26.82 ± 7.75	11.98 ± 2.90	**7.74 ± 3.85**
	FCN	11.07 ± 4.78	16.24 ± 3.07	**9.42 ± 2.43**	9.69 ± 1.01		FCN	16.20 ± 4.48	13.05 ± 9.27	30.49 ± 31.03	**10.19 ± 3.22**
COCO	PSPNet	16.82 ± 5.11	12.56 ± 1.11	10.24 ± 3.97	**8.74 ± 1.52**	ADE20K	PSPNet	11.72 ± 1.27	10.62 ± 4.30	16.48 ± 7.78	**6.20 ± 2.40**
	UNet	7.79 ± 2.16	**3.11 ± 0.82**	4.67 ± 1.20	3.74 ± 0.70		UNet	4.26 ± 0.78	4.74 ± 1.60	**3.02 ± 1.20**	4.27 ± 3.64
	SEGNet	14.00 ± 2.82	26.79 ± 11.29	7.24 ± 2.17	**5.67 ± 3.87**		SEGNet	11.75 ± 1.55	18.67 ± 8.64	**5.64 ± 0.90**	6.60 ± 3.64
	FCN	11.76 ± 2.56	12.32 ± 2.33	20.86 ± 13.24	**10.74 ± 6.60**		FCN	7.32 ± 0.98	7.69 ± 3.26	8.19 ± 3.66	**5.35 ± 1.35**
Average		RS	13.10 ± 5.34	ASE	14.57 ± 7.49			ATS	12.74 ± 9.81	Ours	**7.98 ± 3.21**

Region-16 Level (Budgets $\leq 0.08\%$)

Dataset	Model	RS	ASE	ATS	Ours	Dataset	Model	RS	ASE	ATS	Ours
VOC	PSPNet	19.89 ± 12.77	9.36 ± 4.81	9.63 ± 4.36	**8.12 ± 1.84**	CityScapes	PSPNet	14.24 ± 3.59	14.71 ± 6.70	20.54 ± 11.59	**8.88 ± 1.97**
	UNet	4.96 ± 0.56	4.12 ± 0.82	**3.70 ± 0.80**	4.27 ± 1.40		UNet	14.22 ± 3.52	**5.49 ± 0.48**	16.06 ± 10.64	7.94 ± 3.08
	SEGNet	19.09 ± 6.50	24.43 ± 2.06	9.40 ± 3.09	**9.13 ± 4.78**		SEGNet	12.97 ± 4.28	27.35 ± 11.10	10.76 ± 3.66	**9.02 ± 2.65**
	FCN	12.17 ± 6.32	13.77 ± 1.26	15.00 ± 5.09	**9.65 ± 2.19**		FCN	13.01 ± 6.07	6.24 ± 1.24	13.07 ± 5.45	**5.71 ± 1.88**
COCO	PSPNet	6.95 ± 1.29	11.62 ± 2.22	8.40 ± 1.40	**3.55 ± 1.16**	ADE20K	PSPNet	5.84 ± 4.96	12.58 ± 4.92	8.18 ± 1.83	**3.05 ± 0.90**
	UNet	5.28 ± 1.60	**4.22 ± 3.84**	4.78 ± 2.45	4.76 ± 1.88		UNet	3.68 ± 1.79	3.09 ± 0.85	**2.37 ± 1.13**	3.05 ± 0.71
	SEGNet	4.47 ± 1.69	12.26 ± 2.02	**3.59 ± 0.20**	3.97 ± 1.13		SEGNet	6.33 ± 4.31	8.74 ± 2.41	5.30 ± 1.17	**4.81 ± 0.13**
	FCN	7.09 ± 1.35	12.52 ± 6.75	15.69 ± 9.36	**4.80 ± 0.67**		FCN	5.58 ± 3.10	13.20 ± 9.50	9.62 ± 3.43	**4.10 ± 1.20**
Average		RS	9.74 ± 4.94	ASE	11.48 ± 4.94			ATS	9.76 ± 5.36	Ours	**5.93 ± 2.04**

Table 1: Performance comparison between MetaAT and three competitors for the semantic segmentation task. We show the mean of ER and its standard deviation among multiple runs with random seeds. The best results are highlighted in **bold**. Our MetaAT outperforms others across various scenarios, demonstrating the smallest average ERs and variances.

semble models, leading to considerable computational costs. On average, ASE and ATS require 4.6 times more training time compared to MetaAT. Detailed comparisons of training times are available in the supplementary materials.

4.3 Object Detection

We have extended the application of our approach to the object detection task. Unlike in the segmentation task, where non-overlapping patches are used as input for the meta model, we utilize an array of queries [46] from the object detection model. These queries originally served as the features used to determine the classification and localization of potential objects within various regions of the image. The concept of region-level annotation in object detection is similar to segmentation. The objective is to identify highly informative regions and label the corresponding objects, thereby achieving efficient estimation of the test risk.

As mentioned before, it is nontrivial to adapt ASE and ATS for the object detection task. Consequently, we only compare the performance of our MetaAT with RS, applied to three target models and the COCO Detection dataset [28], considering both image-level and region-level annotation settings.

Performance with Respect to Annotation Budget. Figure 5 presents ER results calculated across three target models. We use annotation budget ranges $\leq 4\%$ and $\leq 1\%$ for image and region level experiments, respectively. MetaAT con-

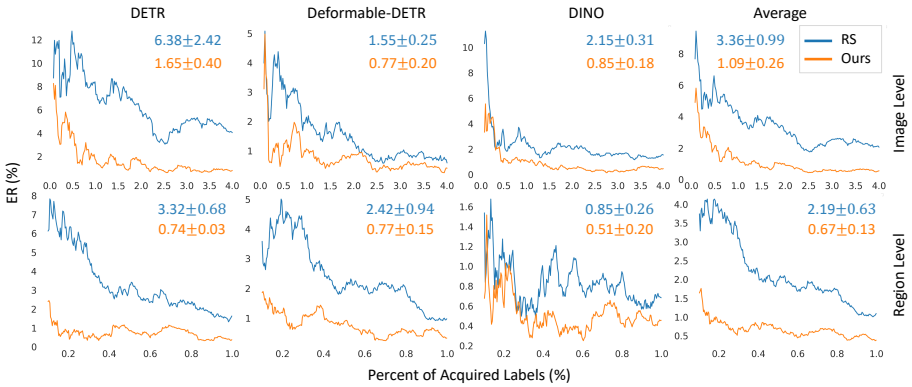

Fig. 5: Performance comparison between MetaAT and RS for evaluating object detection model. The values in the upper right corner are the mean of ER and its standard deviation. Here, ASE and ATS cannot be directly adapted to object detection as the risk of object detection involves both classification and regression. We can observe that our MetaAT outperforms the RS on both image and region level settings.

sistently outperforms RS in risk estimation under all settings, achieving up to a 96% ER reduction, notably in the DINO image-level experiment.

Numerical Comparison of Performance. The values in the upper right corner of Figure 5 are the average ER across the annotation budget range and its corresponding standard deviation. MetaAT always outperforms RS on bias and variance. On average, it archives a reduction of 68% and 69% in ER, and 74% and 80% in standard deviation, at the image level and region level.

4.4 Qualitative Analysis

Accurate loss estimation is crucial for identifying the most informative instances or regions for risk assessment. To provide a clear understanding of how MetaAT functions, we present qualitative results related to loss estimation. In Figure 6, we present a visualization of the estimated loss maps at the region level from MetaAT, ATS, and ASE. Note that all the visualizations presented here share the same color bar scale to ensure comparability and consistency in interpretation.

It is clear that our MetaAT consistently provides estimates closely aligned with ground-truth losses and outperforms other approaches in the given examples. It is worth mentioning that when facing some challenging cases (bottom left), ASE makes a reverse prediction that the plate incurs more loss than overlapping food. This is because ASE makes estimations based on epistemic uncertainty, where the unusual background can also lead to high uncertainty values. On the other hand, ATS has the worst performance as it cannot make distinguishable estimations among different regions. One potential reason for this is that the deep ensemble models used in ATS lack diversity, thus reaching a consensus on most regions and failing to identify the informative ones.

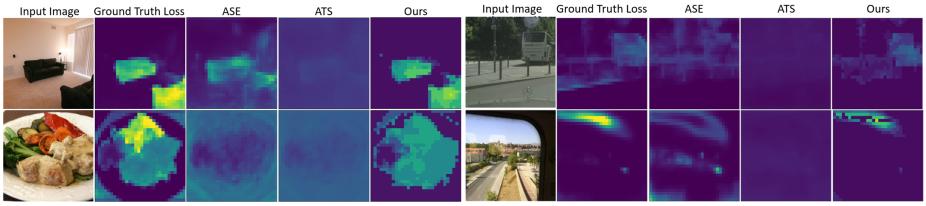

Fig. 6: Examples of estimated loss maps obtained using ASE, ATS, and MetaAT.

4.5 Ablation Study

Model	Error Rate	
	Segmentation	Detection
MetaAT	**6.21 ± 0.79**	**1.52 ± 0.42**
MetaAT w/o $\mathcal{L}_{region}$	10.95 ± 0.42	2.35 ± 0.92
MetaAT w/o binning	8.02 ± 1.68	2.12 ± 0.60
ResNet	7.69 ± 1.06	2.40 ± 1.06
MLP	10.73 ± 7.03	3.21 ± 2.06

Table 2: Ablation study for loss functions and model architectures.

We carried out an extensive ablation study to evaluate the individual impacts of different settings in our MetaAT. For semantic segmentation, we used UNet [33] and Pascal VOC [14]; for object detection, we used DETR [7] and COCO Detection [28]. Ablation studies for both the **loss function** and **model architecture** were limited to the image level, as alternate architectures cannot inherently handle region level experiments. We then broadened our study to include **inputs** ablation for semantic segmentation, focusing on the more challenging region-16 level experiment.

Loss Function. We examined the impact of key elements in our loss function. Initially, removing the region level loss from our meta model significantly worsens performance, as shown in Table 2, almost doubling the ER. Applied as a form of regularization, it encourages each patch to focus on its specific region, thereby enabling more accurate region level predictions. Consequently, the meta model can achieve a better overall representation of the entire image, leading to improved image level estimation and resulting in a lower ER. Additionally, the equal-width binning technique, transforming regression to classification, also shows improved results in Table 2. It helps the model manage the uneven distribution of ground truth data, as discussed in Subsection 3.3.

Model	Error Rate
MetaAT	**4.27 ± 1.40**
MetaAT w/o input image	4.54 ± 1.69
MetaAT w/o model output	5.38 ± 2.26
MetaAT w/o output entropy	10.36 ± 4.98

Table 3: Ablation study on different inputs for the semantic segmentation task.

Model Architecture. Table 2 compares various meta model architectures, such as ViT, ResNet [21], and MLP [19]. The results show that ViT-based meta model outperforms the rest, mainly due to its effectiveness in handling long-range dependencies. This feature enhances the meta model's ability to connect small but significant regions and discern complex patterns in images, resulting in a more precise estimation of test errors. A qualitative analysis of this is available in the supplementary materials.

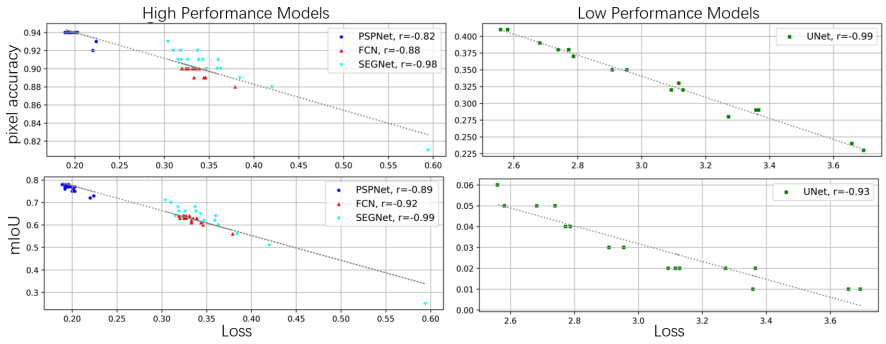

Fig. 7: Pearson correlations between loss and evaluation metrics (pixel accuracy, mIoU) for different models on the Pascal VOC dataset.

Inputs. Table 3 shows results from using various inputs in our method at the region-16 level for the semantic segmentation task. It is clear that the output entropy emerges as a crucial factor because it provides an uncertainty estimation, highly correlated with the test loss [8]. Conversely, the original image contributes minimally, as most areas, such as the background, are easily classified and thus offer redundant information for predicting the test loss.

4.6 Discussions and Limitations

We estimate the expected loss of model predictions on the testing dataset as the model's risk for two reasons: firstly, it is a standard benchmark in AT, as demonstrated in the ATS and ASE studies. Secondly, we found that loss strongly correlates with other evaluation metrics. For instance, in our analysis of different models on the Pascal VOC dataset, we found a high correlation between loss and metrics like pixel accuracy and mIoU, regardless of the models' performance levels, as illustrated in Figure 7. This suggests that loss alone is a reliable indicator of model performance. However, we have observed that the direct estimation of mIoU and mAP is challenging and remains an unsolved problem in AT. This is because these metrics cannot be computed by simply averaging the metric for each individual instance, thereby circumventing the need for Eq 4. This presents an opportunity for future research.

5 Conclusions

We introduce the MetaAT: an active testing method for label-efficient valuation of dense recognition tasks. By leveraging a meta model and the subsample risk estimator, it efficiently estimates model performance on unlabeled datasets with minimal annotation. Demonstrating superior performance across diverse benchmarks, MetaAT sets a new standard for label-efficient evaluation in dense recognition tasks, marking a significant leap in active testing methodologies.

References

1. Abe, T., Buchanan, E.K., Pleiss, G., Zemel, R., Cunningham, J.P.: Deep ensembles work, but are they necessary? arXiv preprint arXiv:2202.06985 (2022)
2. Atlas, L., Cohn, D., Ladner, R.: Training connectionist networks with queries and selective sampling. Advances in neural information processing systems **2** (1989)
3. Badrinarayanan, V., Kendall, A., SegNet, R.C.: A deep convolutional encoder-decoder architecture for image segmentation. arXiv preprint arXiv:1511.00561 **5** (2015)
4. Beluch, W.H., Genewein, T., Nürnberger, A., Köhler, J.M.: The power of ensembles for active learning in image classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9368–9377 (2018)
5. Bilgic, M., Getoor, L.: Link-based active learning. In: NIPS Workshop on Analyzing Networks and Learning with Graphs. vol. 4, p. 9 (2009)
6. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1209–1218 (2018)
7. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
8. Chan, R., Rottmann, M., Gottschalk, H.: Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation. In: Proceedings of the ieee/cvf international conference on computer vision. pp. 5128–5137 (2021)
9. Chi, W., Ma, L., Wu, J., Chen, M., Lu, W., Gu, X.: Deep learning-based medical image segmentation with limited labels. Physics in Medicine & Biology **65**(23), 235001 (2020)
10. Cordts, M., Omran, M., Ramos, S., Scharwächter, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset. In: CVPR Workshop on the Future of Datasets in Vision. vol. 2. sn (2015)
11. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
13. Elezi, I., Yu, Z., Anandkumar, A., Leal-Taixe, L., Alvarez, J.M.: Not all labels are equal: Rationalizing the labeling costs for training object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14492–14501 (2022)
14. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International journal of computer vision **111**, 98–136 (2015)
15. Farquhar, S., Gal, Y., Rainforth, T.: On statistical bias in active learning: How and when to fix it. arXiv preprint arXiv:2101.11665 (2021)
16. Guo, Y., Li, Y., Wang, L., Rosing, T.: Adafilter: Adaptive filter fine-tuning for deep transfer learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 4060–4066 (2020)
17. Hammersley, J., Morton, K.: A new monte carlo technique: antithetic variates. In: Mathematical proceedings of the Cambridge philosophical society. vol. 52, pp. 449–475. Cambridge University Press (1956)

18. Han, J., Pei, J., Tong, H.: Data mining: concepts and techniques. Morgan kaufmann (2022)
19. Haykin, S.: Neural networks: a comprehensive foundation. Prentice Hall PTR (1994)
20. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
21. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
22. Kim, S., Bae, S., Song, H., Yun, S.Y.: Re-thinking federated active learning based on inter-class diversity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3944–3953 (2023)
23. Kossen, J., Farquhar, S., Gal, Y., Rainforth, T.: Active surrogate estimators: An active learning approach to label-efficient model evaluation. NIPS **35**, 24557–24570 (2022)
24. Kossen, J., Farquhar, S., Gal, Y., Rainforth, T.: Active testing: Sample-efficient model evaluation. In: ICML. pp. 5753–5763. PMLR (2021)
25. Li, C., Kothawade, S., Chen, F., Iyer, R.: Platinum: Semi-supervised model agnostic meta-learning using submodular mutual information. In: International Conference on Machine Learning. pp. 12826–12842. PMLR (2022)
26. Li, Y., Han, H., Shan, S., Chen, X.: Disc: Learning from noisy labels via dynamic instance-specific selection and correction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24070–24079 (2023)
27. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
28. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: 2014 European Conference on Computer Vision. pp. 740–755. Springer (2014)
29. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
30. Lyu, M., Zhou, J., Chen, H., Huang, Y., Yu, D., Li, Y., Guo, Y., Guo, Y., Xiang, L., Ding, G.: Box-level active detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23766–23775 (2023)
31. Owen, A.B.: Monte Carlo theory, methods and examples. `https://artowen.su.domains/mc/` (2013)
32. Parvaneh, A., Abbasnejad, E., Teney, D., Haffari, G.R., Van Den Hengel, A., Shi, J.Q.: Active learning by feature mixing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12237–12246 (2022)
33. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. pp. 234–241. Springer (2015)
34. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. In: International Conference on Learning Representations (2018), `https://openreview.net/forum?id=H1aIuk-RW`
35. Shin, G., Xie, W., Albanie, S.: All you need are a few pixels: semantic segmentation with pixelpick. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1687–1697 (2021)

36. Su, S., Chen, N., Juefei-Xu, F., Feng, C., Miao, F.: α-ssc: Uncertainty-aware camera-based 3d semantic scene completion. arXiv preprint arXiv:2406.11021 (2024)
37. Su, S., Han, S., Li, Y., Zhang, Z., Feng, C., Ding, C., Miao, F.: Collaborative multi-object tracking with conformal uncertainty propagation. IEEE Robotics and Automation Letters (2024)
38. Su, S., Li, Y., He, S., Han, S., Feng, C., Ding, C., Miao, F.: Uncertainty quantification of collaborative detection for self-driving. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 5588–5594. IEEE (2023)
39. Trevor, H., Robert, T., Jerome, F.: The elements of statistical learning: data mining, inference, and prediction. Spinger (2009)
40. Xia, Y., Zhang, J., Jiang, T., Gong, Z., Yao, W., Feng, L.: Hatchensemble: an efficient and practical uncertainty quantification method for deep neural networks. Complex & Intelligent Systems **7**, 2855–2869 (2021)
41. Yang, Y., Zha, K., Chen, Y., Wang, H., Katabi, D.: Delving into deep imbalanced regression. In: International Conference on Machine Learning. pp. 11842–11851. PMLR (2021)
42. Yoo, D., Kweon, I.S.: Learning loss for active learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 93–102 (2019)
43. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605 (2022)
44. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2881–2890 (2017)
45. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 633–641 (2017)
46. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. In: International Conference on Learning Representations (2021)