Multi-Task Consistency for Active Learning

Aral Hekimoglu
Technical University Munich
aral.hekimoglu@tum.de

Philipp Friedrich
BMW Group
philipp.pf.friedrich@bmw.de

Walter Zimmer
Technical University Munich
zimmer@in.tum.de

Michael Schmidt
BMW Group
michael.se.schmidt@bmw.de

Alvaro Marcos-Ramiro
BMW Group
alvaro.marcos-ramiro@bmw.de

Alois Knoll
Technical University Munich
knoll@in.tum.de

Abstract

Learning-based solutions for vision tasks require a large amount of labeled training data to ensure their performance and reliability. In single-task vision-based settings, inconsistency-based active learning has proven to be effective in selecting informative samples for annotation. However, there is a lack of research exploiting the inconsistency between multiple tasks in multi-task networks. To address this gap, we propose a novel multi-task active learning strategy for two coupled vision tasks: object detection and semantic segmentation. Our approach leverages the inconsistency between them to identify informative samples across both tasks. We propose three constraints that specify how the tasks are coupled and introduce a method for determining the pixels belonging to the object detected by a bounding box, to later quantify the constraints as inconsistency scores. To evaluate the effectiveness of our approach, we establish multiple baselines for multi-task active learning and introduce a new metric, mean Detection Segmentation Quality (mDSQ), tailored for the multi-task active learning comparison that addresses the performance of both tasks. We conduct extensive experiments on the nuImages and A9 datasets, demonstrating that our approach outperforms existing state-of-the-art methods by up to 3.4% mDSQ on nuImages. Our approach achieves 95% of the fully-trained performance using only 67% of the available data, corresponding to 20% fewer labels compared to random selection and 5% fewer labels compared to state-of-the-art selection strategy. The code is available at https://github.com/aralhekimoglu/BoxMask.

1. Introduction

Object localization and classification are critical for planning and executing safe and comfortable autonomous driving. Recent deep learning methods have demonstrated state-of-the-art (SOTA) performance on 2D object detection [13, 33, 44] and semantic segmentation [7, 48] tasks. However, achieving high accuracy in these tasks comes at a high computational cost when handled separately, making them unsuitable to be used together for real-time autonomous driving. To address this challenge, multi-task learning has emerged as a promising solution. By sharing computations between related tasks, multi-task learning can achieve high accuracy while meeting real-time requirements. Recent publications showed that networks that predict both 2D object detection, and pixel-wise semantic segmentation perform better on both tasks compared to the single-task trained networks [14,15]. In this paper, we focus on the problem of multi-task active learning in autonomous driving, aiming to maximize performance across multiple tasks while minimizing the need for large amounts of labeled training data.

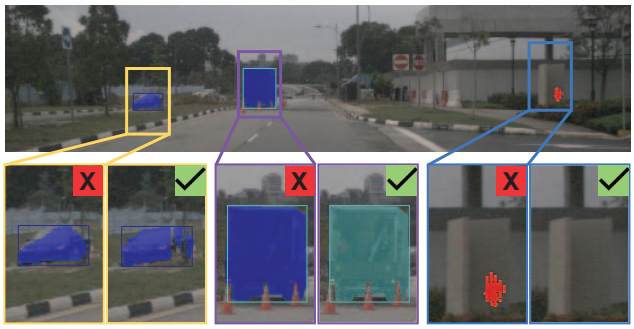


Figure 1: Consistency constraints between object detection and semantic segmentation. (a) Segmentation mask covers all pixels of a detected object. (b) A detected object and the segmentation mask that covers it share the same predicted class distribution. (c) No pixels outside of the detected boxes are segmented with an object class.

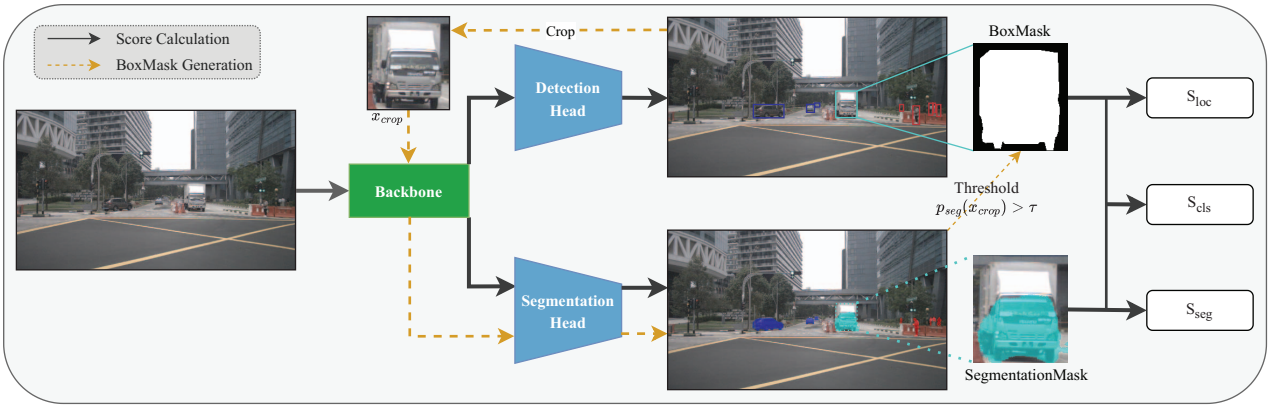


Figure 2: The proposed inconsistency-based selection strategy. Detection boxes and the segmentation mask are obtained from a multi-task network architecture consisting of a backbone and two task heads. The BoxMask is generated by cropping the region around the detected box, passing it through the backbone and segmentation head, and applying a threshold τ to the class probability corresponding to the class of the detected box. Our proposed strategy focuses on three inconsistency scores by using the BoxMask and segmentation mask.

Active learning (AL) [9, 16] is a technique for selecting the most informative samples for training a machine learning model under labeling budget constraints. In a typical single-task active learning loop, the model's predictions on the remaining unlabeled data are used to identify the samples that would be most beneficial for further training. In vision-based single-task settings, AL has been used to improve performance in tasks such as image classification [3, 45], object detection [8, 12, 23, 30, 37, 41], and semantic segmentation [20, 24]. One effective approach is inconsistency-based selection, which identifies samples for annotation by measuring the inconsistency of the model's predictions across different augmentations of the input data. For instance, CALD [46] explores various augmentations for object detection, and EquAL [20] measures inconsistency between an image and its flipped version for semantic segmentation. However, to our knowledge, no existing work has explored the use of inconsistency between multiple tasks in the context of multi-task active learning.

Our novel strategy incorporates the concept of inconsistency-based selection from active learning and applies it to multi-task learning by leveraging the inconsistency between two coupled vision tasks, namely 2D object detection and semantic segmentation. Our approach quantifies the inconsistency between these two tasks to identify informative samples across both of them to maximize the performance while minimizing the amount of labeled data needed for training.

To measure the inconsistency between 2D object detection and semantic segmentation, we define three constraints that specify how the two tasks are coupled together. The first constraint requires that the segmentation mask covers all pixels of the detected objects (Fig. 1a). The second constraint requires that a detected object and the segmentation mask that covers it share the same predicted class distribution (Fig. 1b). The third constraint requires that no pixels outside the detected boxes are segmented with an object-class (Fig. 1c). To map the object detection predictions to a similar pixel-wise output as the semantic segmentation predictions, we define BoxMask to identify the pixels belonging to the object detected by a bounding box. BoxMask enables us to quantify the three constraints by measuring the overlap between the object detection and semantic segmentation predictions. Based on these constraints, we propose three scores utilizing the BoxMask that quantify the inconsistency between the two coupled tasks.

Our main contributions are the following:

- A novel multi-task active learning strategy that effectively leverages the inconsistency between 2D object detection and semantic segmentation to improve performance on both tasks and reduce the amount of labeled data needed for training.
- A novel method for identifying the pixels belonging to a detected object (*BoxMask*) and using it to quantify the constraints between two tasks into selection scores.
- A comprehensive qualitative and quantitative comparison of the proposed approach and multi-task active learning baselines against SOTA baselines that are outperformed by up to 3.4% mDSQ and 5% in data savings rate.

2. Related work

2.1. Active learning

AL methods for object detection measure uncertainty of a detected box through either classification or localization uncertainty [2, 4, 8, 16, 26, 47]. Recent methods [16, 46] leverage inconsistency between the predictions of the network when given different augmented versions of the sample to define the robustness of a sample and select samples that are less robust. For example, Elezi et al. [16] use horizontal flipping, while Yu et al. [46] explore various augmentations to obtain multiple outputs and measure the inconsistency to define a selection score. Once the uncertainty of an object is estimated, the scores of all detections are aggregated using either the sum, the average, or the maximum of the scores, and the resulting image score is used to rank the images for annotation.

AL methods for semantic segmentation also utilize inconsistency methods to define uncertainty [20, 24, 29, 40]. For instance, Golestaneh et al. [20] apply horizontal flipping to the image and measure the inconsistency through the KL-divergence of the predictions from the original and the flipped image. The unit of data queried in AL methods for semantic segmentation varies, with methods querying either whole images [20, 43] or regions [6, 27, 29, 40, 42]. In our work, we choose to query whole images for labeling since we are also interested in obtaining object detection labels. Notably, our approach is the first to utilize inconsistency between predictions of different tasks to define multi-task uncertainty and use it for AL selection.

Learning Loss is a task-agnostic strategy proposed by Yoo *et al*. [45], using a loss prediction module. The network learns to predict the target loss for unlabeled inputs, and samples with the highest predicted loss are selected for labeling. As this approach is task-agnostic, it can be adapted for multi-task networks and serves as a relevant comparison in our work.

Diversity-based methods aim to ensure a diverse training set that covers the input space. One such method is the use of a core-set, as proposed by Sener et al. [39], where diversity is defined as the Euclidean distance between intermediate network features for each image. Another method, CDAL, proposed by Agarwal et al. [1], exploits contextual diversity with respect to the predicted classes, and has been applied to object detection and semantic segmentation. These methods are applicable to the multi-task active learning scenario and provide additional baselines for comparison.

2.2. Multi-task learning

Multi-task learning has been studied extensively, and readers are referred to a survey by Crawshaw *et al*. [10]. In multi-task architectures, the hidden layers of a backbone model are shared among different tasks, and have separate heads that predict each task [19, 31, 49]. Multi-task optimization deals with the joint objective function in multi-task settings, such as how to weigh losses of individual loss functions for different tasks [21, 28, 31, 32]. Kendall *et al*. [28] explored weighting each loss by its corresponding single-task uncertainty, using homoscedastic uncertainty for weighing the multi-task loss.

The idea of joint semantic segmentation and object detection was first investigated for shallow networks in [22, 34, 36]. These studies demonstrated that learning both tasks simultaneously can be better than learning them independently. Salscheider *et al*. [38] employed a shared backbone and heads in their work, and we adopted this approach as our multi-task network.

2.3. Multi-task active learning

To our knowledge, no prior research has investigated multi-task active learning (MTAL) for object detection and semantic segmentation. However, active learning has been successfully applied to multi-task settings in other domains, such as Natural Language Processing (NLP). Reichart and Rappoport [35] proposed alternating two single-task focused data selection strategies in each cycle, while Ikhwantri et al. [25] randomly selected a task for each cycle. Instead of alternating between two single-task scores, we propose a novel approach that generates a single score for selecting interesting samples relevant to both tasks.

3. Methodology

3.1. MTAL problem overview

The goal of this work is to tackle the AL problem of iteratively selecting samples from a large pool of unlabeled data X^U to be labeled by an oracle, to improve the performance of a multi-task object detection and semantic segmentation network. Specifically, we consider each sample (x, y_{det}, y_{seg}) as a triplet, where x is an image, y_{det} is the set of objects, and y_{seg} is the pixel-wise segmentation label belonging to one of the semantic classes C_{seg}. Detection labels y_{det} consists of bounding box coordinates (y_{box}) and corresponding categories (y_{cls}) belonging to one of the object classes C_{det}, where $C_{det} \subseteq C_{seg}$.

Our multi-task network consists of a shared backbone and two single-task heads, as shown in Fig. 2. The network predicts object boxes p_{det} consisting of (p_{box}, p_{cls}) and a segmentation mask p_{seg} for each input image x. In each AL cycle, the network is trained on the labeled data (X^L, Y^L), and a subset S of unlabeled samples is selected for labeling. In the next cycle, the selected samples are added to the labeled dataset, and the network is trained again on the updated labeled dataset.

3.2. Method overview

Object detection and semantic segmentation are two interconnected tasks shown to benefit from each other when combined [14]. The predictions from both tasks are inherently coupled, as objects detected in the former should align with the labeled regions in the latter. We propose an AL selection strategy that identifies samples where either task fails. To achieve this, we measure the inconsistency between the predictions of object detection and semantic segmentation. These inconsistent areas indicate potential points of failure for both tasks and, as such, are deemed interesting for further labeling.

To this end, we define three constraints between the tasks to formulate a selection score as illustrated in Fig. 2.

1. The segmentation mask should cover all pixels belonging to the detected object, ensuring that the entire object is accurately segmented for the given class. (Sec. 3.4)

2. The segmentation mask and the detected object should have consistent class distributions. (Sec. 3.5)

3. There should be no segmented pixel belonging to a class from the object detection outside the predicted bounding boxes. (Sec. 3.6)

3.3. BoxMask generation strategy

To quantify the constraints, we define a binary segmentation mask, BoxMask, that covers all pixels within a detected box belonging to the class of the detected object. A perfect BoxMask covers each pixel of the entire object of interest. Fig. 2 illustrates our BoxMask generation strategy. We begin by cropping a region of the image around the detected box, and then pass it through the network. Using the segmentation head of our multi-task network, we generate a segmentation mask for the cropped region $p_{seg}(x_{crop})$. BoxMask is then defined as a binary mask where the predicted class probability in the new segmentation label is above a threshold τ for the class of the detected object.

3.4. Localization consistency

Our localization-focused inconsistency score measures the alignment between a detected object and its corresponding segmentation mask. To ensure consistent detection and segmentation, the segmentation mask within the detected box should cover the entire detected object without any missing regions.

We define the localization inconsistency score as the number of pixels in the predicted segmentation mask that does not match the corresponding pixels in the BoxMask for that detection. To account for varying object sizes and scales, we normalize the score by $|BM|$, the number of pixels in the BoxMask, resulting in a scale-invariant score. Mathematically, the localization inconsistency score is given by the following equation:

$$S_{loc} = \frac{1}{|BM|} \sum_{i,j \in BM} \mathbb{I}(p_{seg}(i,j) \neq c) \tag{1}$$

where, $\mathbb{I}$ represents the indicator function, c denotes the predicted class of the detected object, and i, j represent the pixel coordinates in the BoxMask (BM).

3.5. Classification consistency

In Sec. 3.4, we addressed the localization inconsistency between the BoxMask and the segmentation mask. However, this approach treats classes that are very different, such as *Truck* and *Pedestrian*, the same as classes that are more similar, such as *Truck* and *Bus*. To account for this, we propose a classification inconsistency score that considers the predicted class probabilities during the sample score calculation.

To achieve this, we transform the predicted object class distribution p_{cls} into the same probability domain as the class probabilities predicted by semantic segmentation. Specifically, we set the probability of the classes not trained in object detection, i.e., $C_{seg} - C_{det}$, such as *Road* and *Sky*, to a negligible value to ensure that both tasks have the same number of classes, without impacting the score calculation. Since pixels within BoxMask should not contain any classes from $C_{seg} - C_{det}$, we consider this a valid assumption. The transformed probability vector, denoted as $\tilde{p}_{cls}$, contains the same classes as the segmentation task (C_{seg}).

The classification inconsistency score, S_{cls}, for a sample is calculated using the transformed object class probability distribution $\tilde{p}_{cls}$ and the class probability distribution of the segmentation p_{seg} for each pixel i, j in BoxMask as follows:

$$S_{cls} = \frac{1}{2|BM|} \sum_{i,j \in BM} KL\big(p_{seg}(i,j), \tilde{p}_{cls}\big) + KL\big(\tilde{p}_{cls}, p_{seg}(i,j)\big) \tag{2}$$

where KL represents the Kullback-Leibler divergence between the predicted probability distributions, and the resulting value is averaged to obtain a similarity measure.

3.6. Segmentation consistency

In Sec. 3.4 and 3.5, we addressed inconsistencies within the boundaries of detected boxes. However, another constraint that must be met between the two tasks is that there should be no segmentation mask outside the boundaries of the detected box for a class that also belongs to the set of classes predicted in the object detection C_{det}.

To account for the constraint that outside of the detected boxes, there should be no segmentation mask for classes

predicted in object detection, we combine all BoxMasks by taking the pixel-wise maximum and taking the inverse of the resulting binary mask to define the region falling outside of the detected objects, which we denote as BM'. The inconsistency score for the remaining segmented areas S_{seg} is calculated using Eq. (3):

$$S_{seg} = \frac{1}{|BM'|} \sum_{i,j \in BM'} \mathbb{I}(p_{seg}(i,j) \in C_{det}) \qquad (3)$$

where $\mathbb{I}$ is the indicator function. For each pixel falling within the inverted BoxMask region, we penalize the class probabilities for the classes predicted by object detection. The resulting score is normalized by the number of pixels, ensuring it is scale-invariant and shares the same range as the other inconsistency scores.

3.7. Combination of all constraints

The pseudo-code of combining the individual constraint scores into a single inconsistency score between the two tasks is given in Algorithm 1. We first calculate each detected box's BoxMask as described in Sec. 3.3. Then, we calculate the localization and classification consistency scores using Eq. (1) and Eq. (2), respectively. We then add the localization and classification scores to obtain a per-box consistency score S^{box}. Next, as explained in Sec. 3.6, we combine all BoxMasks and traverse the inverse region to search for segmentation pixels belonging to classes from the object detection and calculate the segmentation inconsistency using Eq. (3). Finally, we add this score with the maximum per-box score to estimate the inconsistency between two tasks as a single selection score.

Algorithm 1 The pseudo-code of combining the constraints

Input: p_{det}, p_{seg}
Output: score S

$BM^{comb} = \{\}$
for $box \in p_{det}$ **do**
    Obtain BM^{box} explained in Sec. 3.3
    Compute S_{loc} using Eq. (1)
    Compute S_{cls} using Eq. (2)
    $S^{box} = S_{loc} + S_{cls}$
    $BM^{comb} = pixelwise\_max(BM^{comb}, BM^{box})$
end for
$BoxMask' = inverse(BM^{comb})$
Compute S_{seg} using Eq. (3)
$S = S_{seg} + \max_{box \in p_{det}} (S^{box})$

4. Experiments

4.1. Experimental setup

Datasets. We evaluate the performance of our approach on two publicly available datasets: nuImages [5] and the A9-Dataset [11]. The nuImages dataset provides 3D and 2D sensor data collected from autonomous vehicles operating in urban settings. We use images taken from the front camera, resulting in a training set size of 13,187 images and 3,249 images for the validation set with a total of 138,569 objects. The first release of the A9 dataset offers camera and LiDAR frames from two overhead gantry bridges on the A9 autobahn near Munich, Germany. It provides annotations for object detection and semantic segmentation with 33,378 labeled image frames and a total of 672,049 3D and 2D object labels. The A9 dataset was created using the *proAnno* labeling toolbox which is based on [50].

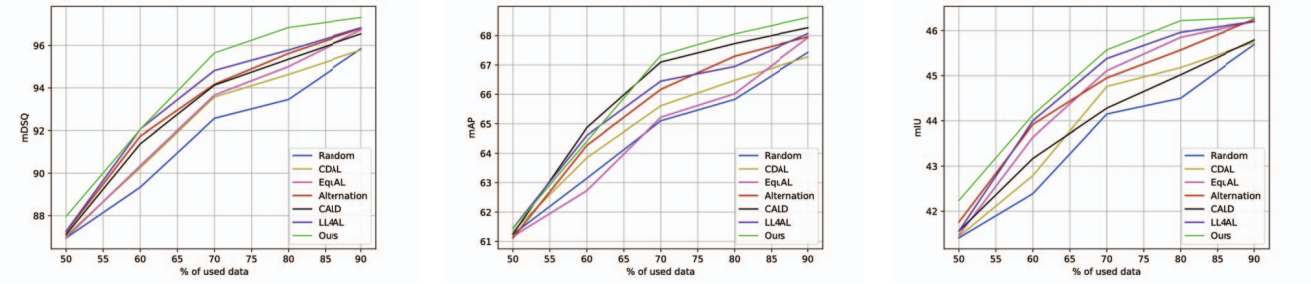
Implementation details. We employ the multi-task network architecture proposed by Salscheider *et al*. [38], augmented with the loss prediction module and the active learning framework. The hyperparameters proposed in the original work are used. The experiments are performed using a batch size of 4 and a learning rate of 0.001.

To perform active learning, we randomly divide the training set into a labeled pool of 40% and an unlabeled pool of 60%. The initial labeled pool is used to pre-train the network, and at each iteration, the top 10% of the samples with the highest scores are selected from the unlabeled pool to add to the labeled pool, based on the available annotations. We perform six active learning iterations of 30,000 steps per iteration for each dataset. We employ a continuous training strategy, where each active learning iteration is initialized with the best-performing checkpoint from the previous iteration. All experiments are conducted using two Tesla V100 GPUs and evaluated on the respective validation sets.

Evaluation metrics. The evaluation metrics for object detection and semantic segmentation are typically measured using mean Average Precision (mAP) [18] and mean Intersection-over-Union (mIoU) [17], respectively. However, a new metric that can capture the performance of both tasks is necessary to evaluate the performance of multi-task active learning methods. Therefore, we propose the mean Detection Segmentation Quality (mDSQ) metric, which normalizes mAP and mIoU by the performance of the fully-trained network and combines them, as shown in Eq. (4).

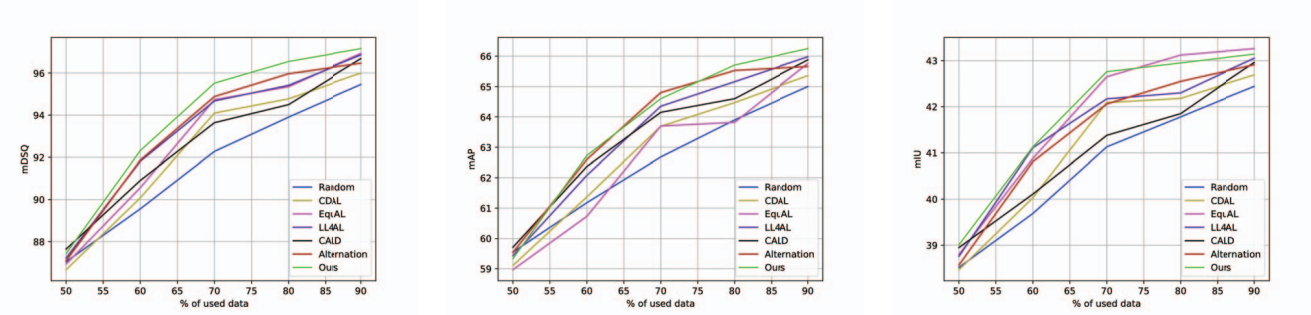
$$mDSQ = \left(\frac{mAP}{mAP_{fully}} + \frac{mIoU}{mIoU_{fully}}\right)/2 \qquad (4)$$

where mAP_{fully} and $mIoU_{fully}$ represent the performance of the network trained with 100% of data for 300,000 steps. This metric is more suitable for comparing multi-task active learning methods than the individual metrics used in



(a) Multi-task performance (mDSQ) (b) Detection performance (mAP) (c) Segmentation performance (mIoU)

Figure 3: Comparison of our proposed method with SOTA AL methods on the nuImages dataset. Lines indicate the average results over three trials. Note that all the methods start with the same network trained with 40% of labeled samples.



(a) Multi-task performance (mDSQ) (b) Detection performance (mAP) (c) Segmentation performance (mIoU)

Figure 4: Comparison of our proposed method with SOTA AL methods on the A9 dataset. Lines indicate the average results over three trials. Note that all the methods start with the same network trained with 40% of labeled samples.

each task, as it combines both metrics into a single score normalized by the fully-trained performance.

We evaluate the performance of our method using the mDSQ metric and report the mean of the metric by running three experiments with three random initial data pools. We present each experiment's numerical values and variance in the supplementary.

4.2. Baselines

To compare the effectiveness of our multi-task active learning method, we compare it against several baselines from the literature. We choose two inconsistency-based AL methods from the literature: we use *CALD* [46] as the SOTA method for object detection, and *EquAL* [20] for the semantic segmentation. We also compare against the alternating selection strategy, *Alternation*, proposed by Reichart and Rappoport [35], and alternate between two SOTA detection and segmentation selection scores *CALD* and *EquAL*. Due to its task-agnostic nature, we also compare our method against the loss prediction strategy, *LL4AL*, proposed by Yoo *et al*. [45]. We extend the network architecture by two loss prediction modules that learn to predict the loss of each task. The loss of both tasks is then summed together to form a combined loss score. We use *CDAL* [1] as our diversity-based baseline, and to mimic passive learning, we use *Random* selection, where each sample is assigned a score following a uniform distribution.

4.3. Quantitative results

nuImages. Our results on the nuImages dataset are presented in Fig. 3, which shows the mDSQ, mAP, and mIoU metrics. In the initial AL cycle, our method outperforms all the baselines by at least 0.71%. As the number of actively selected labels increases, for example, using 80% of all available data, with 50% actively labeled, our method outperforms Random by 3.39% and the second-best method, *LL4AL*, by 1.07%.

Our approach reaches 95% of the fully-trained performance using only 67% of the data, compared to 74% of *LL4AL* and 87% of random selection, corresponding to 20% of more data savings. We observe that both of the multi-task selection scores (Ours, *LL4AL*) outperform the single-task scores, their alternation and the diversity-based method. This demonstrates that a score that considers both tasks is more suitable for multi-task networks, compared to alternating between single-task scores as previously done.

Score	50%	70%	90%
S_{loc}	86.9	94.0	96.4
S_{cls}	86.5	93.5	96.6
S_{seg}	87.1	94.4	96.9
$S_{loc} + S_{cls}$	87.4	93.7	96.6
$S_{cls} + S_{seg}$	87.0	<u>95.0</u>	97.0
$S_{loc} + S_{seg}$	<u>87.7</u>	94.6	**97.5**
$S_{loc} + S_{cls} + S_{seg}$	**88.0**	**95.6**	<u>97.3</u>

Table 1: Ablation study of the contribution of each scoring constraint for each amount of used data on nuImages.

Regarding single-task performance, as shown in Fig. 3b and Fig. 3c, our method is on par with the SOTA detection algorithm *CALD* and even outperforms it as the number of actively selected samples increases. For semantic segmentation, our method outperforms the SOTA segmentation algorithm *EquAL*. These results demonstrate that both tasks benefit from inconsistency information from the other task.

A9. We present the mDSQ, mAP, and mIoU metrics for the A9 dataset in Fig. 4. Our method outperforms all the baselines for all data percentages, demonstrating the effectiveness of our data selection strategy in a larger dataset. Our selection strategy has the highest performance in the first cycle, leading to the best performance throughout the remaining cycles. We achieve 95% mDSQ using only 66% of the data, which means a 34% savings in labeling budget compared to full training.

For single-task detection performance (Fig. 4b), our method is on par with the *Alternation* baseline. For semantic segmentation (Fig. 4c), our method is on par with the SOTA segmentation algorithm *EquAL*. These results demonstrate that our multi-task approach can achieve comparable or better performance than state-of-the-art single-task AL methods for both detection and segmentation.

4.4. Qualitative results

We present qualitative results to compare the final performances obtained by following different selection strategies in Fig. 5. Compared to the *CALD* baseline, our method provides more fitting segmentation masks within each detection. For example, for the bottom vehicle in the last row, our method correctly segments the regions as *Car* instead of *Truck*, demonstrating the effectiveness of using class inconsistency. Additionally, our method produces more accurate localization for the left front wheel, which is another constraint we set in our method. Finally, the pixels outside of the boxes are worse than our method, demonstrating the effectiveness of the segmentation constraint. Overall, our qualitative comparison shows that our method outperforms the baselines in producing accurate and consistent object detection and segmentation results.

4.5. Ablation studies

Ablation on each component. We perform an ablation study to evaluate the contribution of each scoring constraint function to the overall performance of our method. The results are shown in Tab. 1. Among the single-score versions, S_{seg} has the highest performance, indicating the importance of avoiding segmented pixels outside the bounding boxes. S_{loc} and S_{cls} have comparable results, as they are both focused on different types of inconsistency, namely classification, and localization. We observe that the two combinations lead to better performance compared to their single-score counterparts. The best performance is achieved when we combine all three scores, as shown in the last row of the table. These results demonstrate the effectiveness of each constraint and the importance of combining them to achieve optimal performance.

Analysis of BoxMask accuracy and threshold. The accuracy of the BoxMask is a crucial factor in obtaining accurate constraints in our method. We conduct an experiment to evaluate the accuracy of the BoxMask and compared it with using a separate network trained solely for semantic segmentation. We calculate the mIoU between the ground-truth binary segmentation label and the predict segmentation mask. We are only interested in the areas in the ground-truth segmentation that belong to the class from the detected box and only for the area bounded by the detected boxes. Based on our results, we select a threshold value τ of 0.3 for BoxMask generation. Even though our method does not have any additional parameters, it still performs comparably to using a separate network. Therefore, we use the same segmentation head from our multi-task network for generating the BoxMask predictions.

Correlation of each consistency score with the actual error. We analyze the correlation of each consistency score with respect to the losses and each other, presented in Fig. 6. We observe that S_{seg} is better at measuring segmentation error, while S_{cls} and S_{loc} are the most effective at measuring classification and localization error in the detected boxes, respectively. Since these three losses are the main components in a joint detection and segmentation loss function, all three constraints effectively capture areas where the individual losses are high. We also observe that S_{loc} and S_{seg} have the lowest correlation across the selection scores, which explains the highest performance when combined in Tab. 1.

Single-task LL4AL and alternation. To support our hypothesis that multi-task active learning is more effective in dealing with multiple tasks simultaneously, we compare the performance of the LL4AL strategy when using two single-task scores and one multi-task score. As shown in Fig. 7, using a single multi-task score outperforms the single-task scores and their alternation. This suggests that a score considering both tasks is more suitable for multi-task networks than alternating between single-task scores.

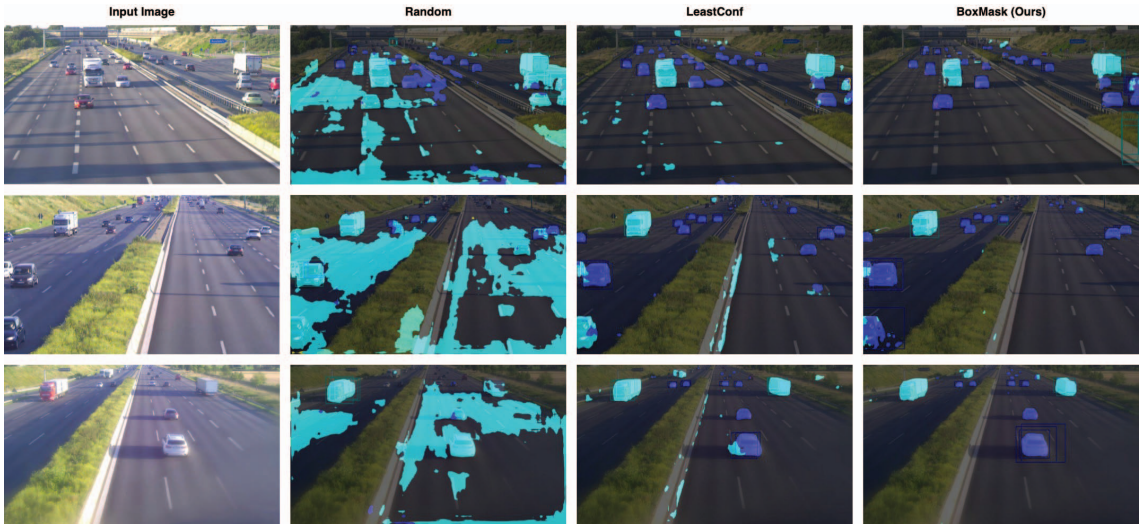


Figure 5: Qualitative comparison of the Random (second), CALD (third), and Our (fourth) sampling strategies on the A9 dataset. Light blue and dark blue correspond to the *Truck* and *Car* classes.

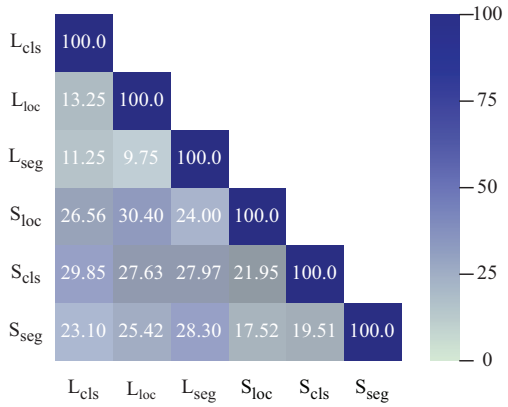


Figure 6: Correlation of each constraint and losses

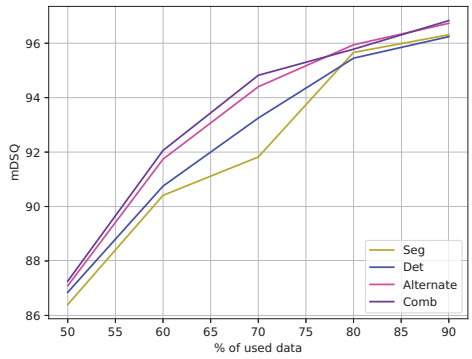


Figure 7: Comparison between multi-task LL4AL to single-task losses and their alternation. *Seg* and *Det* refer to LL4AL using only the segmentation and detection losses, respectively. All methods start with the same network trained with 40% of labeled data.

Mask Gen.	τ	Add Mem.	Accuracy
BoxMask	0.1	-	74.7
	0.3	-	<u>78.4</u>
	0.5	-	77.9
	0.7	-	73.2
HR-Net	-	6.31 GB	**80.2**

Table 2: Performance of BoxMask across different thresholds.

5. Conclusion

This study addressed the gap in research on active learning for multi-task networks in the vision domain. Our proposed selection strategy combines knowledge from the two task domains, object detection and semantic segmentation, into a single multi-task selection score. This strategy relies on three constraints between the two tasks and measures them by identifying the pixels belonging to a detected object through the BoxMask. Our experiments on two multi-task datasets demonstrate the effectiveness of our approach, as it outperforms all the baselines by 3.4% and achieves 5% fewer annotations. Future work will focus on adapting our task inconsistency-based selection strategy to other multi-task networks.

References

[1] Sharat Agarwal, Himanshu Arora, Saket Anand, and Chetan Arora. Contextual diversity for active learning. In *ECCV*, 2020. 3, 6

[2] Hamed H Aghdam, Abel Gonzalez-Garcia, Joost van de Weijer, and Antonio M López. Active learning for deep detection neural networks. In *ICCV*, 2019. 3

[3] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *CVPR*, 2018. 2

[4] Clemens-Alexander Brust, Christoph Käding, and Joachim Denzler. Active learning for deep object detection. In *VISAPP*, 2019. 3

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 5

[6] Arantxa Casanova, Pedro O Pinheiro, Negar Rostamzadeh, and Christopher J Pal. Reinforced active learning for image segmentation. In *ICLR*, 2020. 3

[7] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *CVPR*, 2020. 1

[8] Jiwoong Choi, Ismail Elezi, Hyuk-Jae Lee, Clement Farabet, and Jose M Alvarez. Active learning for deep object detection via probabilistic modeling. In *ICCV*, 2021. 2, 3

[9] Pascal Colling, Lutz Roese-Koerner, Hanno Gottschalk, and Matthias Rottmann. Metabox+: A new region based active learning method for semantic segmentation using priority maps. In *ICPRAM*, 2020. 2

[10] Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020. 3

[11] Christian Creß, Walter Zimmer, Leah Strand, Maximilian Fortkord, Siyi Dai, Venkatnarayanan Lakshminarasimhan, and Alois Knoll. A9-dataset: Multi-sensor infrastructure-based dataset for mobility research. In *IV*, 2022. 5

[12] Sai Vikas Desai, Akshay L Chandra, Wei Guo, Seishi Ninomiya, and Vineeth N. Balasubramanian. An adaptive supervision framework for active learning in object detection. In *BMVC*, 2019. 2

[13] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *ICCV*, 2019. 1

[14] Nikita Dvornik, Konstantin Shmelkov, Julien Mairal, and Cordelia Schmid. Blitznet: A real-time deep network for scene understanding. In *ICCV*, 2017. 1, 4

[15] Nikolas Ebert, Patrick Mangat, and Oliver Wasenmuller. Multitask network for joint object detection, semantic segmentation and human pose estimation in vehicle occupancy monitoring. In *IV*, 2022. 1

[16] Ismail Elezi, Zhiding Yu, Anima Anandkumar, Laura Leal-Taixe, and Jose M Alvarez. Not all labels are equal: Rationalizing the labeling costs for training object detection. In *CVPR*, 2022. 2, 3

[17] Mark Everingham, SM Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 2015. 5

[18] Di Feng, Ali Harakeh, Steven L Waslander, and Klaus Dietmayer. A review and comparative study on probabilistic object detection in autonomous driving. In *ITS*, 2021. 5

[19] Golnaz Ghiasi, Barret Zoph, Ekin D Cubuk, Quoc V Le, and Tsung-Yi Lin. Multi-task self-training for learning general representations. In *ICCV*, 2021. 3

[20] S Alireza Golestaneh and Kris M Kitani. Importance of self-consistency in active learning for semantic segmentation. In *BMVC*, 2020. 2, 3, 6

[21] Ting Gong, Tyler Lee, Cory Stephenson, Venkata Renduchintala, Suchismita Padhy, Anthony Ndirango, Gokce Keskin, and Oguz H Elibol. A comparison of loss weighting strategies for multi task learning in deep neural networks. *IEEE Access*, 2019. 3

[22] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *ECCV*, 2014. 3

[23] Aral Hekimoglu, Michael Schmidt, Alvaro Marcos-Ramiro, and Gerhard Rigoll. Efficient active learning strategies for monocular 3d object detection. In *IV*, 2022. 2

[24] Siyu Huang, Tianyang Wang, Haoyi Xiong, Jun Huan, and Dejing Dou. Semi-supervised active learning with temporal output discrepancy. In *ICCV*, 2021. 2, 3

[25] Fariz Ikhwantri, Samuel Louvan, Kemal Kurniawan, Bagas Abisena, Valdi Rachman, Alfan Farizki Wicaksono, and Rahmad Mahendra. Multi-task active learning for neural semantic role labeling on low resource conversational corpus. In *ACLW*, 2018. 3

[26] Chieh-Chi Kao, Teng-Yok Lee, Pradeep Sen, and Ming-Yu Liu. Localization-aware active learning for object detection. In *ACCV*, 2018. 3

[27] Tejaswi Kasarla, Gattigorla Nagendar, Guruprasad M Hegde, Vineeth Balasubramanian, and CV Jawahar. Region-based active learning for efficient labeling in semantic segmentation. In *WACV*, 2019. 3

[28] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *CVPR*, 2018. 3

[29] Bo Li and Tommy Sonne Alstrøm. On uncertainty estimation in active learning for image segmentation. In *ICMLW*, 2020. 3

[30] Ying Li, Binbin Fan, Weiping Zhang, Weiping Ding, and Jianwei Yin. Deep active learning for object detection. *Information Sciences*, 2021. 2

[31] Shikun Liu, Edward Johns, and Andrew J Davison. End-to-end multi-task learning with attention. In *CVPR*, 2019. 3

[32] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Multi-task deep neural networks for natural language understanding. In *ACL*, 2019. 3

[33] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *CVPR*, 2022. 1

[34] Dim P Papadopoulos, Jasper RR Uijlings, Frank Keller, and Vittorio Ferrari. Extreme clicking for efficient object annotation. In *ICCV*, 2017. 3

[35] Roi Reichart, Katrin Tomanek, Udo Hahn, and Ari Rappoport. Multi-task active learning for linguistic annotations. In *ACL*, 2008. 3, 6

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 3

[37] Soumya Roy, Asim Unmesh, and Vinay P Namboodiri. Deep active learning for object detection. In *BMVC*, 2018. 2

[38] Niels Ole Salscheider. Simultaneous object detection and semantic segmentation. In *ICPRAM*, 2020. 3, 5

[39] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR*, 2018. 3

[40] Yawar Siddiqui, Julien Valentin, and Matthias Nießner. Viewal: Active learning with viewpoint entropy for semantic segmentation. In *CVPR*, 2020. 3

[41] Fuhui Tang, Dafeng Wei, Chenhan Jiang, Hang Xu, Andi Zhang, Wei Zhang, Hongtao Lu, and Chunjing Xu. Towards dynamic and scalable active learning with neural architecture adaption for object detection. *BMVC*, 2021. 2

[42] Binhui Xie, Longhui Yuan, Shuang Li, Chi Harold Liu, and Xinjing Cheng. Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation. In *CVPR*, 2022. 3

[43] Shuai Xie, Zunlei Feng, Ying Chen, Songtao Sun, Chao Ma, and Mingli Song. Deal: Difficulty-aware active learning for semantic segmentation. In *ACCV*, 2020. 3

[44] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *ICCV*, 2021. 1

[45] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *CVPR*, 2019. 2, 3, 6

[46] Weiping Yu, Sijie Zhu, Taojiannan Yang, and Chen Chen. Consistency-based active learning for object detection. In *CVPR*, 2022. 2, 3, 6

[47] Tianning Yuan, Fang Wan, Mengying Fu, Jianzhuang Liu, Songcen Xu, Xiangyang Ji, and Qixiang Ye. Multiple instance active learning for object detection. In *CVPR*, 2021. 3

[48] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *ECCV*, 2020. 1

[49] Xiangyun Zhao, Haoxiang Li, Xiaohui Shen, Xiaodan Liang, and Ying Wu. A modulation module for multi-task learning with applications in image retrieval. In *ECCV*, 2018. 3

[50] Walter Zimmer, Akshay Rangesh, and Mohan Trivedi. 3d bat: A semi-automatic, web-based 3d annotation toolbox for full-surround, multi-modal data streams. In *IV*, 2019. 5