DEEP ACTIVE LEARNING FOR OBJECT DETECTION WITH MIXTURE DENSITY NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Active learning aims to reduce the labeling costs by selecting only samples that are informative to improve the accuracy of the network. Few existing works have addressed the problem of active learning for object detection, and most of them estimate the informativeness of an image based only on the classification head, neglecting the influence of the localization head. In this paper, we propose a novel deep active learning approach for object detection. Our approach relies on mixture density networks to provide a distribution for every output parameter of the network. Through these distributions, our approach is able to compute, separately and in a single forward pass of a single model, the epistemic and aleatoric uncertainty. In addition, we propose another efficient approach to reduce the computational cost of the mixture model. For active learning, we propose a scoring function that aggregates uncertainties from both the classification and localization outputs of the network. We demonstrate the benefits of our approach in PASCAL VOC and COCO datasets. Our mixture model based object detection outperforms the corresponding original models with accuracy improvements up to 2.82% in the strict method. In active learning, our approach outperforms the state-of-the-art methods using a single model and, yields competitive accuracy compared to methods using multiple models at a fraction of the compute cost. We empirically demonstrate that aggregating uncertainties from both tasks is a key factor for the improvement. In addition, we show that our approach scales to different object detection networks, and datasets acquired actively using our approach can successfully be transferred to different networks.

1 INTRODUCTION

Deep learning models can achieve high accuracy in object detection by leveraging a massive quantity of labeled data (Liu et al., 2016; Ren et al., 2015). While crawling a large amount of image data is a trivial task, labeling this data is an expensive and time-consuming activity. An image typically contains multiple objects, and each object requires a category and a bounding box. Therefore, devising a smart labeling strategy becomes very desirable.

Active learning aims at selecting the smallest possible training set to solve a specific task (Cohn et al., 1994). Active learning methods involve the model in the selection of what images to learn from. By doing this, they can boost the performance of the models (Beluch et al., 2018; Yoo & Kweon, 2019; Chitta et al., 2019) and reduce the labeling costs. A key component of active learning is the scoring function that, usually based on the predictive uncertainty of the model, aims at providing a single value per image representing its informativeness. The predictive uncertainty can be decomposed into Aleatoric and Epistemic uncertainty (Hora, 1996). The former refers to the notion of randomness, or the noise inherent in the observations, such as sensor noise, and can be attributed to occlusions, lack of visual features, or object distance (Kendall & Gal, 2017; Feng et al., 2018). The latter refers to the uncertainty caused by the lack of knowledge. Aleatoric uncertainty represents the non-reducible part while the epistemic one can be reduced given enough additional data (Liu et al., 2019). Modeling and distinguishing these two types of uncertainty is very important when mining data as aleatoric uncertainty is useful in large data situations and real-time applications, whereas epistemic uncertainty is relevant for safety-critical applications and in small datasets (Kendall & Gal, 2017).

Few works have addressed the problem of active learning for object detection. The first approaches are extensions of image classification methods, computing pixel-level scores in the detector's confidence branch, and then aggregating them into a frame-level score (Aghdam et al., 2019). These methods ignore the localization branch to compute an image's informativeness score. More recent approaches have shown promising results for modeling bounding box uncertainty in object detection (Choi et al., 2019; He et al., 2019). However, these methods, called *Box-uncertainty*, model uncertainty as a single probability distribution, thus failing to decouple between epistemic and aleatoric uncertainty. A common approach to decouple epistemic and aleatoric uncertainties is to use multiple object detection networks such as ensembles (Haussmann et al., 2020) or Monte Carlo (MC) sampling (Gal & Ghahramani, 2016). However, these methods become impractical as they require much higher computational cost.

In this paper, we propose a novel active learning approach for deep object detection based on the uncertainty of both the predicted object class and its bounding box. In contrast to other methods, our approach relies on a mixture density network to learn the parameters of a Gaussian mixture model (GMM) for each of the outputs of the object detector. Given the parameters of these GMMs, our approach can explicitly compute, in a single forward pass of a single model, the aleatoric and epistemic uncertainties for the classification and localization heads. We further improve the computational cost of our approach by proposing a more efficient modeling method for the classification head. To train our mixture density based object detector, we propose a loss function that serves as a regularizer for inconsistent data leading to more robust models. For active learning, our scoring function aggregates localization and classification-based uncertainties for each object in the image to obtain the final image's informativeness score. We demonstrate the benefits of our approach in two public datasets PASCAL VOC and COCO. Our mixture model based object detection outperforms the corresponding original models with accuracy improvements up to 2.82% on the strict IoU metric. When used for active learning, our approach outperforms single model based methods, and, compared to methods using multiple models, our approach yields a similar accuracy while reducing the forward time up to 92.68% compared to MC dropout. We empirically show that aggregating uncertainties from the classification and localization heads is a key factor for such better performance.

2 Related Work

Active Learning has been actively studied over the last two decades. The main idea is to choose the most informative samples for a classifier. The excellent survey (Settles, 2012) well describes the problem in the regime of low-level data.

Deep Active Learning has found an interest in the last few years with many works tackling the problem from different directions. The work of Beluch et al. (2018) trains an ensemble of neural networks and then selects the samples with the highest score defined by some acquisition function, i.e., entropy (Shannon, 2001) or mutual information (Chitta et al., 2018). Concurrent works (Gal et al., 2017; Kirsch et al., 2019) explore similar direction, but by approximating the uncertainty via MC-dropout (Gal & Ghahramani, 2016). The work of Beluch et al. (2018) compares the approaches, decisively concluding that the ensemble approach reaches higher results at the cost of more computational power. Other works have considered Bayesian (Tran et al., 2019; Sinha et al., 2019) or core-set (Sener & Savarese, 2018) approaches. Most of these methods (Lewis & Catlett, 1994; Gal & Ghahramani, 2016; Beluch et al., 2018; Sener & Savarese, 2018) have been extended in a straightforward way to the problem of object detection.

In addition to these methods, there are several methods which have been proposed specifically for object detection. The work of Aghdam et al. (2019) proposes a solution by training a network that computes dense object prediction probabilities for each unlabeled image, followed by computing pixel-scores and aggregating them into a frame-level score. A different solution was given by Kao et al. (2018) where the authors define two different scores: "localization tightness" which is the overlapping ratio between the region proposal and the final prediction; and "localization stability" that is based on the variation of predicted object locations when input images are corrupted by noise. In all cases, the images with the highest scores are chosen to be labeled. The work of Roy et al. (2018) proposes a "query by committee" paradigm to choose the set of images to be queried. Another approach is that of Desai et al. (2019) where instead of directly querying bounding box annotations (strong labels) for the most informative samples, they first query weak labels and optimize the model. Then,

using a switching condition, the required supervision level is increased. Yet another approach has been proposed in Haussmann et al. (2020), where an ensemble of object detectors provides potential bounding boxes and probabilities for each class of interest. Then, a scoring function is used to obtain a single value representing the informativeness of each unlabeled image. The work of Yoo & Kweon (2019) gives a heuristic but elegant solution, while reaching state-of-the-art results compared with other single-model methods. The authors train a network in the task of detection while learning to predict the final loss. In the sample acquisition stage, samples with the highest prediction loss are considered as the most interesting ones and are chosen to be labeled.

Mixture Density Network has been widely used for several deep learning tasks in recent years. The approach of Choi et al. (2018) focus on regression task of the steering angle and the works of He & Wang (2019); Varamesh & Tuytelaars (2020) attempt to solve a multimodal regression task. Other work of Yoo et al. (2019) focus on density estimation and another approach of Choi et al. (2020) attempt to solve the supervised learning problem with corrupted data. However, previous studies in Choi et al. (2018); He & Wang (2019); Varamesh & Tuytelaars (2020) did not consider classification task, which is an essential part of object detection, and all these previous studies did not estimate and take into account two types of uncertainty of bounding box regression and classification tasks, and also none of these works has explicitly addressed active learning for deep object detection. In the next section, we introduce our approach for estimating both the aleatoric and epistemic uncertainty in a single forward pass with a single model in the context of active learning for object detection.

3 Active Learning for Object Detection

In this section, we introduce our active learning approach that uses only a single model for estimating uncertainty with a single forward pass. As depicted in Figure 1, the key novelty of our approach is a modification in the output layers to predict a probability distribution instead of a single value for each output of the network. To this end, we propose to make use of a mixture density network where the output of the network consists of the parameters of a GMM. That is the mean μ^k, the variance Σ^k and the mixture weight π^k for the k-th component of the GMM. Given these parameters, we can estimate the aleatoric u_{al} and epistemic u_{ep} uncertainties as (Choi et al., 2018):

$$u_{al} = \sum_{k=1}^{K} \pi^k \Sigma^k, \;\; u_{ep} = \sum_{k=1}^{K} \pi^k \|\mu^k - \sum_{i=1}^{K} \pi^i \mu^i\|^2. \tag{1}$$

Below we first introduce the mixture modeling for object detection for both localization and classification and then, we describe the scoring function to be used during active learning.

3.1 Object Detection with Uncertainty Modeling

As shown in Figure 1, our network builds upon Single Shot MultiBox Detector (SSD) (Liu et al., 2016), which is widely used in active learning studies, with a VGG16 backbone (Simonyan & Zisserman, 2015) and a MultiBox module (Erhan et al., 2014). Additionally, the network combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes via the Extra layers. SSD predicts, on each location of the feature map, d default bounding boxes with different sizes and aspect ratios. For each default box, the network predicts its coordinates (via the localization head) and the class (via the confidence head). To introduce our approach, we first focus on the regression task and then, extend it to the classification side.

Localization: Instead of predicting a deterministic value for each bounding box coordinate, our algorithm outputs 3 groups of parameters for each bounding box: the mean ($\hat{\mu}_x$, $\hat{\mu}_y$, $\hat{\mu}_w$, and $\hat{\mu}_h$), the variance ($\hat{\Sigma}_x$, $\hat{\Sigma}_y$, $\hat{\Sigma}_w$, and $\hat{\Sigma}_h$), and the weights of the mixture ($\hat{\pi}_x$, $\hat{\pi}_y$, $\hat{\pi}_w$, and $\hat{\pi}_h$).

Let $\{\hat{\pi}_b^k, \hat{\mu}_b^k, \hat{\Sigma}_b^k\}_{k=1}^K$ be the bounding box outputs obtained using our approach. The parameters of a GMM with K models for each coordinate of the bounding box are obtained as follows:

$$\pi_b^k = \frac{exp(\hat{\pi}_b^k)}{\sum_{j=1}^{K} exp(\hat{\pi}_b^j)}, \quad \mu_b^k = \hat{\mu}_b^k, \quad \Sigma_b^k = \sigma(\hat{\Sigma}_b^k), \quad b \in \{x, y, w, h\}, \tag{2}$$

where π is the mixture weight for each component, μ is the predicted value for each output of the bounding box, and Σ is the variance for each coordinate representing its aleatoric uncertainty. As

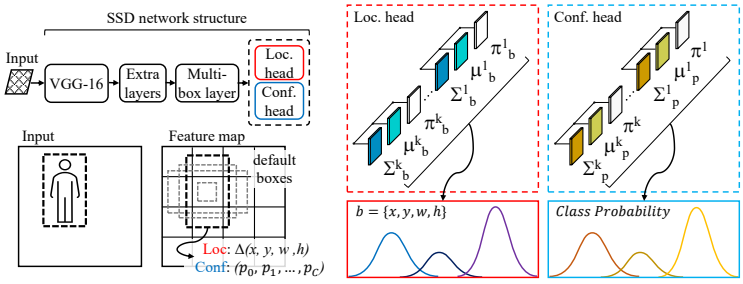


Figure 1: An overview of the proposed object detection network. The main difference with original SSD (Liu et al., 2016) is in the localization and classification branches. Instead of having deterministic outputs, our approach learns the parameters of a K-component GMM for each of the outputs.

suggested in Choi et al. (2018) we use a softmax function to keep π in probability space and use a sigmoid function to satisfy the positiveness constraint of the variance, $\Sigma_b^k >= 0$.

For training the mixture density network for localization, instead of using the smooth L1 loss (Girshick, 2015), we propose a localization loss based on the negative log-likelihood (NLL) loss to regress the parameters of the GMM to the offsets of the center (x, y), width (w), and height (h) of the default (anchor) box (d) for positive matches:

$$L_{loc}(x, l, g) = -\sum_{i \in Pos}^{N} \sum_{b \in B} \lambda_{ij} log(\sum_{k=1}^{K} \pi_b^{ik} \mathcal{N}(\hat{g}_b^j | \mu_b^{ik}, \Sigma_b^{ik}) + \varepsilon), \tag{3}$$

$$\lambda_{ij} = \begin{cases} 1, & \text{if } IoU > 0.5. \\ 0, & \text{otherwise.} \end{cases}, \quad \hat{g}_x^j = \frac{(g_x^j - d_x^i)}{d_w^i}, \quad \hat{g}_y^j = \frac{(g_y^j - d_y^i)}{d_h^i}, \quad \hat{g}_w^j = log(\frac{g_w^j}{d_w^i}), \quad \hat{g}_h^j = log(\frac{g_h^j}{d_h^i}).$$

where N is the number of positive matches, B is the offset of the bounding box coordinate, $\hat{g}_b^j$ is the ground-truth (GT) of the j-th box, and λ_{ij} is an indicator function for matching the i-th default box to the j-th GT box. In experiments, we set $\varepsilon = e^{-9}$ for the numerical stability of the log function.

Classification: We now focus on the classification branch of the object detector where we model the output of every class as a GMM, see Figure 1. Our approach estimates the mean $\hat{\mu}_p^k$ and variance $\hat{\Sigma}_p^k$ for each class, and the weights of the mixture $\hat{\pi}^k$ for each component of the GMM. We process the parameters of the GMM following Eq. 2, and obtain the class probability distribution for the j-th bounding boxes by applying Gaussian noise and variance Σ_p^i to μ_p^i (Kendall & Gal, 2017):

$$\hat{c}_p^j = \mu_p^j + \sqrt{\Sigma_p^j}\gamma, \quad \gamma \sim \mathcal{N}(0, 1) \tag{4}$$

For training, in this case, we propose a loss function that takes into account the IoU of the default bounding boxes compared to GT and hard negative mining. More precisely, we formulate the classification loss as a combination of two terms L_{cl}^{Pos} and L_{cl}^{Neg} representing the contribution of positive and negatives samples:

$$\begin{aligned} L_{cl}^{Pos}(x, c) &= -\sum_{i \in Pos}^{N} \lambda_{ij} \sum_{k=1}^{K} \pi^{ik} (\hat{c}_g^{ik} - log(\sum_{p=0}^{C} e^{\hat{c}_p^{ik}}) \\ L_{cl}^{Neg}(x, c) &= -\sum_{i \in Neg}^{M \times N} \sum_{k=1}^{K} \pi^{ik} (\hat{c}_0^{ik} - log(\sum_{p=0}^{C} e^{\hat{c}_p^{ik}}), \end{aligned} \tag{5}$$

where C is the number of classes, with 0 representing the background class, N is the number of positive matches, $\hat{c}_g^{ik}$ is the ground-truth class for the i-th match, and $M = 3$ is the ratio of hard negative mining (Liu et al., 2016). In experiments, instead of using all the negative matches, we sort them using mixture classification loss and choose top $M \times N$ as final negative matches for training.

Finally, the overall loss to train the object detector using mixture density network is defined as:

$$L(x, c, l, g) = \begin{cases} \frac{1}{N}(L_{loc}(x, l, g) + L_{cl}^{Pos}(x, c) + L_{cl}^{Neg}(x, c)), & \text{if } N > 0. \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

where N is the number of positive matches.

At inference, we can compute the coordinates of the bounding box R_b and the confidence score for each class P_i by summing the components of the mixture model as follows:

$$Localization: \ R_b = \sum_{k=1}^{K} \pi_b^k \mu_b^k, \ \ Classification: \ P_i = \sum_{k=1}^{K} \pi^k \frac{exp(\mu_i^k)}{\sum_{j=0}^{C} exp(\mu_j^k)}. \tag{7}$$

Improving parameter efficiency. Our approach to predict a probability distribution of the output values involves modifying the last layer of the network and therefore incurs in an increment in the number of parameters, especially in the classification branch. More precisely, assuming a feature map of size $K \times K$, C classes, D default boxes to be predicted, and each bounding box defined using 4 coordinates, the number of parameters in the new layer added to estimate a K-component GMM with 3 parameters is $F \times F \times D \times (4 \times 3 \times K)$ and $F \times F \times D \times (C \times 2 \times K + K)$, for the localization and classification heads respectively. We can see that the number of parameters in the classification branch is proportional to the number of classes leading to a larger computational cost.

In this section, we focus on improving the efficiency of the algorithm by reducing the number of parameters in the classification branch. To this end, we eliminate estimating the variance Σ_p to reduce the number of parameters to $F \times F \times D \times (C \times K + K)$. This variance is used in the original formulation to compute the class probability, the aleatoric uncertainty and to implicitly regularize the training process towards low uncertainty solutions. Alternatively, we can obtain class probabilities as $\hat{c}_p^j = \mu_p^j$, and use them to estimate the aleatoric uncertainty as follows (Kwon et al., 2020):

$$u_{al} = \sum_{i=1}^{K} \pi^i (diag(\hat{c}_p^i) - (\hat{c}_p^i)^{\otimes 2}), \tag{8}$$

where $diag(q)$ is a diagonal matrix with the elements of the vector q and $q^{\otimes 2} = qq^T$. In this case, u_{al} is $C \times C$ matrix where the diagonal elements represent the aleatoric uncertainty.

Finally, we modify the loss function for classification (similarly for localization) to explicitly promote solutions with low uncertainty as follows:

$$L_{cl}^{Pos}(x, c) = - \sum_{i \in Pos}^{N} \lambda_{ij} \frac{(1 - u_{al}^{loc})}{(1 + u_{al_r})^{\alpha}} \sum_{k=1}^{K} \pi^{ik} (\hat{c}_g^{ik} - log(\sum_{p=0}^{C} e^{\hat{c}_p^{ik}}), \tag{9}$$

where $u_{al}^{loc} = \frac{1}{4} \sum_{b \in B}^{4} u_{al}^b$ aggregates the aleatoric uncertainty of the coordinates of the bounding box, and α controls the strength of this constraint. In our experiments, we set $\alpha = 4$.

3.2 Scoring Function

The scoring function in active learning provides a single value per image indicating its informativeness. In our case, we estimate the informativeness of an image by aggregating all the aleatoric and epistemic uncertainty values for each parameter of each bounding box present in the image.

Specifically, let $U = \{u^{ij}\}$ be the set of uncertainties values (aleatoric or epistemic) of a group of images where u^{ij} is the uncertainty for the j-th object in the i-th image. For localization, u^{ij} is the maximum value over the 4 bounding box outputs. We first normalize these values using z-score normalization ($\tilde{u}^{ij} = (u^{ij} - \mu_U)/\sigma_U$) to compensate the fact that the values for the coordinates of the bounding box are unbounded and each image might have a different range of values. We then assign to each image the maximum uncertainty over the detected objects $u^i = \max_j \tilde{u}^{ij}$. We empirically find that taking the maximum over the coordinates and the objects performs better than by taking the average.

Using the algorithm described above we obtain four different normalized uncertainty values for each image: epistemic and aleatoric for classification and localization, $\mathbf{u} = \{u_{ep_c}^i, u_{al_c}^i, u_{ep_b}^i, u_{al_b}^i\}$ respectively. The remaining part is to aggregate these scores into a single one. We experiment with two popular approaches, such as averaging or taking the maximum, as other active learning studies (Haussmann et al., 2020). As shown in the appendix, taking maximum works better.

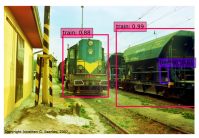
Table 1: Accuracy of different instances of our approach compared to the original SSD network. SGM and MDN refer to single and multiple Gausian models, and we apply those to localization (Loc), classification (Cl), and their combination (Loc+Cl).

Method	Head	IoU > 0.5	IoU > 0.75
SSD	–	69.29 ± 0.51	43.36 ± 1.24
SGM	Loc	70.20 ± 0.27	45.39 ± 0.23
MDN	Loc	70.09 ± 0.22	46.01 ± 0.27
SGM	Cl	69.95 ± 0.41	44.25 ± 0.26
MDN	Cl	$\mathbf{70.47} \pm 0.17$	44.47 ± 0.06
Ours	Loc + Cl	70.19 ± 0.36	46.11 ± 0.38
Ours*	Loc + Cl	70.45 ± 0.06	$\mathbf{46.18} \pm 0.26$

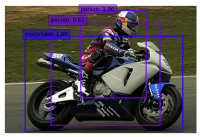
(a) **VOC07**

method	head	IoU > 0.5	IoU > 0.75
SSD	–	25.63 ± 0.40	11.93 ± 0.60
SGM	Loc	27.20 ± 0.08	12.70 ± 0.16
MDN	Loc	27.67 ± 0.12	13.53 ± 0.05
SGM	Cl	27.23 ± 0.12	12.50 ± 0.08
MDN	Cl	27.33 ± 0.09	12.67 ± 0.09
Ours	Loc + Cl	$\mathbf{27.70} \pm 0.08$	$\mathbf{13.57} \pm 0.19$
Ours*	Loc + Cl	27.33 ± 0.04	13.33 ± 0.12

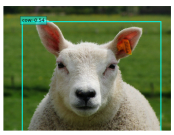
(b) **COCO**



u_{al_b}: **3.60** u_{al_c}: 0.96
u_{ep_b}: 1.06 u_{ep_c}: −0.19



u_{al_b}: 1.71 u_{al_c}: −0.50
u_{ep_b}: **11.45** u_{ep_c}: −0.38



u_{al_b}: −1.09 u_{al_c}: **8.80**
u_{ep_b}: −0.38 u_{ep_c}: 1.35

u_{al_b}: −0.19 u_{al_c}: 1.58
u_{ep_b}: −0.35 u_{ep_c}: **10.06**

Figure 2: Examples of aleatoric and epistemic uncertainties for inaccurate detections, see more examples in the appendix. From left to right: Person is a false positive; Person bounding box is not correct; A sheep is misclassified as a cow; A bird is misclassified as Aeroplane.

4 Experiments

In this section, we demonstrate the benefits of our approach. We first study the impact of using mixture modeling for object detector and then, analyze the proposed acquisition function in the context of active learning.

Datasets: We use PASCAL VOC (Everingham et al., 2010) and COCO (Lin et al., 2014) datasets. For PASCAL VOC, which provides 20 object categories, we use VOC07 or VOC07+12 for training and evaluate our results on VOC07 testing set. For COCO, which provides 80 object categories, we use COCO *train2014* for training and evaluate our results on *val2017*.

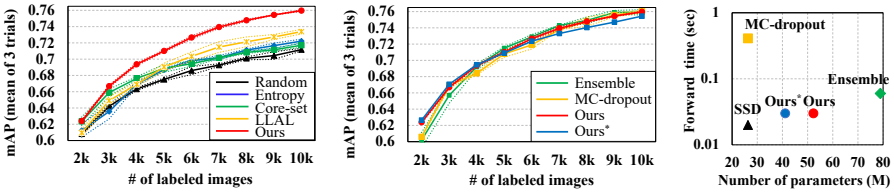
Experimental settings: We train our models for $120,000$ iterations using SGD with a batch size of 32 and a maximum learning rate of 0.001. We use learning rate warm-up for the first $1,000$ iterations and divide the learning rate by 10 after $80,000$ and $100,000$ iterations. We set the number of Gaussian mixtures to 4, see the appendix for ablation studies. Unless specified, we report performance using the mean and standard deviation of mAP of three experiments with the standard (IoU$>$0.5) and the strict (IoU$>$0.75) metrics.

4.1 Object Detection with Uncertainty Modeling

We first analyze the impact of using mixture density networks for object detection on VOC and COCO. For COCO, we use a random subset of $5,000$ training images from *train2014*. We compare the performance of our approach (Ours) and the more efficient version (Ours*) to the original SSD and several network configurations either using single or multiple Gaussians for the classification or localization heads. For the evaluation, we provide the average mAP of three experiments with the standard metric (IoU$>$0.5) and the strict metric (IoU$>$0.75).

Table 1a and 1b summarize the results of this experiment on VOC07 and COCO, respectively. As shown, all networks that include uncertainty modeling outperform the SSD on both datasets. The improvement is larger in IoU$>$0.75 for those instances using uncertainty on the localization head, probably due to the regularization effect of the proposed loss function (Choi et al., 2019). As a result, we obtain models that are robust to noisy data. We also observe that the accuracy is higher for models using a mixture network compared to using a single Gaussian. Our approach using a GMM and its more efficient variation outperforms all other variations in VOC07. In COCO, GMM outperforms all other instances and baseline and the efficient variation provides competitive results.

Figure 2 shows representative examples of uncertainty scores for several images where the detector fails to detect the object. As shown, each uncertainty value provides a different insight into some particular failure. Localization uncertainties are related to the accuracy of the bounding box pre-



(a) (b) (c)

Figure 3: **VOC07+12:** a) Comparison to published work using a single model for scoring; b) Comparison to ensemble and MC-dropout; c) Model parameters in milions and forward time in seconds.

diction, whereas classification uncertainties are related to the accuracy of the category prediction. Interestingly, in these examples, even if the predictions are wrong, uncertainty values seem to be uncorrelated suggesting each uncertainty could predict inaccurate results independently. From these results, we can conclude that the proposed approach not only can compute uncertainty in a single forward pass but also boosts the performance of the object detection network.

4.2 Active Learning Evaluation

We focus now on evaluating the performance of our approach to active learning on PASCAL VOC and COCO datasets. We use an initial set of $2,000$ and $5,000$ (Kao et al., 2018) training images from VOC07 and COCO, respectively. Then, during the active learning stage, for each image, we apply non-maximum suppression and we compute the uncertainties for each of the "surviving" objects. The scoring function aggregates these uncertainties using the maximum to provide the final informativeness score for the image. We score the set of unlabeled images and select the $1,000$ images with the highest score. We repeat this process three times for VOC07 and eight times for VOC07+12 and COCO. For each iteration, we train each model from scratch, using ImageNet pretrained weights as initialization. To verify the influence of the initial training set, we ran 5 times the first iteration obtaining an average mAP of 62.3 ± 0.09 on VOC07 which suggests little variations when experiments use a different initial subset of images.

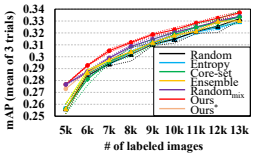
Table 2: **VOC07:** Active learning comparison to other methods.

	mAP in % (# images)		
	1st (2k)	2nd (3k)	3rd (4k)
Random	61.21	65.49	67.77
Entropy	61.21	66.81	68.08
Box-uncertainty	61.31	64.97	68.40
MC-dropout	60.59	66.45	68.29
Ensemble	60.20	66.75	68.54
Random$_{mix}$	62.43	66.36	68.47
Ours	62.43	67.32	69.43
Ours*	**62.91**	**67.61**	**69.66**

PASCAL VOC: comparison to state of the art methods. Table 2 summarizes the performance of our method compared to most relevant active learning approaches in the literature. As baselines, we use random sampling on the original SSD and, in addition, random sampling using the proposed architecture (referred as $Random_{mix}$). As shown, both instances of our approach consistently outperforms all the other methods in every active learning iteration. Interestingly, although all methods use the same initial subset of data for the first iteration, our approach yields slightly higher accuracy in the first iteration which is consistent with our experiments in the previous section. Moreover, using only $4,000$ images, our approaches outperform the accuracy achieved using the original SSD trained with the entire dataset (see Table 1a).

We now compare our approach to existing single model-based approaches on VOC07+12. We consider the state-of-the-art results reported in Yoo & Kweon (2019) including LLAL (Yoo & Kweon, 2019) and core-set (Sener & Savarese, 2018), in addition to simple baselines such as entropy (Shannon, 2001) and random sampling. We use the same open source used in Yoo & Kweon (2019). As shown in Figure 3a, our method outperforms all the other single model-based methods.

Finally, we compare our approach with methods using multiple models such as ensemble and MC-dropout active learning for object detection. For ensembles, we follow Beluch et al. (2018), building an ensemble of three independent models. For MC-dropout, we add dropout layers with $p = 0.1$ to the six convolutional layers composing the extra-layers module. We compute the image scores using 25 forward passes (Beluch et al., 2018). For these two methods, we estimate the final image informativeness score $u_{\mathcal{H}}$ as the average entropy on the classification head. Figure 3b and Figure 3c show the performance comparison and compute costs of these methods, respectively. As shown, in terms of performance, our approach performs on par with MC-dropout and ensembles. However, our approach uses a single forward pass to estimate the uncertainties, which is more efficient than



(a)

		Baseline	Ours	Ours*
mAP (%)	IoU>0.5	75.31	**75.90**	75.80
	IoU>0.75	48.70	49.36	**49.83**
# of parameters (M)		41.17	42.23	41.61
Forward time (s)		0.059	0.062	0.060

(b)

(c)

Figure 4: a) Accuracy comparison on COCO; b) Performance comparison using Faster-RCNN on VOC07; b) Transferability of datasets created using our approach.

ensembles and MC-dropout based methods. With respect to the number of parameters, MC-dropout has the same number of parameters as SSD since dropout layers do not add any new parameters. Our approach adds extra parameters for the estimation of the GMM and its parameter efficiency version to the latest layers of each head and therefore, the number of parameters is larger than SSD. In ensemble-based methods, the number of parameters is proportional to the number of models in the ensemble. As shown, our proposed methods require significantly less computational cost than MC-dropout and ensemble-based method. In short, our method provides the best trade-off between accuracy and computational cost.

COCO: comparison to state of the art methods. We compare our approach to entropy, coreset and ensembles using four independent models as suggested in Beluch et al. (2018). As baselines, we consider random sampling using the original SSD and random sampling using our mixture model network (Random$_{mix}$. Results for this experiment are shown in Figure 4a. As shown, our approach yields again higher accuracy in the first iteration and, more importantly, our approach consistently outperforms the other active learning methods through every iteration. These results suggest our approach generalizes to larger datasets that have a larger number of categories. Interestingly, the standard deviation of the results of our method is, in general, smaller than that of the other methods, suggesting that our approach is more stable than the others.

4.3 Scalability and Transferability

Our method is not limited to single-stage detectors. Here, in a first experiment we show how our approach can be applied to a two-stage detector such as Faster RCNN (Ren et al., 2015) with FPN (Lin et al., 2017). Figure 4b shows the performance and computational cost of our approach compared to the the original model using the same data from VOC dataset as in Table 1a. As shown, the accuracy of our approach outperforms while, in this case, minimizing the computational cost burden. Finally, in the last experiment, we focus on the transferability of actively acquired dataset. To this end, we compare the performance of different backbones such as VGG and Resnet-34, Resnet-50 (He et al., 2016) and Faster-RCNN trained using 4k samples acquired using our approach. For comparison, we also report the accuracy obtained using random sampling selection. These two datasets corresponds to the ones used to obtain results in Table 2. Results for this experiment are shown in Figure 4c. As shown, networks trained using samples selected by our method outperform the counterpart trained using randomly selected samples. These results suggest that our approach not only scales to other detection networks but also the datasets acquired using our approach can be transferred to other architectures.

5 Conclusions

We have proposed a novel deep active-learning approach for object detection. Our approach relies on mixture density networks to provide, in a single forward pass, a multi-modal distribution for every output of the model. We can efficiently estimate the epistemic and aleatoric uncertainty for every of these outputs. To train the mixture model, we have proposed a loss function that yields up to 2.8% accuracy improvements when compared to the baseline models. For active learning, our scoring aggregates the uncertainty from both the classification and localization heads of the model. Results in public datasets demonstrates that our approach outperforms state-of-the-art active learning methods using a single model and performs on par compared to MC-dropout and ensembles, but requiring a significantly lower computational cost. A key factor for such better performance is the combination of uncertainties from both tasks. We also demonstrated the scalability, and transferability of those datasets actively acquired using our approach.

REFERENCES

Hamed Habibi Aghdam, Abel Gonzalez-Garcia, Antonio M. López, and Joost van de Weijer. Active learning for deep detection neural networks. In *International Conference on Computer Vision (ICCV)*, 2019.

William H. Beluch, Tim Genewein, Andreas Nürnberger, and Jan M. Köhler. The power of ensembles for active learning in image classification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Kashyap Chitta, Jose M. Alvarez, and Adam Lesnikowski. Large-Scale Visual Active Learning with Deep Probabilistic Ensembles. *arXiv*, art. 1811.03575, 2018.

Kashyap Chitta, Jose M. Alvarez, Elmar Haussmann, and Clement Farabet. Less is more: An exploration of data redundancy with active dataset subsampling. *arXiv:1811.03542*, 2019.

Jiwoong Choi, Dayoung Chun, Hyun Kim, and Hyuk-Jae Lee. Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. In *International Conference on Computer Vision (ICCV)*, 2019.

Sungjoon Choi, Kyungjae Lee, Sungbin Lim, and Songhwai Oh. Uncertainty-aware learning from demonstration using mixture density networks with sampling-free variance modeling. In *International Conference on Robotics and Automation (ICRA)*, 2018.

Sungjoon Choi, Sanghoon Hong, Kyungjae Lee, and Sungbin Lim. Task agnostic robust learning on corrupt outputs by correlation-guided mixture density networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3872–3881, 2020.

David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine Learning*, pp. 201–221, 1994.

Sai Vikas Desai, Akshay Chandra Lagandula, Wei Guo, Seishi Ninomiya, and Vineeth N. Balasubramanian. An adaptive supervision framework for active learning in object detection. In *British Machine Vision Conference (BMVC)*, 2019.

Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. Scalable object detection using deep neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal in Computer Vision*, 88(2):303–338, 2010.

Di Feng, Lars Rosenbaum, and Klaus Dietmayer. Towards safe autonomous driving: Capture uncertainty in the deep neural network for lidar 3d vehicle detection. In *International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning (ICML)*, 2016.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International Conference on Machine Learning (ICML)*, 2017.

Ross Girshick. Fast r-cnn. In *International Conference on Computer Vision (ICCV)*, 2015.

Elmar Haussmann, Michele Fenzi, Kashyap Chitta, Jan Ivanecky, Hanson Xu, Donna Roy, Akshita Mittel, Nicolas Koumchatzky, Clement Farabet, and Jose M Alvarez. Scalable active learning for object detection. In *Intelligent Vehicles (IV)*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Yihui He and Jianren Wang. Deep multivariate mixture of gaussians for object detection under occlusion. *arXiv preprint arXiv:1911.10614*, 2019.

Yihui He, Chenchen Zhu, Jianren Wang, Marios Savvides, and Xiangyu Zhang. Bounding box regression with uncertainty for accurate object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2888–2897, 2019.

Stephen C. Hora. Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering and System Safety*, 54:217–223, 1996.

Chieh-Chi Kao, Teng-Yok Lee, Pradeep Sen, and Ming-Yu Liu. Localization-aware active learning for object detection. In *Asian Conference on Computer Vision (ACCV)*, 2018.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems (NeurIPS)*. 2017.

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Yongchan Kwon, Joong-Ho Won, Beom Joon Kim, and Myunghee Cho Paik. Uncertainty quantification using bayesian neural networks in classification: Application to biomedical image segmentation. *Computational Statistics & Data Analysis*, 142:106816, 2020.

David D. Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *International Conference on Machine Learning (ICML)*, 1994.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

Jeremiah Liu, John Paisley, Marianthi-Anna Kioumourtzoglou, and Brent Coull. Accurate uncertainty estimation and decomposition in ensemble learning. In *Advances in Neural Information Processing Systems*, pp. 8952–8963, 2019.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems (NeurIPS)*, 2015.

Soumya Roy, Asim Unmesh, and Vinay P. Namboodiri. Deep active learning for object detection. In *British Machine Vision Conference (BMVC)*, 2018.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations (ICLR)*, 2018.

Burr Settles. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. 2012.

Claude E. Shannon. A mathematical theory of communication. *Mobile Computing and Communications Review*, 5(1):3–55, 2001.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *International Conference on Computer Vision (ICCV)*, 2019.

Toan Tran, Thanh-Toan Do, Ian D. Reid, and Gustavo Carneiro. Bayesian generative active deep learning. In *International Conference on Machine Learning (ICML)*, 2019.

Ali Varamesh and Tinne Tuytelaars. Mixture dense regression for object detection and human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13086–13095, 2020.

Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Jaeyoung Yoo, Geonseok Seo, and Nojun Kwak. Mixture-model-based bounding box density estimation for object detection. *arXiv preprint arXiv:1911.12721*, 2019.

A Appendix

A.1 Scoring aggregation functions for active learning.

We compare the active learning results obtained using different functions to aggregate the uncertainty scores. In particular, we consider four different instances of our approach: 1) The sum of epistemic and aleatoric uncertainty on the localization head together with the entropy on the classification side; 2) The maximum value of those four measures; 3) The sum of epistemic and aleatoric uncertainties for both localization and classification and 4) The maximum value of these four uncertainties. The results for this comparison are shown in Table 3. Our approach using the maximum value of epistemic and aleatoric uncertainties consistently outperforms all the other aggregation functions for all the active learning iterations.

Table 3: **VOC07:** Comparison of scoring aggregation functions for active learning.

Aggregation function	mAP in % (# images) 1st (2k)	2nd (3k)	3rd (4k)
$\sum_{j \in \{u_{al_b}, u_{ep_b}, \mathcal{H}\}} u_j$	61.49	65.72	68.37
$\max_{j \in \{u_{al_b}, u_{ep_b}, \mathcal{H}\}} u_j$	61.49	65.94	68.67
$\sum_{j \in \{u_{al_b}, u_{ep_b}, u_{al_c}, u_{ep_c}\}} u_j$	62.43	67.04	69.09
$\max_{j \in \{u_{al_b}, u_{ep_b}, u_{al_c}, u_{ep_c}\}} u_j$	**62.43**	**67.32**	**69.43**

A.2 Parameter sensitivity

A.2.1 Accuracy as a function of K

In the main paper, we presented experiments using $k = 4$ as the number of components in the mixture model. In Table 4 we analyze the sensitivity of our results with respect to the number of components in the GMM. Specifically, we provide numbers for $K = 2$, $K = 4$, and $K = 8$. As in the main paper, we repeat the experiment three times and provide the average mAP and standard deviation for the standard metric (IoU > 0.5) and the strict metric (IoU > 0.75). We also provide the number of parameters and the forward time for each of these instances. As shown, the accuracy remains stable for these configurations with minor variations in mAP. However, there are significant variations in terms of the number of parameters and forward time as the number of parameters is proportional to K. Given these results, we selected $K = 4$ as a good trade-off between accuracy (strict and standard) and compute.

Table 4: Parameter sensitivity on **VOC07**: Accuracy and compute as a function of the number of components in the mixture model

K # of mixture	mAP (%) IoU > 0.5	IoU > 0.75	# of parameters (M)	Forward time (s)
2	70.29±0.29	45.98±0.38	37.6	0.025
4	70.19±0.36	46.11±0.38	52.3	0.031
8	70.01±0.29	45.69±0.28	81.8	0.051

A.2.2 ACCURACY AS A FUNCTION OF INPUT IMAGE RESOLUTION

In order to check for the robustness of our method with respect to the image size, here we compare the performance of the network trained using higher resolution images (512×512). The experiment is analogous to the experiment we showed in Table 1a in the main text. We compare the results of SSD (Liu et al., 2016), with the results of our method. As we can see in Table 5, as expected, increasing the resolution of the input image yields a significant improvement in mAP score for all the methods. For high-resolution input images, our method outperforms SSD in the standard metric (IoU > 0.5) by $0.28pp$, and shows significant improvement when evaluated in the strict metric (IoU > 0.75), with an improvement of $2.49pp$. That is, our method is notably better in those scenarios where we need a higher intersection between the predicted bounding box and the ground truth.

Table 5: **VOC07:** Accuracy as a function of the resolution of input image.

Method	SSD 512		SSD 300	
	IoU > 0.5	IoU > 0.75	IoU > 0.5	IoU > 0.75
SSD	73.22±0.35	45.74±0.70	69.29 ± 0.51	43.36 ± 1.24
Ours	**73.50±0.12**	**48.23±0.53**	70.19 ± 0.36	46.11 ± 0.38

A.3 SCORING AGGREGATION FUNCTIONS FOR ACTIVE LEARNING.

We summarize in Table 6 the overlap in the selection as a function of the uncertainty measure. The overlapping ratio using both uncertainties is 48% and 33% on localization and classification, respectively. More importantly, If we consider both uncertainties on localization and classification together, the overlapping ratio decreases to barely 15%. This suggests that uncertainty measures obtained for localization and classification are diversified and their combination improves the image selection process.

Table 6: **VOC07+12:** Overlapping ratio (in %) of selected images as a function of the type of uncertainty used.

	Localization		Classification	
	Aleatoric	Epistemic	Aleatoric	Epistemic
	u_{al_b}	u_{ep_b}	u_{al_c}	u_{ep_c}
u_{al_b}	100	48	6	11
u_{ep_b}	48	100	7	14
u_{al_c}	6	7	100	33
u_{ep_c}	11	14	33	100

A.4 UNCERTAINTY ANALYSIS

We now focus on analyzing the uncertainty estimates generated by our model. To this end, we compute the epistemic uncertainty for two of our models trained using different training sets: the entire dataset and a subset of $2,000$ training images. Table 7a shows the aleatoric and epistemic uncertainty values for each task as a function of the training data. For a fair comparison, these values are computed as the average value of uncertainty for the objects that belong to the intersection of instances of each model. As expected, in this case, the epistemic uncertainty of each task tends to decrease as the number of training images increases.

We further analyze the ability of the model to predict the aleatoric uncertainty. That is, the noise inherent in the observations such as sensor noise (Kendall & Gal, 2017). We train a model using the entire dataset and evaluate its performance in the original test set and an additional test set created by adding Gaussian noise $\mathcal{N}(0, 0.01)$ to the original test set. The results of this experiment are summarized in Table 7b. As in the epistemic comparison experiment, these values are computed as the average value of uncertainty for the objects that belong to the intersection of the test set instances. As shown, the aleatoric uncertainty estimated for each task tends to increase when noise is added to the test set. Based on these results, we can conclude that our model predicts not only the ignorance of the model predictions but also is able to predict noise in the data.

Table 7: **VOC07:** Aleatoric and epistemic uncertainties as a function of (**left**) training data and (**right**) noise in the test set. The epistemic uncertainty decreases as the training set increases whereas the aleatoric uncertainty increases as the noise in the test data increases.

Training data	Localization		Classification	
	Aleatoric	Epistemic	Aleatoric	Epistemic
2000	$3.01e^{-1}$	$4.58e^{-2}$	$7.02e^{-3}$	$3.16e^{-6}$
5011	$3.32e^{-1}$	$3.81e^{-2}$	$3.52e^{-3}$	$2.10e^{-6}$

(a)

Test set	Localization		Classification	
	Aleatoric	Epistemic	Aleatoric	Epistemic
Original	$3.23e^{-1}$	$3.89e^{-2}$	$3.45e^{-3}$	$1.92e^{-6}$
Noisy test set	$3.59e^{-1}$	$5.21e^{-2}$	$4.18e^{-3}$	$1.72e^{-6}$

(b)

A.5 More visual examples selected by our approach

Figure 5 shows more representative examples selected by our active learning approach. Each normalized uncertainty value provides a different insight into some particular failure. From left to right and top to bottom: One of the several bounding boxes detected as person is false positive; One of the several bounding boxes detected as cow is false positive; A horse is misclassified as a bird; A motorbike is misclassified as bicycle; One of the several bounding boxes detected as person is false positive; One of the several bounding boxes detected as horse is false positive; A bottle is misclassified as a TV/monitor; A sheep is misclassified as a bird; One of the several bounding boxes detected as person is false positive; One of the several bounding boxes detected as person is false positive; A person is misclassified as a chair; A toy (not in the VOC dataset) is misclassified as a person.

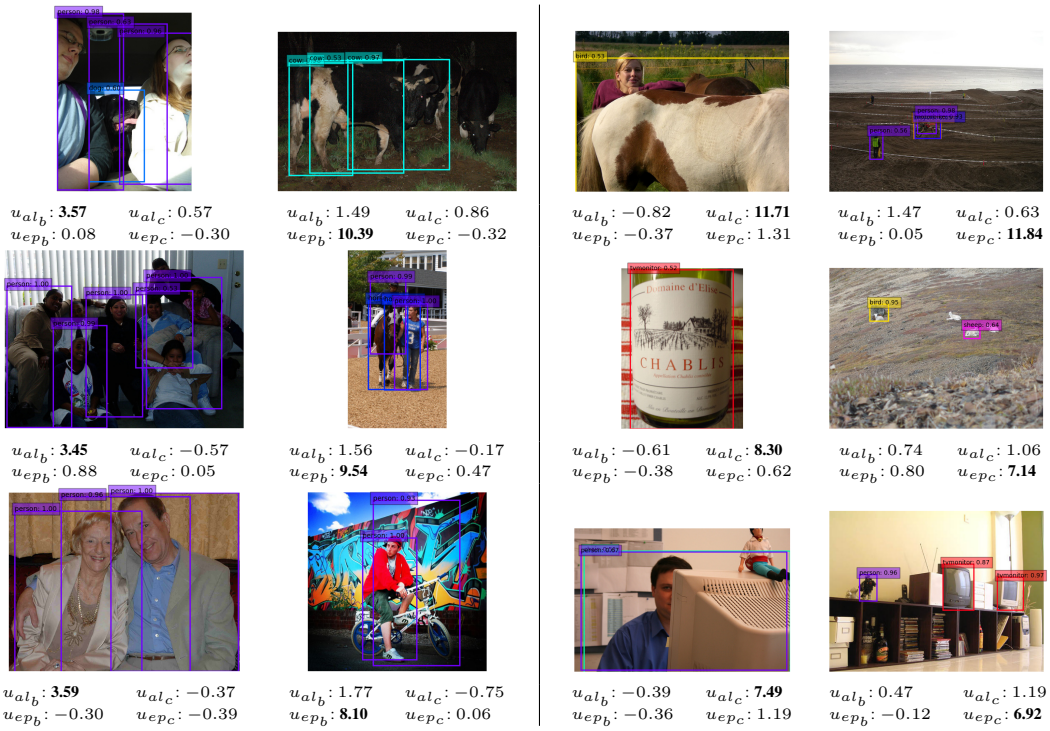


Figure 5: Examples of normalized aleatoric and epistemic uncertainties for inaccurate detections. Best viewed digitally.

A.6 Comparison to other methods on VOC07+12

In the main text, we present plots for active learning results using **VOC07+12** in Figures 3a, 3b, and 3c. Tables 8, 9 and 10 summarizes the actual numbers used to create the plots. As mentioned in the paper, in Table 8, numbers corresponding to Random, Entropy, Core-set, and LLAL are copied directly from Yoo & Kweon (2019). For MC-Dropout, to further verify the influence in the number of forward passes, we include two instances: using 25 (the one included in the main text) and 50 forward passes. As we can see in Table 9 the variation in accuracy for these two approaches is negligible while the compute needed is significantly larger for the one using 50 forward passes.

Table 8: **VOC07+12:** Comparison to published work using a single model for scoring. Numbers taken from Yoo & Kweon (2019). In bold the best values for each active learning cycle.

# of labeled images	Random	Entropy	Core-set	LLAL	Ours
2k	60.82±0.19	61.23±0.81	62.36±0.52	60.95±0.42	**62.39±0.14**
3k	64.23±0.22	63.57±0.91	65.90±0.43	64.91±0.47	**66.68±0.15**
4k	66.33±0.18	66.94±0.21	67.63±0.21	66.90±0.28	**69.37±0.16**
5k	67.51±0.17	68.70±0.15	68.88±0.48	69.05±0.45	**71.01±0.12**
6k	68.60±0.50	69.82±0.11	69.44±0.32	70.35±0.55	**72.66±0.19**
7k	69.27±0.16	70.18±0.27	70.16±0.13	71.49±0.66	**73.95±0.16**
8k	70.10±0.17	71.12±0.12	70.83±0.12	72.13±0.60	**74.78±0.12**
9k	70.44±0.47	71.66±0.31	71.15±0.16	72.73±0.30	**75.44±0.02**
10k	71.17±0.16	72.22±0.24	71.71±0.25	73.38±0.28	**75.97±0.09**

Table 9: **VOC07+12:** Accuracy Comparison to MC-Dropout and ensemble. For MC-Dropout we include two instances: using 25 forward passes and using 50 forward passes. In bold the best values for each active learning cycle.

# of images	MC-droput (50 fwd)	MC-droput (25 fwd)	Ensemble	Ours	Ours*
2k	60.59 ± 0.26	60.59±0.28	60.20±0.93	62.39±0.14	**62.65±0.14**
3k	66.60 ± 0.23	66.90±0.30	65.70±0.99	66.68±0.15	**67.06±0.18**
4k	68.90 ± 0.18	68.40±0.19	69.20±0.34	69.37±0.16	**69.46±0.05**
5k	70.60 ± 0.45	70.80±0.41	**71.50±0.18**	71.01±0.12	70.90±0.26
6k	72.00 ± 0.12	71.90±0.50	**72.90±0.27**	72.66±0.19	72.36±0.18
7k	73.67 ± 0.15	73.81±0.03	**74.29±0.04**	73.95±0.16	73.30±0.05
8k	74.68 ± 0.27	74.75±0.56	**74.91±0.41**	74.78±0.12	74.04±0.10
9k	75.49 ± 0.13	75.58±0.23	**75.89±0.25**	75.44±0.02	74.70±0.22
10k	75.67 ± 0.48	**76.01±0.19**	75.90±0.33	75.97±0.09	75.44±0.07

Table 10: Model parameters in millions and forward time in seconds using a resolution of 300×300 for the input image and $K = 4$.

	SSD	Ensemble	MC-dropout	Ours	Ours*
# of parameters (M)	26.3	78.9	26.3	52.3	41.1
Forward time (s)	0.02	0.06	0.41	0.03	0.03