BayesOD: A Bayesian Approach for Uncertainty Estimation in Deep Object Detectors

Ali Harakeh
Institute for Aerospace Studies
University of Toronto
Toronto, ON, Canada
Email: ali.harakeh@utoronto.ca

Michael Smart
Mechanical and Mechatronics Engineering
University of Waterloo
Waterloo, ON, Canada
Email: michael.smart@uwaterloo.ca

Steven L. Waslander
Institute for Aerospace Studies
University of Toronto
Toronto, ON, Canada
Email: stevenw@utias.utoronto.ca

***Abstract*—One of the challenging aspects of incorporating deep neural networks into robotic systems is the lack of uncertainty measures associated with their output predictions. Recent work has identified aleatoric and epistemic as two types of uncertainty in the output of deep neural networks, and provided methods for their estimation. However, these methods have had limited success when applied to the object detection task. This paper introduces, BayesOD, a Bayesian approach for estimating the uncertainty in the output of deep object detectors, which reformulates the neural network inference and Non-Maximum suppression components of standard object detectors from a Bayesian perspective. As a result, BayesOD provides uncertainty estimates associated with detected object instances, which allows the deep object detector to be treated as any other sensor in a robotic system. BayesOD is shown to be capable of reliably identifying erroneous detection output instances using their estimated uncertainty measure. The estimated uncertainty measures are also shown to be better correlated with the correctness of a detection than the state of the art methods available in literature.**

I. Introduction:

Deep neural networks have arisen as the dominant method for the object detection problem, demonstrating near human level performance on both the 2D [1, 2, 3, 4] and 3D [5, 6, 7] object detection tasks. Due to their high level of performance, deep object detectors have become standard components of perception stacks for safety critical tasks such as autonomous driving [5, 6, 7] and automated surveillance [8]. Therefore, the quantification of how trustworthy these detectors are for subsequent modules, especially in safety critical systems, is of utmost importance. To encode the level of confidence in an estimate, a meaningful and consistent measure of uncertainty should be provided for every detection instance.

A meaningful uncertainty measure is defined as one that is discriminant enough to allow a robotic system to achieve two important goals. First, the robotic system should be capable of using the uncertainty measure to fuse a deep object detector's output with prior information from different sources, effectively treating it as any other sensor [9]. As such, the estimated uncertainty measure should be **negatively correlated** to the correctness of the output of a detector. Second, the robotic system should be able to use the provided uncertainty measure to reliably identify incorrect estimates, including those resulting from *unknown unknowns*, where object categories, scenarios, textures, or environmental conditions have not been seen during the training phase [9] (see Fig. 1).

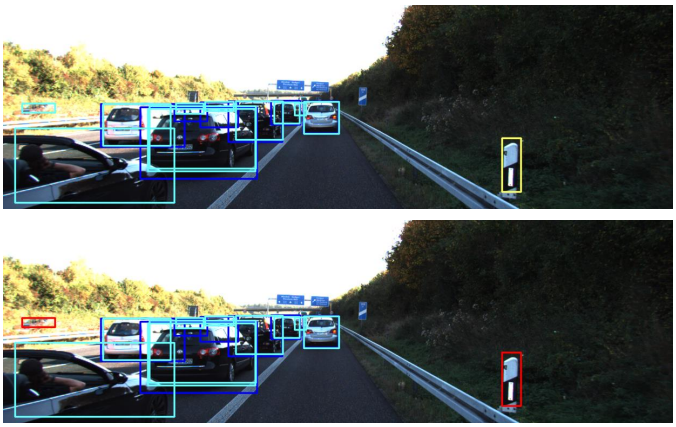

Fig. 1. **Top:** Detection results (vehicles in cyan, and pedestrians in yellow) provided by BayesOD, built on top of RetinaNet [2], and run on a frame from the KITTI object detection dataset [12]. Ground truth boxes are shown in blue for vehicles, and green for pedestrians. Erroneous detections can be seen as a pole on the side of the road detected as a pedestrian, and a vegetation patch in the top-left corner of the image detected as a vehicle. **Bottom:** The two erroneous detections can be rejected using a threshold on the maximum allowed uncertainty for a detection to be trusted, and are flagged in red.

Two sources of uncertainty can be identified in any machine learned model. *Epistemic* or model uncertainty is the uncertainty in the model's parameters, usually as a result of the confusion about which model generated the training data, and can be explained away given enough representative training data points [10]. On the other hand, *aleatoric* or observation uncertainty results from the stochastic nature of the observed input, and persist in network output despite expanded training on additional data [11].

Methods to estimate both uncertainty types in deep neural network models have been recently proposed [11], with applications in one-to-one perception tasks such as semantic segmentation or monocular depth regression. Object detectors usually output a large number of redundant detections [1, 2, 3, 4, 5, 6, 7], and as such, extending the proposed framework to object detection is not trivial. Multiple approaches to solve this problem have been proposed in literature ranging from solely considering epistemic uncertainty [13, 14], to proposing independent methods that tackle both uncertainties individually

[15, 16]. These approaches do not tackle the incorporation of prior information using the proposed uncertainty measures, and are shown in Section IV to not be able to reliably identify incorrect object detections using their estimated bounding box uncertainty. To this end, this paper offers the following contributions:

- BayesOD, a Bayesian approach for estimating the uncertainty in the output of deep object detectors, is proposed to estimate uncertainty measures associated with both category classification and bounding box regression tasks in standard object detectors.
- When applied to RetinaNet [2] on the 2D object detection problem, BayesOD is shown to provide uncertainty measures that produce large gains in discriminating power over state of the art methods in literature when used to identify erroneous detections.
- BayesOD is shown to be capable of efficiently incorporating object priors at multiple stages of the neural network inference process with closed form solutions.

II. Related Work:

A. A Common Recipe for Existing Deep Object Detectors

The majority of state of the art object detectors in 2D [1, 2, 3, 4] or in 3D [5, 6, 7] follow a standard procedure, which maps a scene representation to object instances. The object detection problem requires an object detector to provide an estimate of two states for every object instance in the scene: the category to which an object belongs and the spatial location and extent of the object, often expressed as the tightest fitting bounding box. The bounding box state $\mathcal{B}$ is modeled as a random variable drawn from a multivariate Gaussian distribution $\mathcal{B} \sim \mathcal{N}(\mu, \Sigma)$, where $\mu \in \mathcal{R}^n$ is the distribution's mean, and $\Sigma \in \mathcal{R}^{n \times n}$ is the distribution's covariance matrix. On the other hand, the category state $\mathcal{S}$ is modeled as being drawn from a Categorical (Multinoulli) distribution $\mathcal{S} \sim Cat(p_1, \ldots, p_K)$, where p_k describes the probability of the state $\mathcal{S}$ being class k written as $p(\mathcal{S} = c_k)$.

Given an input scene representation, the number of object instances in the scene is unknown a priori, and as such, the neural network is usually provided with a densely sampled grid of prior object bounding boxes, referred to as *anchors* [1, 2], or default boxes [3]. Every anchor $\mathbf{a}_i$ is spatially associated with a portion of the input scene denoted $\mathbf{x}_i$. The object detector is trained to output the parameters $[p_1, \ldots, p_K]$ and μ of the conditional distributions for each individual anchor:

$$\begin{aligned} &p(\mathcal{S}|\mathbf{x}_i, \mathcal{D}, \boldsymbol{\theta}) \\ &p(\mathcal{B}|\mathbf{x}_i, \mathcal{D}, \boldsymbol{\theta}), \end{aligned} \tag{1}$$

where $\mathcal{D}$ is the training dataset, and $\boldsymbol{\theta}$ are the object detector's parameters. Since the anchor grid is densely sampled, many anchors may be associated with each object instance in the scene. The subsequent problem is to derive a single set of object states for each set of associated anchors through the joint distribution:

$$\begin{aligned} &p(\mathcal{S}|\mathcal{X}, \mathcal{D}, \boldsymbol{\theta}) \\ &p(\mathcal{B}|\mathcal{X}, \mathcal{D}, \boldsymbol{\theta}), \end{aligned} \tag{2}$$

where $\mathcal{X}$ is the set of M input scene portions $[\mathbf{x}_i \mid i = 1 \ldots M]$ associated with a single object instance. To solve Eq. (2) using the set of outputs from Eq. (1), post-processing via Non-Maximum Suppression (NMS) is used to eliminate redundancies. Greedy NMS[17] in particular assumes that the anchor with the highest individual category score p_k within $\mathcal{X}$ is the anchor with the highest joint probability in Eq. (2).

The described procedure is considered one of the main meta-architectures for object detection, and has been extensively studied by researchers in the field [18, 19]. However, it suffers from the shortcomings described in Section I. In Section III, it is shown that **reinterpreting the neural network as a measurement device** allows for a full Bayesian treatment of the object detection task, resulting in reliable state uncertainty estimates, as well as the ability to incorporate object priors into the detection framework.

B. Regression Variance Estimation Through Loss Attenuation

To fully describe a Gaussian distribution associated with a regression output of a deep neural network, both of its sufficient statistics must be estimated by the model. Motivated by heteroscedastic regression[20], a loss attenuation formulation has been proposed [11] that estimates the variance of the regression output of a deep neural network model by modifying the regression training loss as follows:

$$\begin{aligned} L_{reg} &= \frac{1}{N} \sum_{i=1}^{N} L(\mathbf{x}_i) \\ L(\mathbf{x}_i) &= \frac{1}{2\sigma(\mathbf{x}_i)^2} ||\mathbf{y}_i - f(\mathbf{x}_i)|| + \frac{1}{2} \log \sigma(\mathbf{x}_i)^2, \end{aligned} \tag{3}$$

where $\mathbf{x}_i$ is the input to, and $f(\mathbf{x}_i)$ is the output from the neural network. Furthermore, N is the total number of regression instances, $\mathbf{y}_i$ is the ground truth regression target, $||.||$ is an L^p norm. and $\sigma(\mathbf{x}_i)$ is the **estimated** output variance. The total loss is then defined as the sample mean of the losses of individual regression targets. The first term of $L(\mathbf{x}_i)$ serves as an intelligent robust regression loss, where the model is allowed to attenuate the effect of outliers in training examples by increasing their estimated variance. The second term acts as a regularizer, preventing the model from rejecting all training examples by always setting the variance to infinity. We refer the reader to [11] for further explanation of this loss function.

C. Uncertainty Estimation In Deep Object Detectors

To capture uncertainty of the states $\mathcal{S}$ and $\mathcal{B}$, the entropy of their associated probability distributions in Eq. (2) is computed using the distributions' sufficient statistics. High state entropy is positively correlated to the state uncertainty, and is commonly used as an uncertainty measure in the state of the art [13, 14].

To estimate the uncertainty in object detection results, [13] treats the deep object detector as a **black box**, with parameters that can be stochastically sampled through Monte-Carlo (MC) Dropout [10]. The output detections of multiple stochastic runs are then clustered, and the sufficient statistics of the state

distributions in Eq. (2) for every object instance are directly estimated from the cluster members. The main advantage of this formulation lies in treating the underlying structure of the deep object detector as a black box, allowing it to be applied to various architectures with little effort. Later work [14] studied the effect of various merging algorithms on the quality of the estimated uncertainty measures from the black box method in [13]. In Section IV the uncertainty estimated for the bounding box state $\mathcal{B}$ is shown to be of little discriminative power when used to reject erroneous detection outputs, mainly because the black box method observes the output after NMS.

Another way to estimate the uncertainty in object detection results is to directly apply the formulation in Eq. (3) to provide estimates for the covariance matrix of the bounding box state $\mathcal{B}$. Examples of these **sampling free** methods include [15, 16] and are usually faster than black box methods, since a single run of the deep object detector can estimate uncertainty. Sampling free methods usually provide slightly better uncertainty estimates from the bounding box state distributions when applied to identification of erroneous detections. However, in Section IV, these methods are shown to provide a lower quality uncertainty estimate for the category state over methods utilizing MC-Dropout.

Finally, [16] proposes another method to estimate the uncertainty in deep object detectors, which exploits the **redundancy** in the output of the deep object detector *before NMS* to form spatially affiliated clusters of detection outputs, from which sufficient statistics for both object state distributions in Eq. (2) can be estimated. However, when compared to black box and sampling free methods, this redundancy based method is shown to perform the worst in terms of average precision and uncertainty quality.

Unlike all methods described in this section, BayesOD allows the incorporation of object priors at multiple stages of the object detection framework. By replacing NMS with Bayesian inference, BayesOD also outperforms all methods described in this section in terms of the discriminative power of its estimated uncertainty measures (Section IV).

III. A Bayesian Formulation For Object Detection:

The main steps of BayesOD are shown in Fig. 2. This section aims to describe the intuition and formalize the mathematical derivations involved in each of these steps. Note that throughout this section, outputs from the neural network are denoted with a $\hat{.}$ operator, and per-anchor variables are indexed with i. Variables not indexed with an i represent accumulation over several anchors.

A. Capturing Uncertainty In Neural Network Inference Results:

To capture the epistemic uncertainty of a deep object detection model, a prior distribution is imposed over its parameters $\boldsymbol{\theta}$ to compute a posterior distribution $p(\boldsymbol{\theta}|\mathcal{D})$ over the set of all possible parameters given the training data. A marginal distribution is then computed for every object state according to:

$$p(\hat{\boldsymbol{y}}_i|\mathbf{x}_i,\mathcal{D}) = \int_{\theta} p(\hat{\boldsymbol{y}}_i|\mathbf{x}_i,\mathcal{D},\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta}, \tag{4}$$

where $\hat{\boldsymbol{y}}_i$ is the output of the neural network, which can either be $\hat{\mathcal{B}}_i$ or $\hat{\mathcal{S}}_i$ for every anchor $\mathbf{a}_i$.

A simple and computationally efficient Monte-Carlo sampling method, Monte-Carlo Dropout [10], allows drawing **i.i.d** samples from Eq. (4) by performing neural network inference with dropout enabled. Using the drawn samples, the sufficient statistics of the Gaussian marginal probability distribution describing the estimated bounding box state $\hat{\mathcal{B}}_i \sim \mathcal{N}(\boldsymbol{\mu}(\mathbf{x}_i), \Sigma(\mathbf{x}_i))$ can be derived as:

$$\boldsymbol{\mu}(\mathbf{x}_i) = \frac{1}{T}\sum_{t=1}^{T} f(\mathbf{x}_i,\boldsymbol{\theta}_t) \tag{5}$$

$$\Sigma_e(\mathbf{x}_i) = \frac{1}{T}\left(\sum_{t=1}^{T} f(\mathbf{x}_i,\boldsymbol{\theta}_t) f(\mathbf{x}_i,\boldsymbol{\theta}_t)^\intercal\right) - \boldsymbol{\mu}(\mathbf{x}_i)\boldsymbol{\mu}(\mathbf{x}_i)^\intercal, \tag{6}$$

where T is the number of times MC-Dropout sampling is performed, and $f(\mathbf{x}_i,\boldsymbol{\theta}_t)$ is the bounding box regression output of the neural network for the t^{th} MC-Dropout run. The covariance matrix, Σ_e, captures the epistemic uncertainty in the estimated bounding box state $\hat{\mathcal{B}}_i$.

Since the neural network outputs the parameters of a Categorical distribution rather than categorical samples, these parameters can be derived for the Categorical marginal conditional probability distribution $\hat{\mathcal{S}}_i \sim Cat([\hat{p}_1 \dots \hat{p}_K])$ as:

$$\hat{p}_k = \frac{1}{T}\sum_{t=1}^{T} \text{SoftMax}(g(\mathbf{x}_i,\boldsymbol{\theta}_t))_k, \tag{7}$$

where SoftMax(.) is the soft max function, and $g(\mathbf{x}_i,\boldsymbol{\theta}_t)_k$ is the output *logit* of the k^{th} category, estimated at the t^{th} MC-Dropout run of the neural network.

To capture aleatoric uncertainty for the bounding box state $\mathcal{B}$, the neural network is trained to estimate the elements of the diagonal of a per-anchor aleatoric covariance matrix $\Sigma_a(\mathbf{x}_i)$, using a modified version of Eq. (3). Specifically, the loss for *every dimension* of the bounding box representation is modified as:

$$L_{reg} = \frac{1}{N_{pos}}\sum_{i=1}^{N_{pos}} L(\mathbf{x}_i) + \frac{1}{N_{neg}}\sum_{i=1}^{N_{neg}} \frac{1}{\sigma(\mathbf{x}_i)^2}, \tag{8}$$

where N_{pos} is the number of positive anchors, N_{neg} is the number of negative anchors, and $L(\mathbf{x}_i)$ is the loss in Eq. (3). The first term of the proposed loss is simply Eq. (3) applied to the positive anchor set, while the second term encourages the model to increase the total variance of the bounding box state $\mathcal{B}$ of the negative anchors. The proposed modification is empirically found to provide better numeric stability while training with higher learning rates, and for a slightly more discriminative uncertainty measure over the original as shown in Section IV.

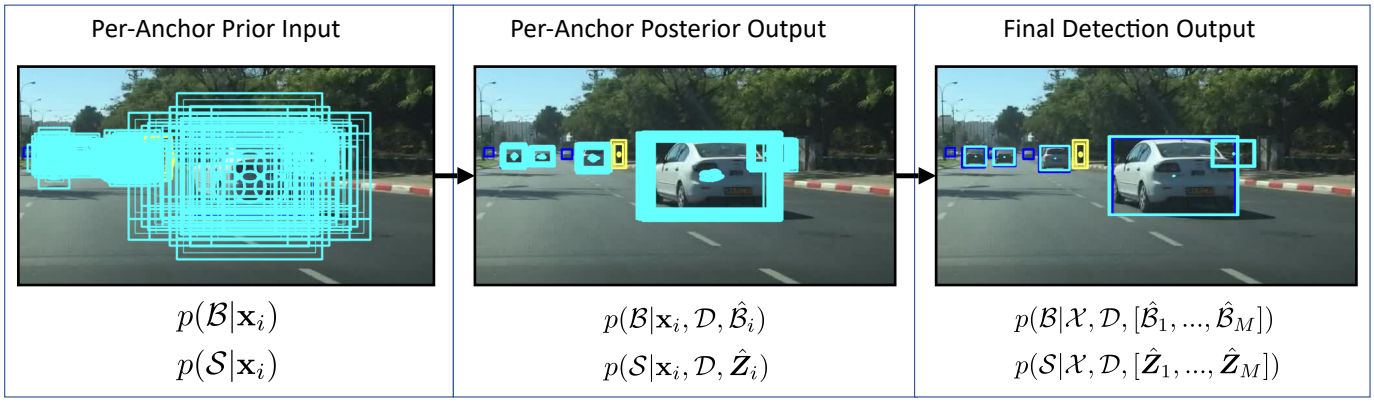

Fig. 2. The different stages of estimation employed in BayesOD, demonstrated on a testing image frame from the Berkeley Deep Drive Dataset [21]. The visualization shows car detection boxes in *cyan*, car ground truth boxes in *blue*, pedestrian detection boxes in *yellow*, and pedestrian ground truth boxes in *green*. For each bounding box, the uncertainty in the bounding box state $\mathcal{B}$ is visualized as the 2σ confidence ellipses of the centroid location, as well as its width and height at $\pm 2\sigma$. **Left:** The prior anchors associated with every detection provided by BayesOD, each having the same bounding box covariance matrix. **Middle:** The output Bayesian inference results after incorporating state estimates from the object detector. It can be seen that sets of estimates are formed by clustering with spatial affinity. **Right:** The final detections resulting from merging clusters through Bayesian inference. The final set of detections is seen to contain errors, such as the yellow pedestrian bounding box, which can be filtered out using the associated uncertainty measure.

The aleatoric covariance matrix can then be constructed from the output regressed variances as:

$$\Sigma_a(\mathbf{x}_i) = diag([\sigma^1, \ldots, \sigma^n(\mathbf{x}_i)]) \tag{9}$$

$$\sigma^j(\mathbf{x}_i) = \frac{1}{T}\sum_{t=1}^{T} \sigma^j(\mathbf{x}_i, \boldsymbol{\theta}_t), \tag{10}$$

where $\sigma^j(\mathbf{x}_i)$ is the estimated variance of the j^{th} element of the bounding box state $\mathcal{B}$, at the t^{th} MC-Dropout run of the neural network. Following [11], the final output covariance $\Sigma(\mathbf{x}_i)$ of the state $\mathcal{B}$ can then be approximated as:

$$\Sigma(\mathbf{x}_i) = \Sigma_e(\mathbf{x}_i) + \Sigma_a(\mathbf{x}_i). \tag{11}$$

No explicit treatment of the aleatoric classification uncertainty is needed, since it was found in [15] to be self-contained within the estimated parameters of the categorical distribution.

B. Incorporating State Prior Distributions:

One of the useful properties of BayesOD is that it enables incorporating per-anchor prior information in the final estimate of the states. This formulation interprets the output of the neural network as measurements of an anchor's states, which can be used to update a prior distribution over each state. Specifically, the per-anchor conditional posterior distribution describing the bounding box state $\mathcal{B}$ can be written as:

$$p(\mathcal{B}|\mathbf{x}_i, \mathcal{D}, \hat{\mathcal{B}}_i) \propto p(\hat{\mathcal{B}}_i|\mathbf{x}_i, \mathcal{D}, \mathcal{B})p(\mathcal{B}|\mathbf{x}_i). \tag{12}$$

$p(\hat{\mathcal{B}}_i|\mathbf{x}_i, \mathcal{D}, \mathcal{B})$ is a Gaussian likelihood function described by the sufficient statistics $[\boldsymbol{\mu}(\mathbf{x}_i), \Sigma(\mathbf{x}_i)]$ in equations Eq. (5) and Eq. (6), while $p(\mathcal{B}|\mathbf{x}_i) \sim \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$ is a predefined per-anchor prior distribution conditioned on the input $\mathbf{x}_i$ and assumed to be independent of the data $\mathcal{D}$. The sufficient statistics can be computed through the multivariate Gaussian conjugate update, as:

$$\Sigma'(\mathbf{x}_i) = (\Sigma_0^{-1} + \Sigma(\mathbf{x}_i)^{-1})^{-1} \tag{13}$$

$$\boldsymbol{\mu}'(\mathbf{x}_i) = \Sigma'(\mathbf{x}_i)(\Sigma_0^{-1}\boldsymbol{\mu}_0 + \Sigma(\mathbf{x}_i)\boldsymbol{\mu}(\mathbf{x}_i)). \tag{14}$$

Instead of incorporating a prior distribution directly over the categorical state $\mathcal{S}$, a Dirichlet distribution is set as a prior over the sufficient statistics $[p_1, \ldots, p_K]$. The posterior distribution of these sufficient statistics can be written as:

$$p(\mathcal{P}|\mathbf{x}_i, \mathcal{D}, \hat{\boldsymbol{Z}}_i) \propto p(\hat{\boldsymbol{Z}}_i|\mathbf{x}_i, \mathcal{D}, \mathcal{P})p(\mathcal{P}|\mathbf{x}_i), \tag{15}$$

where $\mathcal{P}$ is the set of updated sufficient statistics $[p'_1, \ldots, p'_K]$, and $\hat{\boldsymbol{Z}}_i = [\hat{\boldsymbol{z}}_1, \ldots, \hat{\boldsymbol{z}}_F]$ are F **i.i.d.** instances of the categorical random variable described by a categorical distribution with sufficient statistics $[\hat{p}_1, \ldots, \hat{p}_K]$ defined in Eq. (7). Since the likelihood function $p(\hat{\boldsymbol{Z}}_i|\mathbf{x}_i, \mathcal{D}, \mathcal{P})$ is a multinoulli distribution, the prior distribution $p(\mathcal{P}|\mathbf{x}_i)$ is chosen to be a Dirichlet distribution so that the posterior is itself a Dirichlet distribution that can be computed through conjugacy in closed form as:

$$\begin{aligned} p(\mathcal{P}|\mathbf{x}_i, \mathcal{D}, \hat{\boldsymbol{Z}}_i) &\propto \prod_{k=1}^{K} p_k^{\alpha_k - 1} \prod_{f=1}^{F}\prod_{k=1}^{K} p_k^{\mathbb{1}[\hat{z}_{fk}=1]} \\ &= Dir(\alpha'_1, \ldots, \alpha'_K) \end{aligned} \tag{16}$$

where $\mathbb{1}(.)$ is the indicator function, $\hat{z}_{fk}$ is the element in instance $\hat{\boldsymbol{z}}_f$ corresponding to category k, and $[\alpha'_k = \alpha_k + \sum_{f=1}^{F} \mathbb{1}[\hat{z}_{fk} = 1] \;\forall\; k = 1, \ldots, K]$ are the inferred parameters of the Dirichlet posterior distribution. Finally, the categorical posterior distribution describing the category state can be written as:

$$p(\mathcal{S}|\mathbf{x}_i, \mathcal{D}, \boldsymbol{Z}_i) = Cat([p'_1, \ldots, p'_K]), \tag{17}$$

where p'_k is the mean of the posterior distribution [22] in Eq. (16) written as:

$$p'_k = \frac{\alpha'_k}{\sum_{j=1}^{K} \alpha'_j}.$$

The choice of anchor priors depends on the application, and whether object information is actually available a priori. For

the rest of this paper, a weakly informative prior is chosen for the bounding box state $\mathcal{B}$, by setting $\boldsymbol{\mu}_0$ to the initial anchor position, and Σ_0 to a matrix with large diagonal entries. Similarly, by setting the parameters of the Dirichlet distribution $[\alpha_1, \ldots, \alpha_K]$ such that $\alpha_j = 1 \ \forall j \in [1, K]$, the resultant distribution over the parameters of the categorical distribution $[p_1, \ldots, p_K]$ describing $\mathcal{S}$ is also non-informative, and in fact equivalent to a uniform distribution over the open standard $(K-1)$ probability simplex.

Fig. 2 provides a visualization of how such non-informative priors (first column) are updated through neural network inference. It can be seen that multiple updated anchors are clustered around single object instances in the scene. Such redundancy is usually eliminated through post processing via NMS. The elimination process employed by NMS results in a large amount of useful information being discarded, which greatly impacts the quality of the computed uncertainty metric, especially for the bounding box state $\mathcal{B}$.

C. Bayesian Inference as a Replacement to Non-Maximum Suppression:

BayesOD uses Bayesian inference over clusters as a replacement to the elimination scheme employed by Greedy NMS. First, per-anchor outputs from the neural network are clustered using spatial affinity. Similar to NMS, greedy clustering is performed using the output category scores $[p'_1, \ldots, p'_K]$, by choosing the anchor with the highest non-background score as the cluster center, adding any anchor with an intersection over union (IOU) greater than 0.5 to the cluster, and eliminating all members in the cluster from the original updated anchor set. The clustering process terminates when all updated anchors are assigned to a cluster, or when the number of clusters exceed a predefined number. Different from NMS, BayesOD **retains** redundant anchors in clusters, rather than eliminating them, to prevent loss of information.

The output of greedy anchor clustering is H anchor clusters, $\mathcal{A}$, each containing an anchor set, $[\mathbf{a}_1, \ldots, \mathbf{a}_M]$. M is not constant and can vary between clusters in the same frame. The first anchor, $\mathbf{a}_1$, has the highest score, and as such is considered the cluster center, and will be described with its posterior state distributions in Eq. (12) and Eq. (15). The rest of the cluster members are assumed to be measurement outputs from the neural network described by the states $\hat{\mathcal{S}}_i$ and $\hat{\mathcal{B}}_i$. These measurements are used to update the states of the cluster center a_1 to arrive at the final states of an object instance. Specifically, for the bounding box state $\mathcal{B}$, the final posterior state distribution can be written as:

$$p(\mathcal{B}|\mathcal{X}, \mathcal{D}, [\hat{\mathcal{B}}_1, \ldots, \hat{\mathcal{B}}_M]) \propto p(\mathcal{B}|\mathbf{x}_1, \mathcal{D}, \hat{\mathcal{B}}_1) \prod_{i=2}^{M} p(\hat{\mathcal{B}}_i|\mathbf{x}_i, \mathcal{D}, \mathcal{B})$$
$$= \mathcal{N}(\boldsymbol{\mu}''(\mathcal{X}), \Sigma''(\mathcal{X})), \tag{18}$$

where $\mathcal{X}$ is the set of inputs $[\mathbf{x}_i \mid i = 1 \ldots M]$ of the M cluster members. The second term in the equation arrives from assuming conditional independence of the states $\hat{\mathcal{B}}_i$ of the cluster members given state $\mathcal{B}$. The sufficient statistics of Eq. (18) can be estimated in closed form as:

$$\Sigma''(\mathcal{X}) = \left(\sum_{i=1}^{M} \Sigma'(\mathbf{x}_i)^{-1} \right)^{-1} \tag{19}$$

$$\boldsymbol{\mu}''(\mathcal{X}) = \Sigma''(\mathcal{X}) \left(\sum_{i=1}^{M} \Sigma'(\mathbf{x}_i)^{-1} \boldsymbol{\mu}'(\mathbf{x}_i) \right), \tag{20}$$

where $\boldsymbol{\mu}'(\mathbf{x}_i), \Sigma'(\mathbf{x}_i)$ are the sufficient statistics of the per anchor posterior distribution derived in Eq. (12). Notice that every member of the cluster contributes to the estimation of both the mean and the covariance matrix of the final object instance state $\mathcal{B}$.

Similarly, to arrive at the final posterior distribution describing the category state $\mathcal{S}$, a similar analysis can be performed to update the sufficient statistics $\mathcal{P}$ of the cluster center with categorical measurements $[\hat{\boldsymbol{Z}}_2, \ldots, \hat{\boldsymbol{Z}}_m]$ of the rest of the cluster members. Specifically, the posterior probability of $\mathcal{P}$ can be derived as:

$$p(\mathcal{P}|\mathbf{x}_i, \mathcal{D}, [\hat{\boldsymbol{Z}}_1, \ldots, \hat{\boldsymbol{Z}}_M]) \propto p(\mathcal{P}|\mathbf{x}_1, \mathcal{D}, \hat{\boldsymbol{Z}}_1) \prod_{i=2}^{M} p(\hat{\boldsymbol{Z}}_i|\mathbf{x}_i, \mathcal{D}, \mathcal{P})$$
$$= Dir(\alpha''_1, \ldots, \alpha''_K) \tag{21}$$

where $\alpha''_k = \alpha'_k + \sum_{i=2}^{M} \sum_{f=1}^{F} \mathbb{1}[\hat{z}_{ifk} = 1] \ \forall \ k = 1 \ldots K$, and the categorical measurements $[\hat{\boldsymbol{Z}}_2, \ldots, \hat{\boldsymbol{Z}}_m]$ are assumed to be **i.i.d**. In summary, α''_k is derived by updating the per-anchor Dirichlet posterior distribution in (15) of the cluster center with index $i = 1$ with categorical measurements $\boldsymbol{Z}_2, \ldots, \boldsymbol{Z}_M$ from all cluster members. The final categorical distribution describing the state $\mathcal{S}$ can then be computed as:

$$p(\mathcal{S}|\mathcal{X}, \mathcal{D}, [\hat{\boldsymbol{Z}}_1, \ldots, \hat{\boldsymbol{Z}}_M]) = Cat(p''_1, \ldots, p''_K), \tag{22}$$

where $[p''_1, \ldots, p''_K]$ can be computed as the mean of the posterior distribution in Eq. (21) as:

$$p''_k = \frac{\alpha''_k}{\sum_{j=1}^{K} \alpha''_j}, \tag{23}$$

A major result from this subsection is that the two states of any object can be updated easily given an additional measurement from a different component of the robotic system. To perform this update, one can simply use Eq. (18) to update the state $\mathcal{B}$ with a multivariate Gaussian measurement, and Eq. (21) to update the state $\mathcal{P}$ with a Categorical measurement. The state $\mathcal{S}$ can then be inferred from $\mathcal{P}$ using Eq. (22).

IV. Experiments and Results

To show the effectiveness of BayesOD in comparison to the state of the art, it is applied to the problem of 2D object detection in image space. For training, the Berkley Deep Drive 100K Dataset (BDD) [21], which comprises of $70K$ image frames is used. For testing on data closely resembling the frames seen in training, $10K$ image frames of the validation

		Car			Pedestrian		
Test Dataset	Method	AP(%) ↑	GMUE(%) ↓	CMUE(%) ↓	AP(%) ↑	GMUE(%) ↓	CMUE(%) ↓
BDD [21]	Sampling Free [16, 15]	55.16	38.99	21.96	37.64	47.49	30.55
	Black Box [13, 14]	57.34	49.75	21.71	41.54	49.86	29.43
	Redundancy [16]	56.43	49.71	24.80	40.43	49.96	38.56
	Ours	**61.35**	**25.53**	**16.96**	**43.62**	**26.15**	**23.56**
KITTI [12]	Sampling Free [16, 15]	73.27	46.40	20.82	44.98	49.22	29.21
	Black Box [13, 14]	**74.49**	48.67	18.81	**48.20**	49.71	25.46
	Redundancy [16]	68.83	47.86	22.98	45.87	49.69	34.97
	Ours	74.31	**29.73**	**13.10**	45.18	**28.70**	**18.45**

TABLE I
THE RESULTS OF THE EVALUATION OF *Black Box* [13, 14], *Redundancy* [16], AND *Sampling Free* [15, 16] STATE OF THE ART METHODS COMPARED TO BAYESOD(OURS).

set of the BDD dataset are used. For testing on data visually different from frames which have been seen in training, the training split of the KITTI 2D object detection dataset [12] is used. The KITTI data comprises of 7481 frames, which have been collected using a different sensor and in scenes different in appearance than those seen in training data from BDD.

Deep Object Detector: RetinaNet [2] is chosen as the baseline deep object detector, and all methods used in comparison are integrated into its inference process. RetinaNet is trained to detect the *Car* and *Pedestrian* categories on the $70K$ training frames of the BDD dataset for 6 epochs using the ADAM optimizer with a batch size of 4 and an initial learning rate of 0.0003. The learning rate is reduced every 2 epochs with a decay factor of 0.1. The remaining hyperparameters are left as the default ones presented in [2].

The two chosen categories exist in both datasets, and as such, the proposed experimental setup mimics what usually occurs in practice when deploying robotic systems, where object detectors are required to detect the same categories at test time as the ones it has been trained on, but in previously unobserved environments. As seen in Fig. 1, the open-set problem is inherent to the object detection task, even when testing for the same categories in both datasets.

Evaluation Metrics: Two evaluation metrics are used to evaluate different performance criteria of uncertainty estimation methods in comparison to BayesOD. The **Average Precision (AP)** is a standard metric used to evaluate the performance of object detectors [21, 12]. Throughout this section, AP is evaluated separately for the two categories at an IOU of 0.5. The maximum average precision achievable by a detector is 100%. On the other hand, the **Minimum Uncertainty Error (MUE)** [14] is used to determine the ability of an uncertainty measure to discriminate true positives from false positives, where a detection is determined to be a true positive if it has an $IOU \geq 0.5$ with a same category ground-truth bounding box. False positives in this case could include poorly localized detections, or false detections resulting from *unknown unknowns*. Uncertainty error (UE) can then be computed using the determined true positives (TP) and false positives (FP) as:

$$UE(\delta) = 0.5\frac{|TP > \delta|}{|TP|} + 0.5\frac{|FP \leq \delta|}{|FP|}, \tag{24}$$

where δ is the uncertainty measure threshold. MUE is the best uncertainty error achievable by a detector at the best possible value of the threshold δ. The lowest MUE achievable by a detector is 0%.

A. Comparison With State of The Art Methods:

BayesOD is compared against three approaches representing the state of the art methods for uncertainty estimation methods used for object detection. The three approaches will be referred to as: *Black Box* [13, 14], *Sampling Free* [16, 15], and *Redundancy* [16]. Methods utilizing MC-Dropout use 8 fixed random seed stochastic runs of RetinaNet. No improvement in performance for any of the methods used for comparison was seen for a number of runs greater than 8. Fixing the random seed guarantees the same *per-run* weights for every method, resulting in a fair evaluation. The affinity threshold used for clustering in all methods was set to the 0.5 IOU, similar to that used for NMS in RetinaNet. For this dataset/detector combination, this threshold was shown to provide the highest AP for all methods used in comparison. The number of categorical samples F in Eq. (15) is empirically set to 10. In general, reasonable effort was made to ensure the implemented methods achieve their highest possible performance on all metrics, and to make sure a controlled and fair evaluation was achieved.

The MUE is computed for all methods based on the two described uncertainty measures. Gaussian MUE (GMUE) uses the entropy of the Gaussian distribution describing the state $\mathcal{B}$ as its uncertainty measure to be used to discriminate true positives from false positives. Note that using the entropy is seen to provide a better GMUE for all methods used for comparison, when compared to using **Total Variance** (trace of the covariance matrix) as in [14]. Similarly, Categorical MUE (CMUE) uses the entropy of the Categorical distribution describing the state $\mathcal{S}$ as its uncertainty measure.

Table I shows the results of evaluating the three methods in comparison to BayesOD, on both testing datasets. BayesOD is seen to outperform all three methods on all performance metrics when tested on the BDD dataset. The major improvement can be seen in GMUE, where BayesOD provides a tremendous reduction of 13.46% and 21.34% in GMUE over the second best method *Sampling Free* for the *car* and *pedestrian* categories respectively. BayesOD also provides a

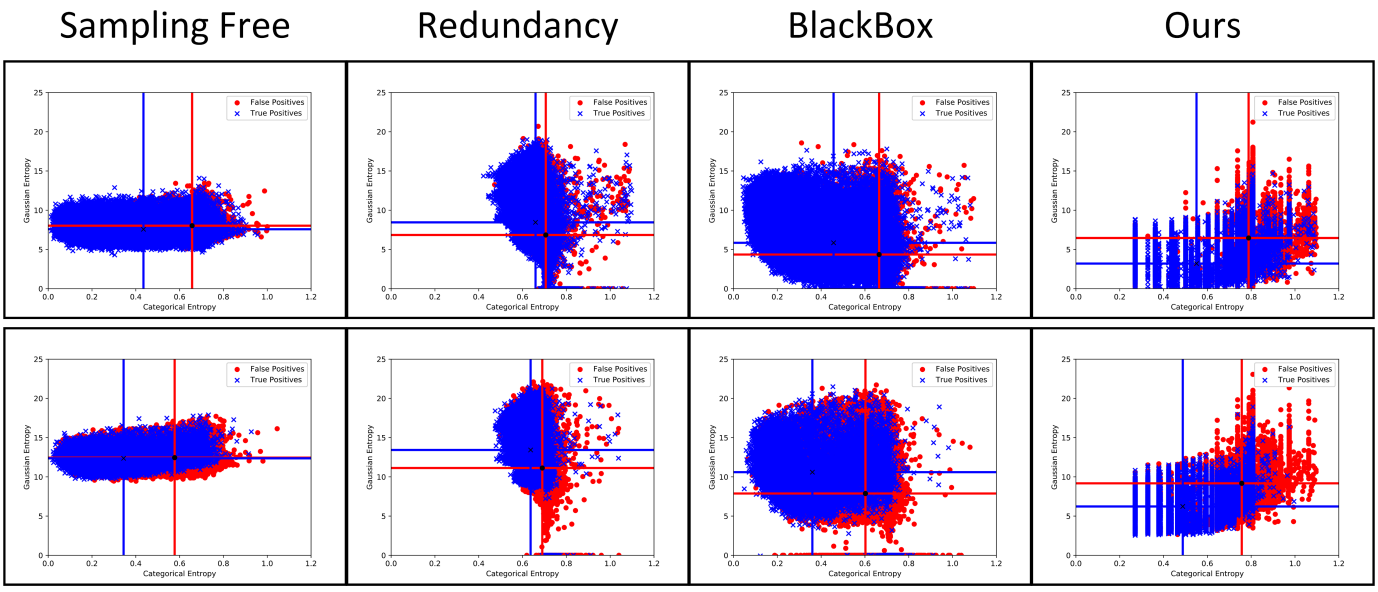

Fig. 3. Plots of the entropy of the Gaussian distribution representing the state $\mathcal{B}$, vs the entropy of the categorical distribution representing the category state $\mathcal{S}$ of true positives (in Blue) and false positives (in Red), combined over the *car* and *pedestrian* categories. The mean of both types of entropy is shown for the true positives and the false positives, as vertical and horizontal lines extending from the respective axes, and colored accordingly. The results of testing on the BDD dataset are shown in the top row, while those of testing on the KITTI dataset are shown in the bottom row. BayesOD is shown to be the only method that provides a substantially lower entropy of the state $\mathcal{B}$ for the true positives over the false positives on both datasets.

reduction of 4.75% and 5.87% in CMUE over the second best method *Black Box* for the *car* and *pedestrian* categories respectively. This reduction in MUE is accompanied with an increase of 4.01% and 2.08% in AP of the *car* and *pedestrian* categories. A similar trend is seen for both MUE metrics on the KITTI dataset, where BayesOD outperforms all other methods used for comparison. However, *Black Box* is seen to score a 0.18% and 3.02% increase in AP over BayesOD.

To get better insight on why BayesOD provides a much lower GMUE over the three methods used for comparison, Fig. 3 provides plots of the Gaussian entropy used to determine GMUE versus the Categorical entropy used to determine the CMUE for the True Positives (shown in Blue) and False Positives (shown in Red) on both the BDD dataset (Top) and the KITTI dataset (Bottom). For a meaningful uncertainty measure, the entropy, and hence the uncertainty in both states of a true positive should be lower than those of a false positive. The most optimal plot should result in a blue cluster in the lower left corner representing the true positives, and a red cluster in the top right corner representing false positives. For the Categorical entropy, all methods are shown to follow this intuitive trend to a certain extent. For the Gaussian entropy however, two of the three methods in the state of the art: *Redundancy* and *Black Box* result in exactly the opposite behaviour, where the mean of the Gaussian entropy of true positives is higher than that of the false positives. To hypothesise on why such behaviour occurs, one should observe the mechanism employed by these two methods to estimate the final covariance matrix of the state $\mathcal{B}$. Both of these methods use the clustered output of M stochastic runs to estimate a sample covariance matrix, with the only difference being that *Black Box* clusters the output of NMS, whereas *Redundancy* clusters the per-anchor output before NMS. Both of these methods lack adequate cluster merging, and explicit variance estimation, which reduces the discriminative power of their estimated uncertainty measure for the bounding box state $\mathcal{B}$. The first support for this hypothesis is that *Sampling Free*, a method that explicitly uses the per-anchor regressed covariance

		Car			**Pedestrian**		
#	Experiment	AP(%) ↑	GMUE(%) ↓	CMUE(%) ↓	AP(%)↑	GMUE(%) ↓	CMUE(%) ↓
1	Full System	**61.35**	25.53	16.96	**43.62**	26.15	23.56
2	Variance Penalty	60.89	27.37	17.44	42.60	27.15	23.83
3	No Aleatoric Variance	60.96	27.55	**16.72**	41.98	28.05	**23.15**
4	Standard NMS	61.30	36.49	17.37	42.32	41.49	24.53
5	No Marginalization	58.96	**20.74**	23.36	35.54	**25.86**	36.57

TABLE II
THE RESULTS OF ABLATION STUDIES PERFORMED ON BAYESOD TO DETERMINE WHICH COMPONENTS CONTRIBUTE THE MOST TO THE OBSERVED PERFORMANCE GAINS.

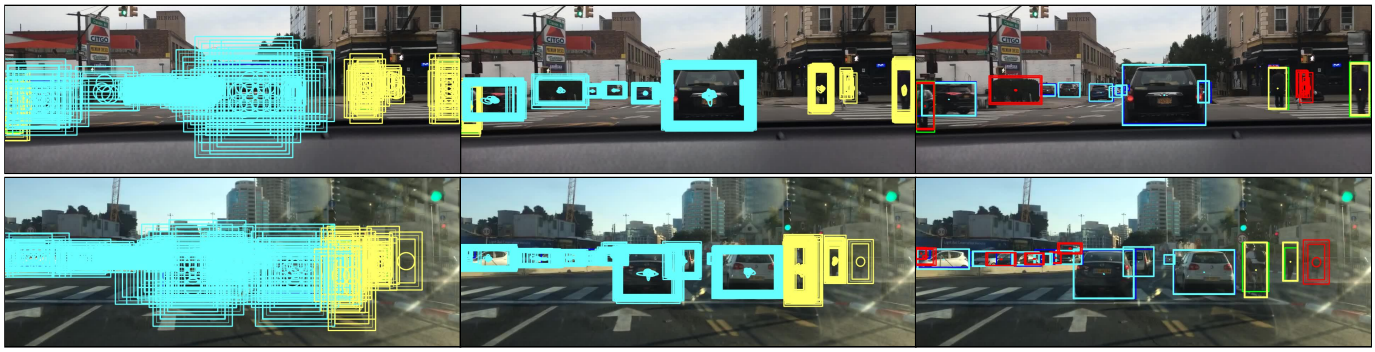

Fig. 4. Qualitative results using the same color scheme as Fig. 2, showing the progression of object state along BayesOD. **Left:** Object state non-informative priors represented by the anchor grid. Although the category state is uniform, prior boxes have been colored according to estimated category to help with correlation. **Middle:** The per-anchor posterior states, updated with the output from the neural network. **Right:** The final detection output, updated with information from spatially clustered anchors. Bounding boxes shown in *red* are rejected by the neural network for having a high entropy for both states.

matrix, provides a 10.76% and 2.37% decrease in GMUE of the *car* and *pedestrian* categories over the second runner up from *Black Box* and *Redundancy*. This is also reflected in the plots of Fig. 3, where *Sampling Free* provides a (slightly) lower mean of the Gaussian entropy for the true positives over the false positives. As a final note, Fig. 3 shows that the behaviour of the uncertainty measures for both datasets are very consistent, which provides support for conclusions being drawn on one to be extended to the other.

B. Ablation Studies:

Table II shows the results of the AP, GMUE, and CMUE for the ablation studies performed on the validation set of BDD. The results of the full BayesOD framework can be seen in experiment $\#1$. By analyzing the results of the ablation studies, the following claims are put forth:

Pushing the variance of negative anchors to increase during training provides a slightly more discriminative uncertainty in the bounding box state $\mathcal{B}$. To support this claim, RetinaNet is trained using the original attenuated loss in Eq. (3) instead of the proposed modified loss in Eq. (8). The results of BayesOD using this original loss formulation are shown in experiment $\#2$. When compared to the full system, an increase of 1.84% and 1% is observed in the GMUE for the *car* and *pedestrian* categories respectively. Although the improvement is not substantial, the proposed loss formulation in Eq. (8) is seen to be much more numerically stable, allowing for higher learning rates to be used in training. Furthermore, slightly better performance in AP and CMUE is observed when using the proposed loss formulation.

Explicit aleatoric covariance matrix estimation provides a slightly more discriminative uncertainty estimate of the bounding box state $\mathcal{B}$. To support this claim BayesOD is implemented without the update step in Eq. (11), to use only the per-anchor sample variance computed from multiple stochastic runs of MC-Dropout. The results, presented in experiment $\#3$, show an increase of around 2% is observed in the GMUE of both categories.

Greedy Non-Maximum Suppression is detrimental to the discriminative power of the uncertainty in the bounding box state $\mathcal{B}$. To support this claim, the elimination scheme of NMS is selected to retain only cluster centers, while discarding the remaining cluster members. The results presented in experiment $\#4$ show a large increase of 10.96% and 15.34% in the GMUE of the *car* and *pedestrian* categories respectively, when compared to the full system. Albeit with modest gains, BayesOD still outperforms all state of the art methods on every performance measure even when using elimination instead of Bayesian inference over cluster members.

The gains in performance on CMUE can be explained through the per-anchor marginalization over neural network parameters. To support this claim, BayesOD is stripped of the per-anchor marginalization step over the neural network parameters in Eq. (4), effectively estimating only aleatoric uncertainty. The results are presented as experiment $\#5$, and show an increase of 6.4% and 13.01% in CMUE for the *car* and *pedestrian* categories over the full system. This increase in CMUE is accompanied with a respective drop of 2.39% and 8.08% in AP for both categories. Surprisingly however, the GMUE for the two categories is 4.79% and 0.29% lower than that the full system, implying that for the bounding box state $\mathcal{B}$, incorporating the epistemic covariance matrix could hurt the discriminative power of the estimated uncertainty measure.

As a summary, the above experiments provide additional evidence of the hypothesis presented in the previous section. Replacing NMS with Bayesian Inference and explicitly incorporating aleatoric covariance matrix estimation allows for a much more meaningful uncertainty measure that has a stronger negative correlation to the correctness of an output detection.

C. Qualitative Results:

Qualitative results showing the progression of object state along BayesOD's framework are presented in Fig. 4. The thresholds producing the minimum uncertainty error for both states are used to eliminate output detections with high entropy, shown in *red*.

V. Conclusion

This paper presents BayesOD, a Bayesian approach for estimating the uncertainty in the output of deep object detector. BayesOD provides a measure of uncertainty associated with

both the bounding box and the category states of every detection instance. BayesOD also allows the incorporation of prior information at different stages of the detection process, and provides state probability distributions that can be interpreted as measurements by subsequent processes in robotic systems. This work aims to pave the path for future research directions that would use BayesOD for active learning, exploration, as well as object tracking. Furthermore, the effect of incorporating object priors into the object detection framework remains to be thoroughly studied. Future work will study the effect of informative priors originating from multiple detectors, temporal information, and different sensors on the perception capabilities of a robotic system.

References

[1] 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 28*, 2015.

[2] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[3] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *European Conference on Computer Vision (ECCV)*, 2016.

[4] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[5] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven Waslander. Joint 3d proposal generation and object detection from view aggregation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[6] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[7] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[8] Trupti M Pandit, PM Jadhav, and AC Phadke. Suspicious object detection in surveillance videos for security applications. In *Inventive Computation Technologies (ICICT), International Conference on*, 2016.

[9] Niko Sünderhauf, Oliver Brock, Walter Scheirer, Raia Hadsell, Dieter Fox, Jürgen Leitner, Ben Upcroft, Pieter Abbeel, Wolfram Burgard, Michael Milford, et al. The limits and potentials of deep learning for robotics. *The International Journal of Robotics Research*, 37(4-5):405–420, 2018.

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

[11] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, 2017.

[12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[13] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7. IEEE, 2018.

[14] Dimity Miller, Feras Dayoub, Michael Milford, and Niko Sünderhauf. Evaluating merging strategies for sampling-based uncertainty techniques in object detection. *arXiv preprint arXiv:1809.06006*, 2018.

[15] Di Feng, Lars Rosenbaum, and Klaus Dietmayer. Leveraging heteroscedastic aleatoric uncertainties for robust real-time lidar 3d object detection. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

[16] Michael Truong Le, Frederik Diehl, Thomas Brunner, and Alois Knol. Uncertainty estimation for deep neural object detectors in safety-critical applications. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018.

[17] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms: Improving object detection with one line of code. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[18] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, and Kevin Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[19] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[20] David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference On*, 1994.

[21] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu,

Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2018.

[22] Andrew Gelman, Hal S Stern, John B Carlin, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.