I Find Your Lack of Uncertainty in Computer Vision Disturbing

Matias Valdenegro-Toro
German Research Center for Artificial Intelligence
Robert-Hooke-Str 1, 28359 Bremen, Germany
matias.valdenegro@dfki.de

Abstract

Neural networks are used for many real world applications, but often they have problems estimating their own confidence. This is particularly problematic for computer vision applications aimed at making high stakes decisions with humans and their lives. In this paper we make a meta-analysis of the literature, showing that most if not all computer vision applications do not use proper epistemic uncertainty quantification, which means that these models ignore their own limitations. We describe the consequences of using models without proper uncertainty quantification, and motivate the community to adopt versions of the models they use that have proper calibrated epistemic uncertainty, in order to enable out of distribution detection. We close the paper with a summary of challenges on estimating uncertainty for computer vision applications and recommendations.
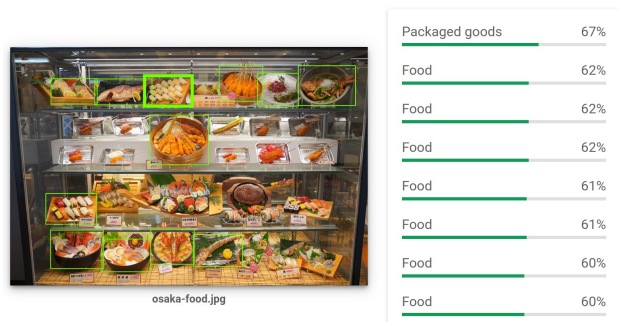
Figure 1: Computer vision models that use machine learning do not often correctly estimate their own uncertainty. In this example, the model is only around 60% confident on these food samples, while confidence should be closer to 100%. Note that some objects are not detected. Results obtained from Google Cloud Vision AI.

1. Introduction

Neural networks are currently used in many real-world applications, from medical systems to autonomous vehicles [14], robots [11] and underwater manipulator systems [22]. Despite these advances, there are many practical issues, for example, fairness and transparency in automated decisions [7], decision and prediction interpretability, and properly estimating epistemic uncertainty to preventing a model from making overconfident predictions [16]. These issues are of increasing importance as machine learning models are used to make decisions in the real world, affecting humans and their lives [14].

In this line we consider the uncertainty of model outputs, and how they are often overlooked while the field of computer vision makes progress. The purpose of uncertainty quantification in a machine learning model is to produce good estimates of predictive uncertainty, with the aims to help the user identify which predictions are trustworthy and which ones are unreliable.

But neural network models often produce overconfident incorrect predictions [16], where the prediction is incorrect but the model confidence is very high. Overconfidence produce a false sense of security, as the model produces higher confidences than it should, and the user is misled to trust incorrect predictions. Confidences are usually interpreted by a human and should "make sense" [10]. Underconfidence can also be a problem, as shown in Figure 1.

In this paper, we perform a meta-analysis of computer vision applications that are aimed or used in real-world scenarios, and show that most of them do not consider uncertainty quantification as an integral part of their system design. We believe that model without proper uncertainty are dangerous to humans and have potential ethical and legal concerns. Our work only considers computer vision methods that are backed by machine learning models, which is the standard paradigm in modern computer vision.

Probabilities produced by a softmax activation are generally uncalibrated [16] and can lead to overconfident incorrect predictions, giving a false sense of security if these confidences are used for further decision making. An appropriate model would produce calibrated probabilities and

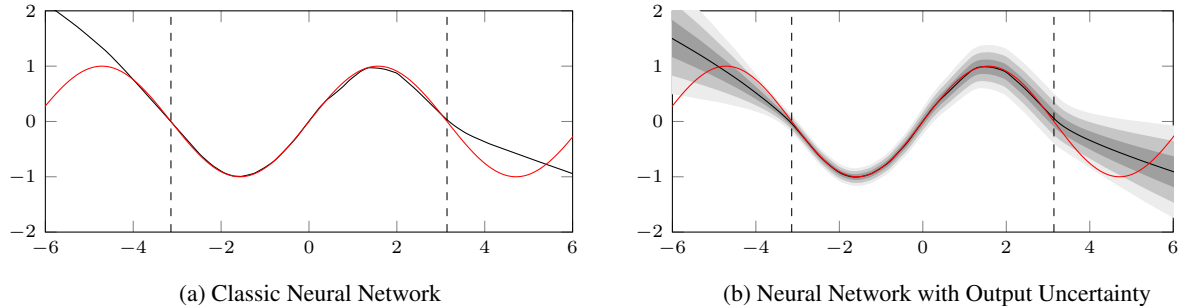(a) Classic Neural Network   (b) Neural Network with Output Uncertainty

Figure 2: Comparison between classic and neural networks with output uncertainty for regression of $f(x) = \sin(x) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2(x))$, $\sigma(x) = 0.15(1 + e^{-x})$, and the training set is $x \in [-\pi, \pi]$. The shaded areas represent one to three σ confidence intervals. Dotted lines indicate the limits of the training set. The output of the classic NN does not indicate any anomaly when predicting incorrectly outside of the training set, while the NN with uncertainty indicates extrapolation through the increasing variance output head.

well behaved output distributions. This applies both to classification and regression problems (and multi-task combinations like object detection [18]).

We point that there is a large gap between applications using machine learning and bayesian or uncertainty quantification methods that are able to estimate uncertainty in a principled way. The lesson that we want to strongly leave on the reader is:

*All applications using computer vision systems should use **only** methods producing calibrated epistemic uncertainty*

Even as some computer vision methods have probabilistic variants with good uncertainty properties [18] [14], these are not generally used in end user applications. There have been large advances in bayesian and uncertainty quantification methods for computer vision [17], there are still gaps in its use on real-world applications.

Many application fields would benefit from correct modeling of uncertainty in inputs and outputs, such as systems making medical decisions, robotics, and autonomous driving. What is common to these fields is the involvement and interaction with humans, and the potential risk of acting incorrectly in unexpected situations.

2. What is Uncertainty?

Uncertainty is a measure for lack of information, or impossibility to exactly describe a state or future outcomes. In machine learning models and computer vision systems, uncertainty is generally translated as confidence scores for different predictions: class probabilities, standard deviation for continuous outputs, etc. These indicate the level of certainty in the system given some input data.

There are two basic types of uncertainty [12] as defined by its source:

Aleatoric Uncertainty. It is the one inherent to the data and how it is captured, for example measurement uncertainty, stochastic processes, etc. It cannot be reduced by incorporating additional data. [24] In computer vision contexts, this kind of uncertainty can be annotation noise (multiple correct classes or bounding box uncertainty), camera noise that translates into uncertain pixels, etc.

Epistemic Uncertainty. This kind is inherent to the model and its processes, it can be reduced by adding more information, such as additional training data, better model specification, or improved methods [24]. This kind of uncertainty is most useful for out of distribution detection [21]. In computer vision contexts, some examples are visual similarity between classes, bounding box localization uncertainty, etc.

An example of both kinds of uncertainty is shown in a regression setting in Figure 2. The left plot shows that the classic model works well inside the training set, but it produces incorrect predictions outside of the training set, with no indication of extrapolation. The right plot shows the output of a Deep Ensemble [29], where the variance output of the model increases significantly when the input is outside of the training set range, which indicates that the model is extrapolating. Aleatoric uncertainty can be seen inside the training set, as the noise variance varies with the input ($\sigma(x) = 0.15(1 + e^{-x})$) and the output variance increase is consistent with the noise term.

2.1. Bayesian Formulation

The Bayesian neural network (BNN) is the standard formulation for uncertainty quantification in neural networks [31], with many methods being approximations of the full BNN through different techniques, such as ensembling [29] or variational inference [5].

In a BNN, the weights of each layer are probability distributions $P(w)$ that are learned in the training process.

We do not cover the learning of these distributions. Once learned, predictions can be made using the posterior predictive distribution shown below:

$$P(y \mid x) = \int_w P(y \mid x, w) P(w) dw \quad (1)$$

Standard machine learning methods usually are not able to provide high quality uncertainty estimation, since they only produce point predictions, while BNNs output probability distributions, which can encode predictive uncertainty directly. A BNN works by propagating the uncertainty in the parameters (weights) through the network, and given an input, the computed output is also a probability distribution that represents predictive uncertainty.

2.2. Calibration

An important concept in uncertainty quantification is how to evaluate the quality of the produced confidence scores. We expect that low confidence predictions have a tendency to be incorrect, and high confidence predictions should be correct most of the time. This can be quantified with the concept of reliability diagrams and different metrics that can be computed over them.

A reliability diagram is built by taking the class predictions c and confidence/probabilities p_i, where $c = \arg\max_i p_i$, i is an index over classes, and per-class confidence is $p = \max_i p_i$, then split the predictions by confidence values p into bins B_i, for each bin the accuracy $\text{acc}(B_i) = |B_i|^{-1} \sum_{j \in B_i} \mathbb{1}[c_j = t_j]$ is computed, then the mean confidence for each bin is computed as $\text{conf}(B_i) = \sum_{j \in B_i} p_j$ and then the values $(\text{conf}(B_i), \text{acc}(B_i))$ are plotted. Note that bins are usually equally spaced, and the number of bins is a hyper-parameter that must be selected carefully.

Regions where $\text{conf}(B_i) < \text{acc}(B_i)$ indicate that the model is underconfident, while regions $\text{conf}(B_i) > \text{acc}(B_i)$ indicate overconfidence. The line $\text{conf}(B_i) = \text{acc}(B_i)$ indicates perfect calibration. Examples and their interpretation are shown in Figure 3. Some metrics that can be computed from a reliability plot are:

Calibration error [16]

$$\text{CE} = \sum_i |\text{acc}(B_i) - \text{conf}(B_i)|$$

Expected Calibration error [36]

$$\text{ECE} = \sum_i N^{-1} |B_i| \, |\text{acc}(B_i) - \text{conf}(B_i)|$$

Maximum Calibration error [36]

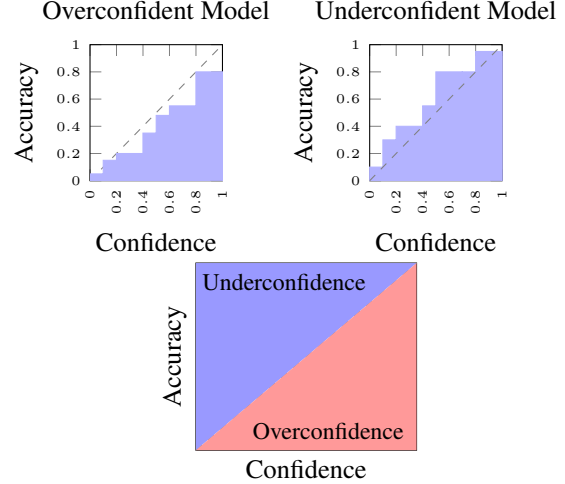$$\text{MCE} = \max_i |\text{acc}(B_i) - \text{conf}(B_i)|$$

Figure 3: Reliability plots and its interpretation to evaluate underconfidence or overconfidence on a model's predictions.

These metrics are used to numerically evaluate the calibration properties of a model. There are definitions of calibration metrics for regression problems, where accuracy is replaced by $1 - \alpha$ confidence intervals containing the true values [28]. Models can be calibrated after training [42], but generally this does not produce high quality uncertainty [39]

3. Computer Vision Models in Real-World Applications

We now make a small survey of some real-world applications of computer vision, which standard models they use, and what kind of uncertainty quantification is performed. We used multiple criteria, including popularity of the computer vision method, societal interests (like medical or agricultural applications), and applications that involve humans and their environment (such as robots). We decided to leave out any research work that presents its main results on simulation or non-realistic data, as we would like to emphasize real-world applications using computer vision methods.

Table 1 presents our survey results. Most applications using computer vision methods do not use any kind of uncertainty quantification. The only exceptions are some methods that produce uncalibrated probabilities, such as image classifiers, or , but generally these are only partial estimates of aleatoric uncertainty.

In Table 2 we also show that many of the popular computer vision methods have bayesian or probabilistic variants in the literature. We evaluated each method according to if they produce aleatoric, epistemic, or both kinds of uncertainty. For aleatoric uncertainty, we usually look if the method outputs a probability distribution and aleatoric

uncertainty is considered in their loss function. Epistemic uncertainty requires bayesian neural network modeling (or approximations like MC-Dropout) using distributions over weights, or ensembles of the same architecture.

We would like to motivate the community to prefer the use of models with appropriate uncertainty quantification.

4. The Sins of Uncertainty in CV

We believe that there are three basic mistakes of uncertainty quantification (UQ) in computer vision.

Not Using UQ in CV. Models without any kind of explicit uncertainty quantification will produce uncalibrated probabilities [16] and generally only make point predictions, which cannot perform high quality uncertainty quantification. This translates in producing incorrect overconfident predictions and not being able to use prediction confidence to detect misclassified and out of distribution examples [21].

Using Inappropriate UQ Methods. When using a method for uncertainty quantification, the type of uncertainty that the model can produce should be considered. Many methods can only estimate aleatoric uncertainty, while not being able to estimate epistemic uncertainty. The latter kind is required for tasks such as out of distribution detection, and is generally more interesting for estimating if the model is performing outside of its training distribution [47]. Aleatoric uncertainty is easier to estimate with an additional output head that estimates variance of the data, with an appropriate loss function, but epistemic uncertainty requires additional modeling such as BNNs with a distribution over the weights [31], or multiple models like ensembling [29].

An important issue is to disentangle epistemic from aleatoric uncertainty. Some methods can produce separate estimates (like Deep Ensembles [29] for regression, or SDE-Net [27], or frameworks like [26]), while others can provide a single estimate that combines both epistemic and aleatoric uncertainty. This is generally called predictive uncertainty [13].

Using UQ but not acting on Uncertainty. Uncertainty quantification can be a powerful method to improve performance, depending on the method. For example, ensembles [29] have good uncertainty quantification properties, but additionally to this they also improve performance on the original task, motivating its use for purposes other than quantifying uncertainty.

A good example of this oversight is [34], where a three member ensemble of object detectors is used as a way to improve the overall performance of the system, but without considering that disagreement between ensemble members can be used as an uncertainty measure that provides additional information to the user (most likely a physician).

Uncertainty quantification method should not be used only with the purpose of improving on a benchmark. There is much more that can be obtained than just task performance. For example, performing and evaluating out of distribution detection to know the model's limits, or just as additional information to the user.

5. The Consequences of Overconfidence

In this section we show several practical and theoretical examples about the consequences of making incorrect but highly confident predictions.

What are overconfident predictions?. A prediction is considered overconfident if it predicts an incorrect class, but the confidence is high. The threshold for an overconfident prediction varies with task and the number of classes. For classification tasks, the maximum probability or confidence is usually used to decide the output class, so any incorrect prediction with confidence higher than $\frac{1}{C}$ where C is the number of classes, can be considered overconfident.

For regression problems, an overconfident prediction is one with low standard deviation or interval length, but the true value is outside of the confidence interval.

Examples. Figure 4 shows the Mask R-CNN predictions on a test image of an Okonomiyaki, which are incorrectly classified as a Bowl and a Pizza, both with high confidence of 73% and 78%. Other objects in this image are also misclassified such as a Spoon and a Fork.

Figure 5 also shows Mask R-CNN predictions, but the image was transformed to grayscale, most likely making it out of distribution. Predicted confidences are not a good indicator that this example is far from the training distribution, and even there are some grossly incorrect predictions such as food plates predicted as persons (with confidences 77% and 81%).

Image Classification. Misclassified and out of distribution examples cannot be detected by using class confidences [21].

Object Detection. Modern object detectors produce predictions based on a thresholding class confidence of candidate bounding boxes. Overconfident misclassifications of the background class produce false positives. An important aspect is the variance of bounding boxes which is generally not considered when deciding on final predictions. An overconfident bounding box is one that poorly localizes the object with a low interval length. Most state of the art object detectors do not output any kind of bounding box uncertainty.

Instance Segmentation. Binary masks predicted for each instance box by thresholding a set of predicted soft masks, and overconfident predictions can produce inaccurate masks, as seen in Figure 4. Most issues are shared with object detectors. An example for the semantic segmentation case is presented in Figure 8.

Domain Task	Ref	Method	Uncertainty?	Challenges	Comments
Object Detection					
Breast Cancer Detection	[34]	Faster R-CNN	Ensemble	B C E G	Ensembles for performance improvement
Fruit Detection for Robots	[49]	Faster R-CNN	None	A D G	
Semantic Segmentation					
Cell Tracking	[44]	U-Net	None	G E	
Robot Navigation	[48]	AdapNet FCN	None	A C D G	
Instance Segmentation					
Fruit Detection for Robots	[50]	Mask R-CNN	None	A D G	
Tree Detection for AUVs	[38]	Mask R-CNN	None	A D G H	
Image Classification					
COVID19 Detection		AlexNet	None		
from Chest X-Ray	[1]	VGG19	None	B C E G H	Full survey at [43]
Facial Emotion Recognition	[3]	mini-Xception	None	C B F G	
Pose Estimation					
Human Pose Estimation	[8]	OpenPose	None	A D E G H	Uncalibrated confidence maps for human parts
Multimodal Driver Behavior Recognition	[32]	OpenPose	None	E G H	
Tracking					
Pedestrian Tracking	[15]	Faster R-CNN	None	A E G	
Feature Learning					
Facial Feature Learning	[45]	FaceNet	None	E F G	Embedding distances could be an (uncalibrated) uncertainty measure.
Various					
Autonomous Driving	[6]	Multiple	None	A-H (All)	
Drone Obstacle Avoidance	[30]	ResNet	None	A E D G	Uncalibrated collision probabilities.
Grasp Point Prediction	[41]	Multi-Label	None	A B F G	Uncalibrated angle probabilities
Vision & Language Robot Navigation	[2]	AlexNet Custom	None	A E G	

Table 1: Selection of computer vision tasks and their use in real-world applications, including information on any uncertainty quantification methods used. The challenges column relates to the future challenges presented in Section 8.

Autonomous Vehicle Perception. In this task the stakes are higher. Any object that is missed by the object detector can be a potential collision accident, as it has been reported in real-life situations with experimental autonomous vehicles by the US National Transport Safety Board [37], due to situations that are not present in the training set. An over-confident prediction could be a pedestrian detection in the incorrect position (sidewalk vs the road), or the production of false detections that could stop the vehicle suddenly or disable autonomous mode. This shows the importance of out of distribution detection for real-world computer vision

applications, as a way to detect situations far from the training set, and provide a signal for the system to be aware and act upon this information.

6. The Advantages of Model Uncertainty

In this section we discuss some advantages that come with properly estimating model uncertainty.

Tasks with High Uncertainty. Some tasks have a natural tendency to produce high uncertainty. For example, classes that are visually similar could lead to a model that
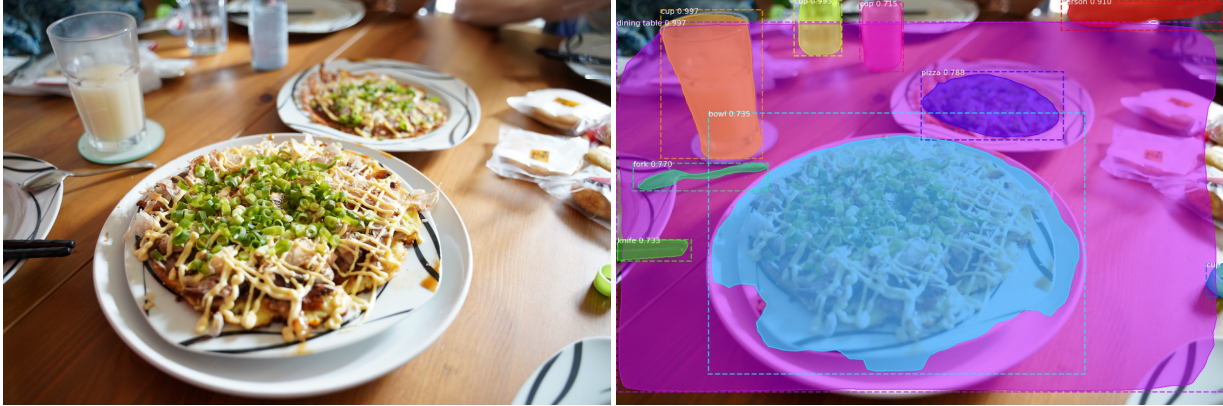
Figure 4: Incorrect detections with Mask R-CNN trained on COCO. Left presents the input image, and right the predictions. The Okonomiyaki is misclassified as a Bowl and a Pizza, and a Spoon is misclassified as a Fork, all with high confidence of $\sim 75\%$. The Bowl-Okonomiyaki has high variability in the predicted mask boundaries, signaling some amount of uncertainty.

CV Method	Probabilistic Variant	Ref	AU	EU
YOLOv3	GaussianYOLOv3	[9]	Yes	No
Faster R-CNN	Soft-NMS FRCNN	[20]	Yes	No
	MC Dropout FRCNN	[35]	Yes	Yes
	Ensemble FRCNN	[35]	Yes	Yes
SSD	MC Dropout SSD	[35]	Yes	Yes
	Ensemble SSD	[35]	Yes	Yes
RetinaNet	BayesOD	[19]	Yes	Yes
Mask R-CNN	Soft-NMS Mask R-CNN	[20]	Yes	No
SegNet	Bayesian SegNet	[25]	Yes	Yes

Table 2: Comparison between classic computer vision methods and their probabilistic/bayesian versions, including information on uncertainty quality: aleatoric (AU) and epistemic (EU).

Figure 5: Mask R-CNN trained on COCO. Test image converted to grayscale produces strange detection such as "person" with high confidence.

can easily confuse them, and this effect can be communicated by the model through properly calibrated per-class probabilities. One clear example is emotion recognition [33], where even humans have difficulties in predicting which emotion is right, and there could be multiple correct emotions being conveyed by a face. This example is shown in Figure 6.

Out of Distribution Detection. The final goal of proper epistemic uncertainty quantification is to perform out of distribution detection. In this task model confidence or uncertainty is used to indicate if a particular input sample is far from the distribution of the training set, indicating that the model might be extrapolating, or if this particular example is difficult for the model to make a decision and might be misclassified.

We believe that this is the most important application of uncertainty quantification in machine learning and computer vision models. We should not only put emphasis on model predictions but also in the model confidence, as it allows us to gauge how much a human should trust the prediction. Low confidence predictions will be interpreted by a human as not trustworthy, and development of computer vision models should take this into account.

Out of distribution detection is even more important for applications that involve humans, such as medical decisions, autonomous driving, and robotics.

7. Practical Issues

There are a series of reasons why uncertainty quantification is not used in real-world applications.
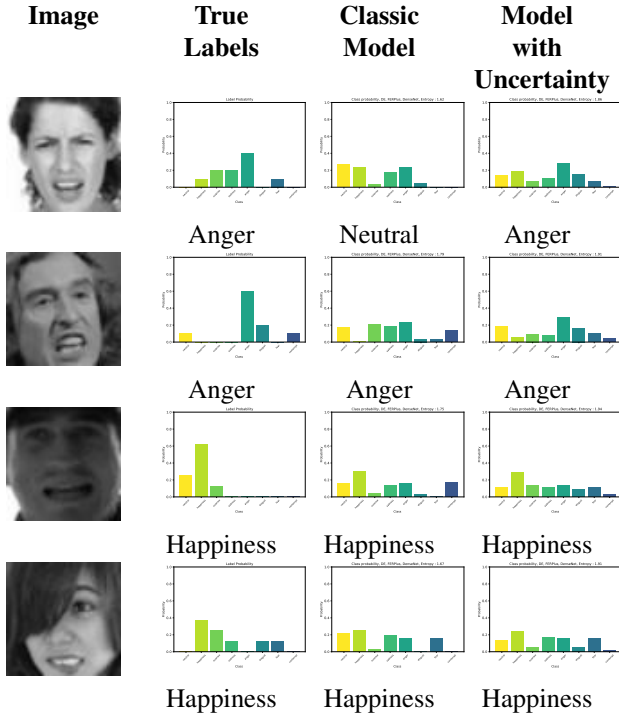
Figure 6: Comparison of DenseNet predictions on the FER-Plus dataset [4]. This is a high-uncertainty task, as the labels are probability distributions over classes (not one-hot encoded), so there are multiple "correct" classes due to the visual ambiguity in facial expressions. A model with proper epistemic uncertainty produces probabilities that make more sense, are closer to the true probabilities, and are less sparse. Classes are: Neutral, Happiness, Surprise, Sadness, Anger, Disgust, Fear, Contempt. Results adapted from [33].

Computational Costs. We believe that the biggest barrier to the adoption of uncertainty methods in computer vision is their increased computational cost compared to non-uncertainty methods. For example, using MC-Dropout, MC-DropConnect, or ensembles, requires N forward passes of the full model, with the value of N controlling a trade-off between a good posterior distribution approximation, and requiring N times the computation cost of a single model.

Approximation Quality. Most BNNs are implemented using approximations to the true posterior, since computing it is intractable. This introduces additional approximation problems, where the quality of uncertainty that is produced by the model can be poor for some applications [39]. For the out of distribution detection task, this has motivated to build methods that produce non-uncertainty outputs [46] that can still be used for out of distribution detection, like ODIN [23].

Figure 7: SSD with Inception-ResNet predictions (trained on OpenImages v4). This model seems to think that the dog is 61% carnivorous and only 44% dog. This could be due to more images labeled as carnivorous than dog, but the predicted confidences are counter-intuitive from a human perspective.

8. Future Challenges

A. Computational Cost. Properly estimating epistemic uncertainty requires the use of weight distributions for Bayesian neural networks, with increased complexity to estimate the posterior predictive distribution through sampling or ensembling. Many methods can estimate aleatoric uncertainty with a single model but this is a trade-off, since epistemic uncertainty is lost. A lot of research is devoted to try to minimize the overhead of uncertainty quantification in modern neural networks.

B. Ground Truth Uncertainty. Most datasets contain labels without uncertainty information on them. Some datasets like PASCAL VOC contain difficulty labels that are used for evaluation purposes. Uncertainty labels would help in the evaluation of probabilistic methods and bias their selection and development into uncertainty that is useful for certain applications.

C. Human Uncertainty. Epistemic uncertainty estimated by Bayesian models can be counter-intuitive for humans. An example is shown in Figure 7, where a dog is detected as 44% dog and 61% carnivorous, which is a bit strange from the human point of view. Many tree detections with low confidence are also present in the same figure, indicating a problem in localizing the tree in the background. This motivates the need for models that can produce uncertainty estimates that are closer to what a human would expect. There is already some work in this area like Peterson et al. [40].

D. Simulation tools with Uncertainty. Simulation is a

(a) Input Image   (b) Class Segmentation   (c) Per-Pixel Entropy   (d) Entropy Density
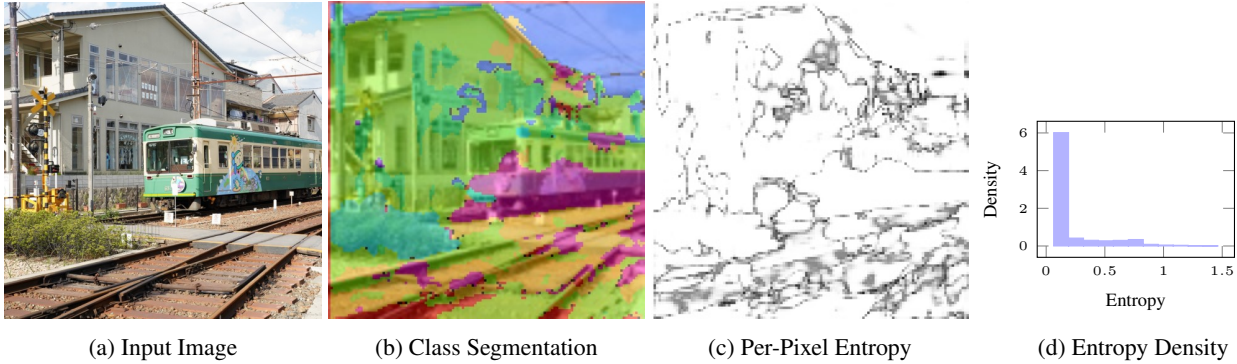
Figure 8: Semantic segmentation example with U-Net trained on Cityscapes. This railway scene is far from the training set, yet entropy does not indicate that anything is wrong, and as expected, high entropy is present between segmentation region boundaries only, not inside each regions. Entropy values are low for light shades, and high for dark shades.

powerful tool to create synthetic data. Building simulation models with uncertainty would allow the automatic production of labels for uncertainty, expanding the usefulness of simulated data. This kind of data would also allow the simulation of dangerous situations (traffic accidents, natural disasters, fires/arson, odd or uncommon scenarios, etc) that cannot be captured in real life, and be used to evaluate out of distribution detection.

E. Decision Making with Uncertainty. A long term challenge is to integrate predictions with uncertainty into the decision making processes of complex systems. A clear example of this is autonomous driving, where an ideal case is to defer driving to a human if the uncertainty in the perception systems is too high, or to use uncertainty in non-perception systems to detect faults and unexpected situations in the road.

F. Metrics. Standard metrics are not aimed at measuring good uncertainty, and there is a need to evaluate uncertainty quality. Metrics like the expected calibration error partially do this job. There should be benchmarks to evaluate the quality of aleatoric and epistemic uncertainty estimation, as well as its disentanglement. There are benchmarks for out of distribution detection.

These metrics also need to be considered during model development, and particularly when reporting a new model in the literature.

G. Out of Distribution and Domain Shift Performance. Last but not least, methods that produce high quality epistemic uncertainty estimates can still fail in certain out of distribution and domain shift scenarios. Ovadia et al. has shown how in a simple image classification setting, uncertainty quality degrades with synthetic corruptions of test images [39]. Additional research [46] points in the direction that the quality of uncertainty should be a target for improvement.

H. Openness. One important component of trustworthy

AI is that models and methods that are used in practice, can be accountable, explainable, and overall they can be trusted. But none of this is possible if we do not know which models and methods are being used by applications, and important details such as training configuration, dataset characteristics, or neural network architectures. These details are important to quantify the quality of uncertainty that these methods produce, and their out of distribution detection capabilities. Regulation will probably require such information for certification of AI systems that are used in public spaces (such as autonomous vehicles).

9. Conclusions and Future Work

This paper presented a meta-review of real world applications using computer vision, and note that most of the models that are used in the state of the art, do not have proper uncertainty quantification. We discussed common issues that arise when selecting and using models with uncertainty, and bad practices made in the literature.

We expect that out work motivates the community to move away from classical neural networks into using Bayesian neural networks and models with proper epistemic uncertainty for real-world applications, and to evaluate performance in out of distribution detection tasks, in order to ensure the safe use of machine learning models.

In the future we should see more computer vision applications that can tell the user when they do not know the answer or predict that are operating outside of their training capabilities, and we believe that this will go in the direction of safe and trustworthy artificial intelligence.

References

[1] Asmaa Abbas, Mohammed M Abdelsamea, and Mohamed Medhat Gaber. Classification of covid-19 in chest x-ray images using detrac deep convolutional neural network. *Applied Intelligence*, 51(2):854–864, 2021. 5

[2] Peter Anderson, Ayush Shrivastava, Joanne Truong, Arjun Majumdar, Devi Parikh, Dhruv Batra, and Stefan Lee. Sim-to-real transfer for vision-and-language navigation. *arXiv preprint arXiv:2011.03807*, 2020. 5

[3] Octavio Arriaga, Matias Valdenegro-Toro, and Paul Plöger. Real-time convolutional neural networks for emotion and gender classification. *arXiv preprint arXiv:1710.07557*, 2017. 5

[4] Emad Barsoum, Cha Zhang, Cristian Canton Ferrer, and Zhengyou Zhang. Training deep networks for facial expression recognition with crowd-sourced label distribution. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 279–283, 2016. 7

[5] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pages 1613–1622. PMLR, 2015. 2

[6] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016. 5

[7] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018. 1

[8] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):172–186, 2019. 5

[9] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 502–511, 2019. 6

[10] Leda Cosmides and John Tooby. Are humans good intuitive statisticians after all? rethinking some conclusions from the literature on judgment under uncertainty. *cognition*, 58(1):1–73, 1996. 1

[11] José de Gea Fernández, Dennis Mronga, Martin Günther, Tobias Knobloch, Malte Wirkus, Martin Schröer, Mathias Trampler, Stefan Stiene, Elsa Kirchner, Vinzenz Bargsten, et al. Multimodal sensor-based whole-body control for human–robot collaboration in industrial settings. *Robotics and Autonomous Systems*, 94:102–119, 2017. 1

[12] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural safety*, 31(2):105–112, 2009. 2

[13] Nicki Skafte Detlefsen, Martin Jørgensen, and Søren Hauberg. Reliable training and estimation of variance networks. In *33rd Conference on Neural Information Processing Systems*, 2019. 4

[14] Di Feng, Ali Harakeh, Steven Waslander, and Klaus Dietmayer. A review and comparative study on probabilistic object detection in autonomous driving. *arXiv preprint arXiv:2011.10671*, 2020. 1, 2

[15] Özgür Göçer, Kenan Göçer, Barış Özcan, Mujesira Bakovic, and M Furkan Kıraç. Pedestrian tracking in outdoor spaces of a suburban university campus for the investigation of occupancy patterns. *Sustainable cities and society*, 45:131–142, 2019. 5

[16] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017. 1, 3, 4

[17] Fredrik K Gustafsson, Martin Danelljan, and Thomas B Schon. Evaluating scalable bayesian deep learning methods for robust computer vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 318–319, 2020. 2

[18] David Hall, Feras Dayoub, John Skinner, Haoyang Zhang, Dimity Miller, Peter Corke, Gustavo Carneiro, Anelia Angelova, and Niko Sünderhauf. Probabilistic object detection: Definition and evaluation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1031–1040, 2020. 2

[19] Ali Harakeh, Michael Smart, and Steven L Waslander. Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 87–93. IEEE, 2020. 6

[20] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 2888–2897, 2019. 6

[21] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017. 2, 4

[22] Marc Hildebrandt, Jan Albiez, and Frank Kirchner. Computer-based control of deep-sea manipulators. In *OCEANS 2008-MTS/IEEE Kobe Techno-Ocean*, pages 1–6. IEEE, 2008. 1

[23] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10951–10960, 2020. 7

[24] Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, pages 1–50, 2021. 2

[25] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680*, 2015. 6

[26] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977*, 2017. 4

[27] Lingkai Kong, Jimeng Sun, and Chao Zhang. Sde-net: Equipping deep neural networks with uncertainty estimates. In *International Conference on Machine Learning*, pages 5405–5415. PMLR, 2020. 4

[28] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *International Conference on Machine Learning*, pages 2796–2804. PMLR, 2018. 3

[29] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, pages 6402–6413, 2017. 2, 4

[30] Antonio Loquercio, Ana I Maqueda, Carlos R Del-Blanco, and Davide Scaramuzza. Dronet: Learning to fly by driving. *IEEE Robotics and Automation Letters*, 3(2):1088–1095, 2018. 5

[31] David JC MacKay. The evidence framework applied to classification networks. *Neural computation*, 4(5):720–736, 1992. 2, 4

[32] Manuel Martin, Alina Roitberg, Monica Haurilet, Matthias Horne, Simon Reiß, Michael Voit, and Rainer Stiefelhagen. Drive&act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2801–2810, 2019. 5

[33] Maryam Matin and Matias Valdenegro-Toro. Hey Human, If your Facial Emotions are Uncertain, You Should Use BNNs! In *Women in Computer Vision @ ECCV*, 2020. 6, 7

[34] Scott Mayer McKinney, Marcin Sieniek, Varun Godbole, Jonathan Godwin, Natasha Antropova, Hutan Ashrafian, Trevor Back, Mary Chesus, Greg S Corrado, Ara Darzi, et al. International evaluation of an ai system for breast cancer screening. *Nature*, 577(7788):89–94, 2020. 4, 5

[35] Dimity Miller, Niko Sünderhauf, Haoyang Zhang, David Hall, and Feras Dayoub. Benchmarking sampling-based probabilistic object detectors. In *CVPR Workshops*, volume 3, 2019. 6

[36] Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015. 3

[37] NTSB. Collision between vehicle controlled by developmental automated driving system and pedestrian tempe, arizona march 18, 2018. *Nat. Transport. Saf. Board, Washington, DC, USA, Tech. Rep. NTSB/HAR-19/03 PB2019-101402*, 2019. 5

[38] Nuri Erkin Ocer, Gordana Kaplan, Firat Erdem, Dilek Kucuk Matci, and Ugur Avdan. Tree extraction from multi-scale uav images using mask r-cnn with fpn. *Remote Sensing Letters*, 11(9):847–856, 2020. 5

[39] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 3, 7, 8

[40] Joshua C Peterson, Ruairidh M Battleday, Thomas L Griffiths, and Olga Russakovsky. Human uncertainty makes classification more robust. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9617–9626, 2019. 7

[41] Lerrel Pinto and Abhinav Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 3406–3413. IEEE, 2016. 5

[42] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999. 3

[43] Michael Roberts, Derek Driggs, Matthew Thorpe, Julian Gilbey, Michael Yeung, Stephan Ursprung, Angelica I Aviles-Rivero, Christian Etmann, Cathal McCague, Lucian Beer, et al. Common pitfalls and recommendations for using machine learning to detect and prognosticate for covid-19 using chest radiographs and ct scans. *Nature Machine Intelligence*, 3(3):199–217, 2021. 5

[44] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 5

[45] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 5

[46] Adrian Schwaiger, Poulami Sinhamahapatra, Jens Gansloser, and Karsten Roscher. Is uncertainty quantification in deep learning sufficient for out-of-distribution detection? *CEUR-WS*, 2020. 7, 8

[47] Natasa Tagasovska and David Lopez-Paz. Single-model uncertainties for deep learning. *Advances in Neural Information Processing Systems*, 32:6417–6428, 2019. 4

[48] Abhinav Valada, Johan Vertens, Ankit Dhall, and Wolfram Burgard. Adapnet: Adaptive semantic segmentation in adverse environmental conditions. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4644–4651. IEEE, 2017. 5

[49] Shaohua Wan and Sotirios Goudos. Faster r-cnn for multi-class fruit detection using a robotic vision system. *Computer Networks*, 168:107036, 2020. 5

[50] Yang Yu, Kailiang Zhang, Li Yang, and Dongxing Zhang. Fruit detection for strawberry harvesting robot in non-structural environment based on mask-rcnn. *Computers and Electronics in Agriculture*, 163:104846, 2019. 5