Improved Predictive Uncertainty using Corruption-based Calibration

Abstract

We propose a simple post hoc calibration method to estimate the confidence/uncertainty that a model prediction is correct on data with covariate shift, as well as Out-of-Distribution (OoD) data. We achieve this by synthesizing surrogate calibration sets by corrupting the calibration set with varying intensities of a known corruption. We then calibrate our model by measuring the similarity of the test set with these surrogate sets. Our method demonstrates significant improvements on benchmark calibration performance [Ovadia et al., 2019] on a wide range of covariate shifts, and OoD data.

1 INTRODUCTION

As deep learning models become more ubiquitous, it has become increasingly critical to estimate their predictive uncertainty i.e. how reliable their predictions are. This is particularly important in healthcare, financial, and legal settings where a human user makes a decision aided by a deep learning model. The predictive uncertainty of a deep classification model is typically the probability distribution of the sample across the classes. The baseline method is to simply use the softmax probabilities of the model as a surrogate for the class membership probabilities [Hendrycks and Gimpel, 2017]. However, such probability estimates are known to lead to overconfident models [Nguyen et al., 2015], and several approaches have been proposed to calibrate these probabilities. Such methods include non-Bayesian ones such as temperature scaling [Guo et al., 2017], dropout [Srivastava et al., 2014, Gal and Ghahramani, 2016], and model ensembles [Lakshminarayanan et al., 2017], as well as Bayesian approaches such as Stochastic Variational Bayesian Inference (SVBI) for deep learning [Graves, 2011, Blundell et al., 2015, Louizos and Welling, 2016, 2017, Wen et al., 2018], among others.

All these approaches produce calibrated probabilities for in-distribution data in varying degrees of success. However, it has been found that the quality of the uncertainty predictions deteriorates significantly for data under distributional shift, also known as covariate shift, as well as OoD data [Hendrycks and Dietterich, 2019, Ovadia et al., 2019]. Ovadia et al. [2019] demonstrate this by performing a large-scale benchmark analysis of existing methods for predictive uncertainty under dataset shift, simulated as varying intensities of known corruptions to in-distribution data.

In this work, we propose a simple and efficient post hoc method for model calibration under distributional shift and for OoD data. Our method provides improved results on the large-scale benchmark Ovadia et al. [2019] (see Figure 1). The key idea is that we can estimate the dataset shift just using the model outputs by comparing with known corruptions of in-distribution data, and use this to better calibrate our models. We propose two variants to account for varying operating conditions at prediction time: (i) a Single Image method, and (ii) a Multi-Image method. Our methods are add-ons to other calibration methods, and show impressive results on never seen types of corruptions, even though they have only been exposed to known corruption.

2 RELATED WORK

Models trained on a given dataset are unlikely to perform as well on a shifted dataset [Hendrycks and Dietterich, 2019, Ovadia et al., 2019], and there are inevitable tradeoffs between accuracy and robustness [Chun et al., 2020]. Several approaches have been proposed to increase model robustness, typically evaluated on the benchmark corrupted datasets CIFAR-10-C and ImageNet-C. Hendrycks et al. [2019a] shows that while fine-tuning a pre-trained model does not improve accuracy compared to training a model from scratch, it does improve the quality of the uncertainty estimates. Our methods are a simple post hoc add-on to calibrate an already trained model.

Submitted to the 37th Conference on Uncertainty in Artificial Intelligence (UAI 2021). **To be used for reviewing only**.

Simply training models against corruptions can fail to make models robust to new corruptions [Vasiljevic et al., 2016, Geirhos et al., 2018]. However, Hendrycks et al. [2020] train models with a carefully designed new data augmentation technique called AUGMIX, and are able to improve both robustness and uncertainty measures. In contrast, our work applies a data augmentation technique at the calibration stage, which avoids having to retrain the models from scratch, and can be a simple post-hoc fix to calibrate trained models.

Krishnan and Tickoo [2020] introduced a new loss function that leverages accuracy versus uncertainty calibration, improving the model uncertainty estimates. They demonstrate that it can be used as post-hoc calibration method as well. This work can be seen as orthogonal to ours as the methods we propose here can be used as an add-on to any calibration method.

In the context of OoD data, Shao et al. [2020] propose a confidence calibration method that uses an auxiliary classifier to identify mis-classified samples, thus allowing them to be assigned low confidence. Nado et al. [2020] argues that the internal activations of deep models also suffer from distributional shift in the presence of OoD data. They thus propose to recompute the batch normalization coefficients at prediction time using a sample of the unlabeled images from the test distribution, hence improving the accuracy and ultimately the calibration. However, their work requires knowing the internal activations of the model, and hence is a white-box model which works on deep neural networks. In contrast, we improve any model's calibration treating it as a black box, and hence do not require to know its internal functionalities.

Park et al. [2020] and Wang et al. [2020] focus on the more general problem of unsupervised domain adaptation, where one assumes that unlabeled examples from the test distribution share the same classes as those in the training distribution. Park et al. [2020] propose an approach based on importance weighting to correct for the covariate shift in the data, together with learning an indistinguishable feature map between training and test distributions. Wang et al. [2020] extend the temperature scaling method into domain adaption achieving more accurate calibrations with lower bias and variance without introducing any hyperparameters.

However, Park et al. [2020], Wang et al. [2020] need to be recalibrated for every new type of corruption. In contrast, our calibration method works across corruptions, even though it has been calibrated for only one type of corruption. While Park et al. [2020], Wang et al. [2020] are interested in the purely OoD setting (MNIST versus SVHN), we show improved results on both covariate shift as well as OoD. Both these require domain adaptation methods which are likely to fail with a small number of test images. In contrast, our Single Image method can automatically calibrate for any shifted distribution without needing a batch of images, while our Multi-Image method is more sample-efficient as it provides comparable performance even with a random small subset of the test batch of images.

Furthermore, the above methods require knowledge of the feature distribution. In contrast, our approach only requires knowing the softmax probabilities of the model. Thus, our method treats the model as a black box and can therefore be extended to classifiers other than neural networks.

While our proposed methods are geared towards confidence calibration, a byproduct is OoD detection performance. We

1. train on CIFAR-10, calibrate using "contrast" corruption as before, and evaluate on the SVHN dataset [Netzer et al., 2011],
2. train on MNIST, calibrate using rotation as the corruption, evaluate on Fashion-MNIST [Xiao et al., 2017] and Not-MNIST [Xiao et al., 2017].

3 CALIBRATION

3.1 SUPERVISED CALIBRATION

Consider the K-class classification problem, where $\boldsymbol{x} \in \mathcal{X}$ is a set of inputs, such as images, and $y \in \{1, \dots, K\}$ denotes the corresponding labels. The inputs and labels are drawn i.i.d. from the joint distribution $p(\boldsymbol{x}, y)$. Here y is a sample from the conditional distribution $p(y \mid \boldsymbol{x})$.

While this analysis applies to general classifiers, we specialize to the familiar case of neural networks parameterized by $\boldsymbol{\theta}$: $f_{\boldsymbol{\theta}}(\boldsymbol{x})$. The model $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ is trained using a training dataset $\mathcal{D}_{\text{in}}^{\text{train}}$, with the hyper-parameters selected using a validation/calibration dataset $\mathcal{D}_{\text{in}}^{\text{cal}}$, and evaluated using a test set $\mathcal{D}_{\text{in}}^{\text{test}}$. All the datasets $\mathcal{D}_{\text{in}}^{\text{train}}$, $\mathcal{D}_{\text{in}}^{\text{cal}}$ and $\mathcal{D}_{\text{in}}^{\text{test}}$ consist of finite samples drawn i.i.d. from $p(\boldsymbol{x}, y)$. In what follows, we drop the subscript $\boldsymbol{\theta}$ on f to ease notation.

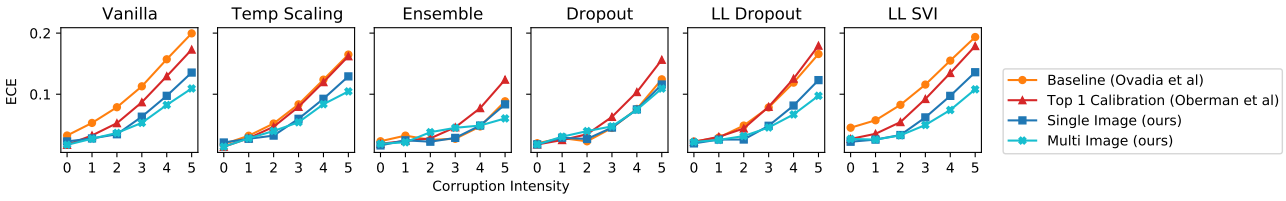
Usually, $f : \mathcal{X} \to [0,1]^K$ has a terminal softmax layer applied to its linear outputs $g(\boldsymbol{x})$, i.e. $f(\boldsymbol{x}) = \text{softmax}(g(\boldsymbol{x}))$. In this case, the model outputs a probability distribution on the K output labels given an input $\boldsymbol{x}$ from $\mathcal{X}$ (we can also consider the case where the model outputs scores). The classification of the model is given by the most likely output,

$$\hat{y}(\boldsymbol{x}) = \arg\max_k f(\boldsymbol{x})_k, \quad \text{and} \quad p^{\max}(\boldsymbol{x}) = \max_k f(\boldsymbol{x})_k.$$

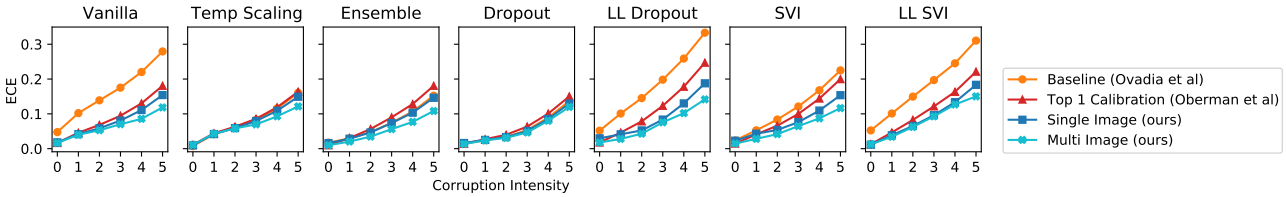
The model confidence (or uncertainty) c_k for label k is defined as the probability that the true label is k given the classifier's softmax output for that label $f(\boldsymbol{x})_k$:

$$c(\boldsymbol{x}; p, f)_k = \mathbb{P}_{p(\tilde{\boldsymbol{x}}, y)} \left[y = k \mid f(\tilde{\boldsymbol{x}})_k = f(\boldsymbol{x})_k \right]. \tag{1}$$

We write $c(\boldsymbol{x}, t; p, f)$ to emphasize c's dependence on both f and $p(\boldsymbol{x}, y)$, since the distribution will change below. We shorten the notation whenever it is clear from context.



(a) ImageNet



(b) CIFAR-10

Figure 1: Mean ECE (lower is better) of the benchmark implementation [Ovadia et al., 2019] (orange), Top1 binning [Oberman et al., 2020] (red), and our Single Image and Multi-Image methods (blue) for ImageNet (top) and CIFAR-10 (bottom). For each method, we show mean ECE across corruption intensities, across different corruption types. The ECE from our methods is better across almost all methods and intensities, with the greatest improvement at the higher intensities. See Tables 3 and 4 in the Appendix for numerical comparisons.

For deep learning models, the *Vanilla* approach consists in simply estimating $c(\boldsymbol{x})$ by the softmax probabilities $f(\boldsymbol{x})$. Generally speaking, these softmax probabilities are not an accurate prediction of the class probabilities $c(\boldsymbol{x})$ [Domingos and Pazzani, 1996]. In particular, for deep neural network models they are overconfident predictions [Guo et al., 2017]. These values become even more overconfident under distribution shift [Ovadia et al., 2019].

The goal of *supervised model calibration* [Park et al., 2020] is to estimate c empirically by $\hat{c}$, using a finite set of labeled samples $\mathcal{D}_{\text{in}}^{\text{cal}}$ drawn from $p(\boldsymbol{x}, y)$. The error between the true and the estimated confidences is typically measured by Expected Calibration Error (ECE↓) [Guo et al., 2017]:

$$ECE = \mathbb{E}_{p(\boldsymbol{x},y)}\left[|c(\boldsymbol{x}) - \hat{c}(\boldsymbol{x})|\right]. \tag{2}$$

Another measure is the Brier score (BS↓) [DeGroot and Fienberg, 1983] which estimates the mean squared error between correctness of prediction and confidence score:

$$BS(\mathcal{D}) = \frac{1}{|\mathcal{D}|}\sum_{(\boldsymbol{x}_i,y_i)\in\mathcal{D}}\left(\mathbf{1}_{\hat{y}_i=y_i} - \hat{c}^{\text{Top-1}}(\boldsymbol{x}_i,;q,f)\right)^2. \tag{3}$$

For both ECE and BS metrics, lower is better. Although other measures have been used ([Nguyen et al., 2015, Hendrycks and Gimpel, 2017]), ECE and BS metrics are the typical benchmarks, which are also used in our main comparison [Ovadia et al., 2019].

3.2 COVARIATE CALIBRATION

In this work, we are interested in calibrating a classifier on data that is shifted from the training distribution. This is the *covariate calibration* problem: find estimated calibrated probabilities c that work as well on distributions q with a covariate shift from the training distribution p, i.e. $q(y|\boldsymbol{x}) = p(y|\boldsymbol{x})$ but $q(\boldsymbol{x}) \neq p(\boldsymbol{x})$. We are thus interested in estimating $c(\boldsymbol{x};q,f)_k$:

$$c(\boldsymbol{x};q,f)_k = \mathbb{P}_{q(\tilde{\boldsymbol{x}},y)}\left[y = k \mid f(\tilde{\boldsymbol{x}})_k = f(\boldsymbol{x})_k\right]. \tag{4}$$

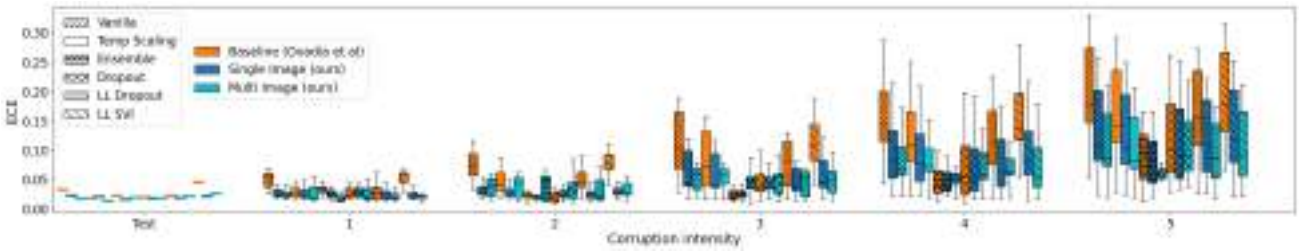
The challenge lies in the fact that while we are given a dataset of labeled examples $\mathcal{D}_{\text{in}}^{\text{cal}}$ drawn from p, we only have a dataset of unlabeled examples $\mathcal{D}_{\text{out}}^{\text{test}}$ drawn from q. In other words, Equation 4 requires labels to estimate c, which are not available.

Often, instead of per-class probabilities, the quantity of interest is the probability of the *correct classification* (Top-1 correctness). For example, Oberman et al. [2020] showed that calibration error using binning is improved by focusing on correct classification. Thus, here we focus on estimating the Top-1 confidence $\hat{c}^{\text{Top-1}}$ given by

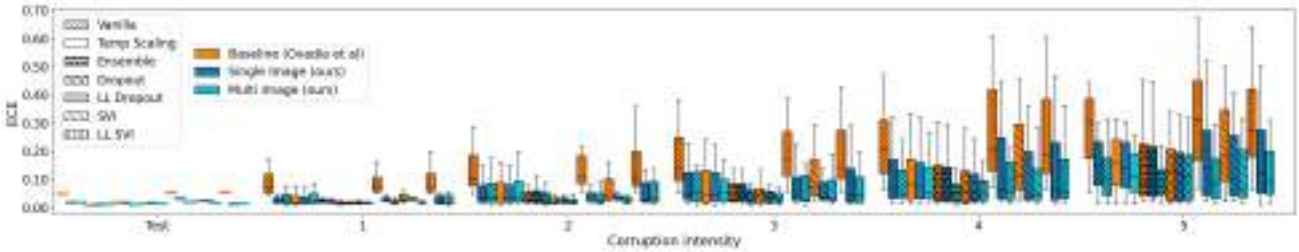
$$c^{\text{Top-1}}(\boldsymbol{x};q,f) = \mathbb{P}_{q(\tilde{\boldsymbol{x}},y)}\left[y = \hat{y}(\tilde{\boldsymbol{x}}) \mid p^{\max}(\tilde{\boldsymbol{x}}) = p^{\max}(\boldsymbol{x})\right] \tag{5}$$

However, we emphasize that it can be easily extended to other variants, such as Top-5 correctness as well as class-wise calibration.

OoD: We are also interested in OoD datasets. Unlike the covariate shift, the ground truth label of OoD datasets is not necessarily one of the K classes, and the goal is simply for the models to output lower confidence levels.



(a) ImageNet



(b) CIFAR-10

Figure 2: ECE (lower is better) of the benchmark implementation [Ovadia et al., 2019], and our Single Image and Multi-Image methods as post hoc calibration after each of them, for ImageNet (top) and CIFAR-10 (bottom), across corruption intensities. For each method, we show the quartiles summarizing the results on each corruption intensity. The ECE using our calibration is consistently better than the baseline.

4 OUR METHOD

Calibration gets worse as the data distribution moves further away from the training distribution. Our contribution is that we propose a fix to current calibration methods to control for distribution shift. Our method can be freely used *on top of* other methods to improve calibration (see section 5).

On an unknown data set, we have access to the $p^{\max}$ model values (as a distribution). We conjecture that the *dataset shift can be replaced by model output shift* for calibration. While the former is a complex distribution shift, the latter is one dimensional and can be handled by simple statistical techniques. We use an approach from anomaly detection based on outlier exposure methods [Hendrycks et al., 2019b].

Recall that we are interested in the *covariate calibration* problem where the goal is to find $c^{\text{Top-1}}(\boldsymbol{x}; q, f)$ on a distribution where we do not have labels to calibrate as in supervised calibration. In practice, this means we cannot use Equation 4 to estimate $c(\boldsymbol{x})$, since it requires labels.

Our solution is to find a surrogate shifted dataset (where the labels are known), with its $p^{\max}$ model distribution being similar to the unknown one. Then, we can use this surrogate dataset to calibrate the unknown one in a supervised fashion by simply setting $c^{\text{Top-1}}(\boldsymbol{x}; q, f)_k = c^{\text{Top-1}}(\boldsymbol{x}; q^*, f)$, where q^* denotes the surrogate dataset distribution.

We empirically observe that these values match provided the $p^{\max}(\tilde{\boldsymbol{x}})$ distribution for $\tilde{\boldsymbol{x}} \sim q^*$ is close to the $p^{\max}(\boldsymbol{x})$ distribution for $\boldsymbol{x} \sim q$. Figure 3 TOP shows the $p^{\max}$ distributions of corrupted CIFAR10-test set, and six surrogate calibration sets synthesized from the CIFAR10-calibration set by adding increasing levels of a different corruption. It can be seen that the 5th surrogate distribution represents the closest match with the test distribution. Indeed, in Figure 3 BOTTOM, we see that this corresponds to the least calibration error.

Hence, in practice, given a set of finite samples $\mathcal{D}_{\text{in}}^{\text{cal}}$ drawn from $p(\boldsymbol{x}, y)$ not seen during the training of the model, we form J distinct calibration sets $\mathcal{D}_{\text{in}}^{\text{cal},j}$, $j = \{1, \ldots, J\}$, by corrupting the data with a known corruption at different levels of intensity. This step is equivalent to drawing samples from distributions $q^j(\boldsymbol{x}, y)$, $j = \{1, \ldots, J\}$ with a covariate shift from $p(\boldsymbol{x}, y)$. However, unlike the unlabelled data drawn from q, the labels of q^j are known. Therefore, we can apply supervised calibration methods on q^j to obtain confidence estimates. The final uncertainty estimate is then an average of these, weighted by the likelihood of the test image (Single Image Method), or the set of images under the best q^j (Multi-Image Method) .

Top-1 Binning: For supervised calibration, we choose the Top-1 binning method [Oberman et al., 2020] due to its simplicity and efficiency. We emphasize that in practice any calibration method can be chosen.

First, we estimate the naive top-1 confidence $p^{\max}(\tilde{\boldsymbol{x}}) = \max_k f(\tilde{\boldsymbol{x}})_k$ for all samples $(\tilde{\boldsymbol{x}}, \tilde{y})$ in $\mathcal{D}_{\text{in}}^{\text{cal},j}$. We then partition these $p^{\max}$ values into equally sized bins B_m. Given an image $\boldsymbol{x}$ drawn from an unknown distribution $q(\boldsymbol{x}, y)$ for which the

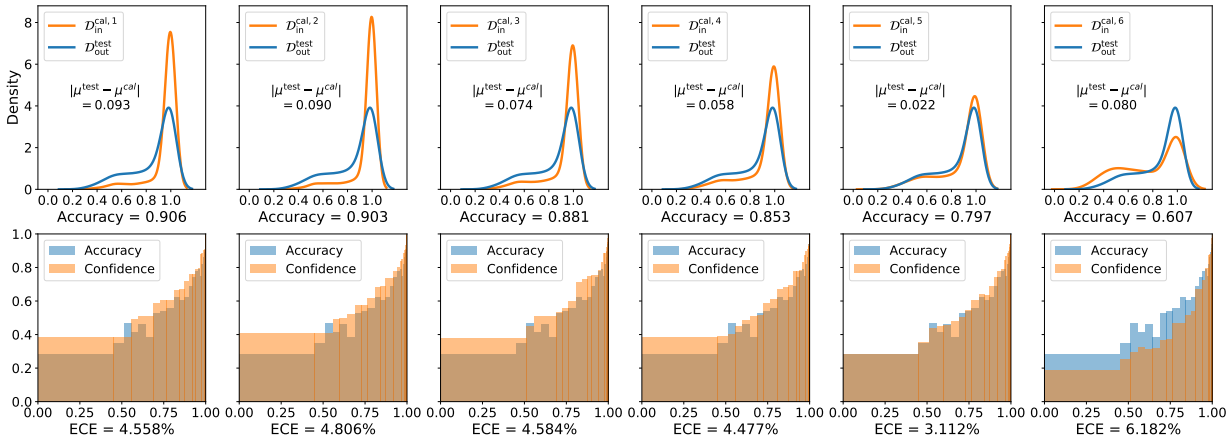
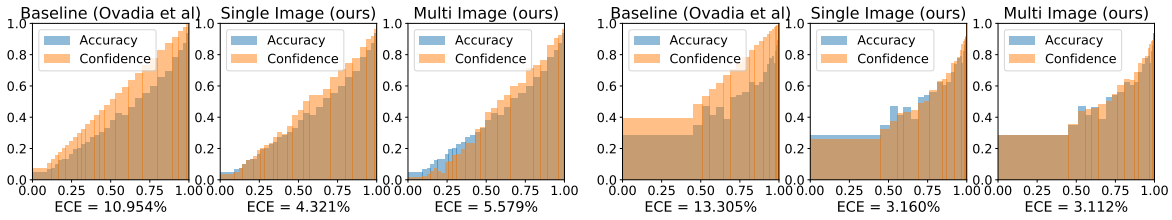


Figure 3: TOP: Probability density (using kernel density estimation) of $\mathcal{D}_{\text{out}}^{\text{test}}$ (blue) obtained by corrupting CIFAR10-test images with the "elastic transform" at intensity 4, and the $p^{\max}$ distribution of each calibration set $\mathcal{D}_{\text{in}}^{\text{cal},j}$ (orange) obtained by corrupting CIFAR10-cal images with varying intensity of a different corruption "contrast". BOTTOM: The respective accuracy and calibration confidence. The minimum calibration error is achieved precisely when the means of the distributions are the closest.



(a) Imagenet (b) CIFAR-10

Figure 4: Visualization of calibration errors for the Vanilla method on corrupted images on ImageNet (left) and CIFAR-10 (right) using the elastic transform corruption with intensity 4. The x-axis is the range of $p^{\max}$ values. We visualize as binned histograms the model accuracy (blue) (Equation 10), and the confidence estimates (orange) (Equation 11) (brown is where they overlap). The gap between the orange and blue curves represents the calibration error.

confidence needs to be estimated, let $p^{\max}(\boldsymbol{x}) \in B_m$. We then approximate $c^{\text{Top-1}}(\boldsymbol{x}; q^j, f)$ for each j as:

$$\hat{c}^{\text{Top-1}}(\boldsymbol{x}; q^j, f) = \frac{1}{|B_m|} \sum_{p^{\max}(\tilde{\boldsymbol{x}}) \in B_m} \mathbf{1}_{\hat{y}(\tilde{\boldsymbol{x}})=\tilde{y}} \tag{6}$$

where $\mathbf{1}_{\hat{y}(\tilde{\boldsymbol{x}})=\tilde{y}}$ is 1 if $\hat{y}(\tilde{\boldsymbol{x}}) = \tilde{y}$, else 0.

For each $j = \{1, \ldots, J\}$, we estimate the probability density $h^{\text{cal},j}$ of the $p^{\max}$ values using a histogram, by binning the $p^{\max}$ of the images in $\mathcal{D}_{\text{in}}^{\text{cal},j}$.

We then propose two methods based on the operating conditions: (i) to classify a single image drawn from $q(\boldsymbol{x}, y)$, or (ii) to classify multiple images from $q(\boldsymbol{x}, y)$ simultaneously.

Single Image Method: We first estimate the likelihood of the covariate shift level of the test image $\boldsymbol{x}$ under each of the calibration sets. We then take the corresponding weighted average of the calibrated probabilities:

(i) The probability that the $p^{\max}(\boldsymbol{x})$ value came from the $p^{\max}$ distribution of $\mathcal{D}_{\text{in}}^{\text{cal},j}$ is:

$$\lambda_j(p^{\max}(\boldsymbol{x})) = \frac{h^{\text{cal},j}(p^{\max}(\boldsymbol{x}))}{\sum_{i=1}^{C} h^{\text{cal},j}(p^{\max}(\boldsymbol{x}))} \tag{7}$$

(We make the standard assumption that the *a priori* likelihoods of the calibration sets are all equal)

(ii) The predicted confidence, i.e., the estimate of $c^{\text{Top-1}}(\tilde{\boldsymbol{x}}; q, f)$ is then the weighted average:

$$\hat{c}^{\text{Top-1}}(\boldsymbol{x}; q, f) = \sum_{j=1}^{J} \lambda_j(p^{\max}(\boldsymbol{x}))\, \hat{c}^{\text{Top-1}}(\boldsymbol{x}; q^j, f) \tag{8}$$

Ideally, we would like to obtain $\lambda_j(p^{\max}(\boldsymbol{x}))$ close to a one-hot vector for the calibration set with the closest $p^{\max}$ distribution to that of the test images. Since we only have a single image, we opted for a Bayesian estimation of the λ_j. However, in the Multi-Image case, we can use a simpler formula based on the statistics of the multi-image test. We indeed find better results in the Multi-Image case than those of Single Image (see section 6).

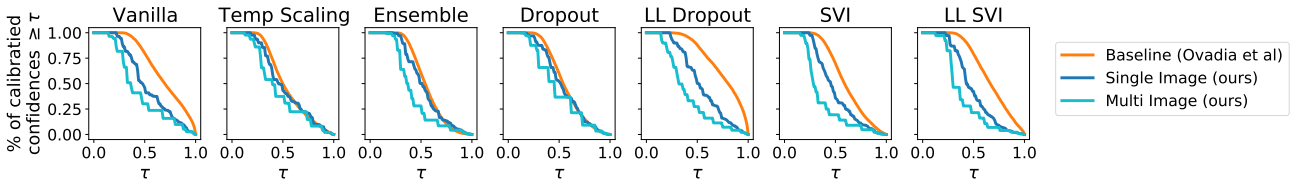
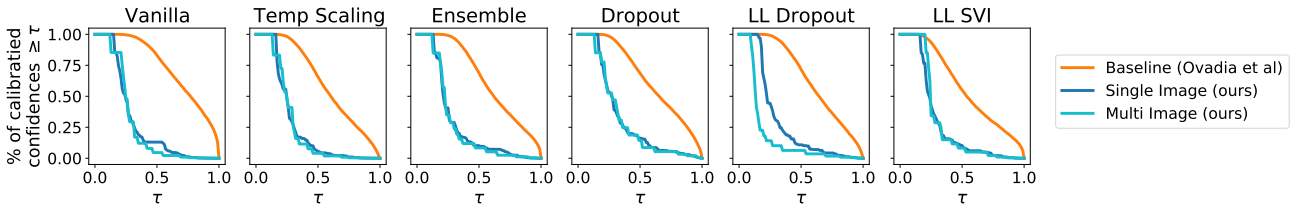
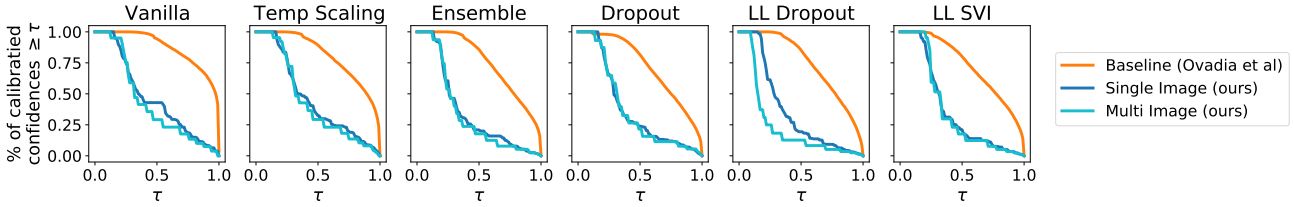


Figure 5: Cumulative histogram of model confidence above threshold τ, of model trained on CIFAR-10, tested on OoD data: SVHN. Our methods (blue) are less confident on OoD data than the benchmark method [Ovadia et al., 2019] (orange).



(a) Fashion MNIST



(b) Not-MNIST

Figure 6: Cumulative histogram of model confidence above threshold τ, of model trained on MNIST and tested on OOD data: Fashion-MNIST (top) and Not-MNIST (bottom). Our methods (blue) are significantly less confident on OoD data than the benchmark method [Ovadia et al., 2019] (orange).

Multi-Image Method: Given a test set of samples $\mathcal{D}_{\text{out}}^{\text{test}} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\}$ drawn from $q(\boldsymbol{x}, y)$, where $m > 1$.

(i) Record the $p^{\max}$ values of the test images: $P^{\text{test}} = \{p^{\max}(\boldsymbol{x}) \mid \boldsymbol{x} \in \mathcal{D}_{\text{out}}^{\text{test}}\}$, and compute their mean μ^{test}.

(ii) Compare μ^{test} to the means $\mu^{\text{cal},j}$ of $P^{\text{cal},j} = \{p^{\max}(\boldsymbol{x}) \mid \boldsymbol{x} \in \mathcal{D}_{\text{in}}^{\text{cal},j}\}$ for $j = \{1, \ldots, J\}$, and find the closest mean for the calibration sets, $i = \arg\min_j |\mu^{\text{test}} - \mu^{\text{cal},j}|$. Set

$$\hat{c}^{\text{Top-1}}(\boldsymbol{x}; q, f) = \hat{c}^{\text{Top-1}}(\boldsymbol{x}; q^j, f) \tag{9}$$

Alternative distances: Instead of the distance between the means, any other difference between distributions could be used. We have tried the Kolmogorov-Smirnoff statistic and the Wasserstein distance between the cumulative distributions of the $p^{\max}$ values (see Figure 8 in the appendix), and do not find significant change in performance.

Corrupted + Clean surrogate calibration sets: We want our multiple surrogate calibration sets to cover a wide range of different distribution shifts. However, without the clean images, the Single Image method would become uncalibrated for in-distribution images as the calibration sets would have a disproportional amount of corrupted images (see Figure 9 in the appendix). Hence, we synthesize the surrogate calibration sets with equal amounts of clean and corrupted data.

More efficient mean calculation: Instead of computing the distance from the full calibration set, we computed it for a random subset of the full calibration set, and found similar performance (see Figure 10 in the appendix). We performed calibration based on surrogate calibration subsets of randomly sampled 100 images from each surrogate calibration set. This shows that the performance improvement is due to our Multi-Image method.

We point out that our method is very efficient as most computation is only in computing histograms in the Single Image method. As such, the added cost of our method is negligible.

5 EXPERIMENTS

5.1 BASELINE APPROACHES

Our proposed method belongs to post-hoc confidence calibration, i.e. our method can be used on top of any other method for calibration. We consider the following calibration approaches provided by the benchmark dataset (Ovadia

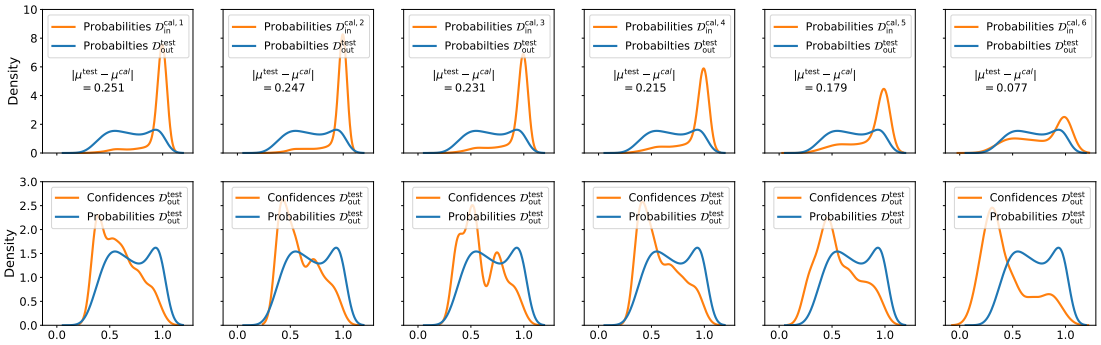


Figure 7: TOP: Probability density (using kernel density estimation) of the model probabilities of $\mathcal{D}_{\text{out}}^{\text{test}}$ (blue), and the $p^{\max}$ values of each calibration set $\mathcal{D}_{\text{in}}^{\text{cal},j}$ (orange). BOTTOM: The confidences $\hat{c}^{\text{Top-1}}$ (orange) on the SVHN dataset as a result of Top-1 binning calibration on each $\mathcal{D}_{\text{in}}^{\text{cal},j}$.

et al. [2019]) to be used before ours:

- Vanilla (Maximum softmax probability)
- Temperature scaling [Guo et al., 2017]
- Dropout [Srivastava et al., 2014, Gal and Ghahramani, 2016]
- Ensembles [Lakshminarayanan et al., 2017]
- Stochastic Variational Bayesian Inference (SVI) [Graves, 2011, Blundell et al., 2015, Louizos and Welling, 2016, 2017, Wen et al., 2018]
- Approximate Bayesian inference on the last layer only [Riquelme et al., 2018]
  - (LL SVI) Mean field stochastic variational inference on the last layer
  - (LL Dropout) Dropout only on the activations before the last layer.

All these methods ultimately use the softmax probabilities, and therefore use $p^{\max}$ as their estimate for $c^{\text{Top-1}}$. The difference between the methods is how these probabilities are obtained. We refer to Ovadia et al. [2019] for more details on how each method was implemented. We show improvements in performance by using our Single Image and Multi-Image calibration on top of each of these methods.

5.2 EVALUATION METRIC

In order to compare the methods, we compute the ECE by binning the data. For Top-1 confidence, the bins are based on based on the probability of the most probable class according to the classifier f, i.e. $p^{\max}(\boldsymbol{x})$. For each bin B_m, $m \in \{1,\dots,M\}$, we estimate the true (Top-1) model confidence by

$$c^{\text{Top-1}}(B_m) = \frac{1}{|B_m|} \sum_{(\boldsymbol{x},y)\in B_m} \mathbf{1}_{\hat{y}(\boldsymbol{x})=y} \tag{10}$$

where $\mathbf{1}_{\hat{y}(\boldsymbol{x})=y}$ is 1 if $\hat{y}(\boldsymbol{x}) = y$, else 0.

Similarly, the empirical bin model confidence is given by

$$\hat{c}^{\text{Top-1}}(B_m) = \frac{1}{|B_m|} \sum_{(\boldsymbol{x},y)\in B_m} \hat{c}^{\text{Top-1}}(\boldsymbol{x};q,f). \tag{11}$$

Then, the bin ECE is calculated as the binned version of equation 2, the weighted-average of the absolute difference between $c^{\text{Top-1}}(B_m)$ and $\hat{c}^{\text{Top-1}}(B_m)$:

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{N} |c^{\text{Top-1}}(B_m) - \hat{c}^{\text{Top-1}}(B_m)|, \tag{12}$$

where N is the total number of test samples, and $|B_m|$ is the number of samples in bin B_m. The actual value of ECE can depend on the binning procedures: equally spaced or equally sized. Equally sized bins are more effective for calibration since they reduce statistical error [Nixon et al., 2019]. In our experiments, we use $M = 30$ equal-sized bins.

5.3 DATASETS

We consider the problem of image classification, in particular for the CIFAR-10 and ImageNet datasets, and their corrupted counterparts CIFAR-10-C and ImageNet-C [Hendrycks and Dietterich, 2019]. The latter were formed by applying common real-world corruptions at different levels of intensity. Such corruptions include brightness (variations in daylight intensity), Gaussian noise (in low-lighting conditions) and Defocus blur (when the image is out of focus). For a complete list of the corruptions for Imagenet see Table 5, and for CIFAR-10 see Table 6 in the Appendix.

5.4 CALIBRATION

Our calibration methods require the synthesis of multiple calibration sets of varying corruption from the original calibration set. We used "contrast" as the corruption, and generated $J = 6$ calibration sets. While the datasets mentioned above potentially contain various types of corruptions, we leave out "contrast" from the test images.

Choice of corruption for calibration: We performed a cross-validation study over the choice of corruption used to generate the calibration sets (always leaving it out of the corruptions used at test time). Figure 11 (in the appendix) plots the mean and variance of the ECE across different choices for calibration corruptions. We find that the choice of corruption for calibration does not affect the overall performance significantly.

Corrupted + Clean: We split the CIFAR-10- and Imagenet-test set images into 5000 $\mathcal{D}_{\text{in}}^{\text{cal}}$, and remaining $\mathcal{D}_{\text{in}}^{\text{test}}$. We form $J = 6$ calibration sets $\mathcal{D}_{\text{in}}^{\text{cal},j}$. For $j = 1$, we simply take the 5000 clean images $\mathcal{D}_{\text{in}}^{\text{cal}}$. For $j > 1$, we take union of the clean images $\mathcal{D}_{\text{in}}^{\text{cal}}$ and their "contrast" corrupted counterparts with intensity level $j-1$, hence containing 10,000 images.

It is to be noted that the corrupted images at test time have never been seen by the model at either the training or the calibration stage. $\mathcal{D}_{\text{out}}^{\text{test}}$ (CIFAR-10-C and Imagenet-C) is formed by perturbing the images in $\mathcal{D}_{\text{in}}^{\text{test}}$ with different corruptions (as mentioned in Hendrycks and Dietterich [2019]), with the exception of contrast. Hence, $\mathcal{D}_{\text{in}}^{\text{cal},j}$ and $\mathcal{D}_{\text{out}}^{\text{test}}$ are disjoint. Despite this, our calibration shows improved results on $\mathcal{D}_{\text{out}}^{\text{test}}$, both in cases of covariate shift and OoD.

6 RESULTS

6.1 RESULTS ON COVARIATE SHIFT

Similar to Ovadia et al. [2019], in Figure 2 we summarize our results using a whisker plot of the distribution of the ECE scores. A zoom in of this for two corruption intensities is shown in Figure 12 in the appendix. In addition, for ease of visual comparison, we report just the ECE means in Figure 1. Please also refer to Tables 3 and 4 in the Appendix for numerical comparisons.

All figures show that both our Single Image and Multi-Image methods *consistently improve* (decrease) ECE across prior calibration methods (Ovadia et al. [2019]), as well as across levels of corruption intensity. Moreover, greater improvements using our calibration method are at higher corruptions i.e. greater dataset shifts.

Single Image vs Multi-Image: The Multi-Image method mostly performs better than Single Image with a few exceptions. This is a natural consequence of using more images to better estimate the dataset shift: with more information about the shifted test set, the correspondence with the calibration sets can be better estimated.

Improvement across prior calibrations: It can be seen (from Figure 1, Figure 2) that among the prior calibration methods, Ensemble performs the best both in the baseline and in our methods. However, note that the final ECE value of our methods across prior calibrations is relatively close. This means even without Ensemble i.e. using Vanilla, our calibration brings the results close to ensembling.

Results using Brier score: The Brier scores provide similar results, see Figure 13 and Figure 14 in the appendix.

Our results are based on the fact that we can use the model outputs, in our case simply the $p^{\max}$ values, as a proxy to evaluate dataset shift. This was illustrated in Figure 3: the calibration set whose $p^{\max}$ mean is closest to the $p^{\max}$ mean of the test set typically has lower calibration error. This is precisely what the Multi-Image method does, while the Single Image method closely approximates this using a weighted confidence estimate. Figure 4 shows the calibrated probabilities for a typical test set.

6.2 RESULTS ON OOD

Ideally, the models should not be confident when presented with completely OoD data. Figure 5 and Figure 6 plot the percentage of calibrated confidence values that are above a threshold vs the threshold, for the two experiments above respectively. It can be seen that both the Single Image and Multi-Image calibration results are significantly less confident compared to the benchmark [Ovadia et al., 2019].

We can explain the increased performance by looking at Figure 7. The confidence estimates $\hat{c}^{\text{Top-1}}$ are shifted towards lower values as the difference between the $p^{\max}$ means of the test and calibration set is smaller. Hence, by exposing the model to corrupted images at the calibration stage, it now "knows what it does not know".

7 CONCLUSIONS

Increasingly, we are asking models trained on a given dataset to perform on covariate shifted and OoD data. Our work focuses on uncertainty estimates, in particular, an estimate of the probability that our model classification is correct. In contrast to most deep uncertainty work, we use a purely statistical approach to reduce the calibration error of deep image classifiers under dataset shift. Previous work has shown that uncertainty estimates degrade on corrupted data. We overcome this limitation by introducing a method which allows a given model to be better calibrated to different dataset shifts.

We add a simple extra calibration step, and detect dataset shift using only the model outputs, and so calibrate for it. Our calibration method involves synthesizing surrogate calibration sets from increasing levels of a known corruption of a single type to the original calibration data, it works effectively against other corruptions as evidenced quantitatively. It is also shown to be effective in detecting OoD data. Our approach is model agnostic, so it can be applied to future models as well.

References

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1613–1622, Lille, France, 07–09 Jul 2015. PMLR. URL http://proceedings.mlr.press/v37/blundell15.html.

Sanghyuk Chun, Seong Joon Oh, Sangdoo Yun, Dongyoon Han, Junsuk Choe, and Youngjoon Yoo. An empirical evaluation on robustness and uncertainty of regularization methods. *arXiv preprint arXiv:2003.03879*, 2020.

Morris H DeGroot and Stephen E Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2): 12–22, 1983.

Pedro Domingos and Michael Pazzani. Beyond independence: Conditions for the optimality of the simple bayesian classi er. In *Proc. 13th Intl. Conf. Machine Learning*, pages 105–112, 1996.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL http://proceedings.mlr.press/v48/gal16.html.

Robert Geirhos, Carlos RM Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. In *Advances in neural information processing systems*, pages 7538–7550, 2018.

Alex Graves. Practical variational inference for neural networks. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2348–2356. Curran Associates, Inc., 2011.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 1321–1330, 2017.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

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

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. *Proceedings of the International Conference on Machine Learning*, 2019a.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *Proceedings of the International Conference on Learning Representations*, 2019b.

Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. AugMix: A simple data processing method to improve robustness and uncertainty. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

Ranganath Krishnan and Omesh Tickoo. Improving model calibration with accuracy versus uncertainty optimization. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18237–18248. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/d3d9446802a44259755d38e6d163e820-Paper.pdf.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6405–6416, 2017.

Christos Louizos and Max Welling. Structured and efficient variational deep learning with matrix gaussian posteriors. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1708–1716, New York, New York, USA, 20–22 Jun 2016. PMLR. URL http://proceedings.mlr.press/v48/louizos16.html.

Christos Louizos and Max Welling. Multiplicative normalizing flows for variational Bayesian neural networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2218–2227, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/louizos17a.html.

Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. In *ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning*, 2020.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Anh Mai Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 427–436, 2015. doi: 10.1109/CVPR.2015.7298640. URL https://doi.org/10.1109/CVPR.2015.7298640.

Jeremy Nixon, Michael W. Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

Adam Oberman, Chris Finlay, Alexander Iannantuono, and Tiago Salvador. Calibrated top-1 uncertainty estimates for classification by score based models. In *ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning*, 2020.

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 H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 13991–14002. Curran Associates, Inc., 2019.

Sangdon Park, Osbert Bastani, James Weimer, and Insup Lee. Calibrated prediction with covariate shift via unsupervised domain adaptation. In Silvia Chiappa and Roberto Calandra, editors, *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 3219–3229. PMLR, 2020. URL http://proceedings.mlr.press/v108/park20b.html.

Carlos Riquelme, George Tucker, and Jasper Roland Snoek. Deep bayesian bandits showdown. In *Proceedings of the International Conference on Learning Representations*, 2018.

Zhihui Shao, Jianyi Yang, and Shaolei Ren. Calibrating deep neural network classifiers on out-of-distribution datasets, 2020.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 06 2014.

Igor Vasiljevic, Ayan Chakrabarti, and Gregory Shakhnarovich. Examining the impact of blur on recognition by convolutional networks. *arXiv preprint arXiv:1611.05760*, 2016.

Ximei Wang, Mingsheng Long, Jianmin Wang, and Michael I. Jordan. Transferable calibration with lower bias and variance in domain adaptation, 2020.

Yeming Wen, Paul Vicol, Jimmy Ba, Dustin Tran, and Roger Grosse. Flipout: Efficient pseudo-independent weight perturbations on mini-batches. In *International Conference on Learning Representations*, 2018.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

A BRIER METRICS RESULTS

Figure 13 and Figure 14 report the means of the Brier scores and their spreads, respectively, for each model and dataset across different corruption types, for fixed corruption intensity going from 0 to 5.

Table 1 and Table 2 report the mean Brier scores for each model and dataset across different corruption types, for fixed corruption intensity going from 0 to 5. The Brier scores can be computed directly from the data, without binning. The ranking provided by the Brier scores is quite similar that provided by the ECE. On ImageNet the only difference is Ensemble at corruption level 1. On CIFAR-10 there were two ranking differences.

B ABLATION AND CROSS VALIDATION STUDY

We start by investigating the impact of the choice of corruption for the calibration set. Ideally, the choice of corruption should be representative of the distribution of corruptions, so a mild corruption or a very strong corruption would give slightly worse results. At the same time, here we demonstrate that choosing a different corruption should not significantly degrade the results.

In Figure 11 we perform a cross-validation study over the choice of corruption used to generate the calibration sets (always leaving it out of the corruptions used at test time). We plot the mean and variance of the ECE across different validation corruptions types.

We find that for CIFAR-10, both the Single Image and Multi-Image method are robust to the choice of validation corruption. On ImageNet, while the improvements are consistent with a few exceptions (ensemble and dropouts methods together with Multi-Image method), both methods are less robust.

The corruption chosen for the calibration sets should be such that we are able to capture the distribution shift. For instance, the choice of the brightness corruption produces almost the same $p^{\max}$ distribution as the clean images and therefore the improvement will be negligible (see Figure 15). On the other spectrum, we have glass blur for which the accuracy at level 1 was roughly half that of clean images and the resulting $p^{\max}$ distribution on the surrogate calibration sets are similar amongst themselves (see Figure 16). This tells us that what should guide the choice of the calibration sets, should be their shift on $p^{\max}$ distributions and respective accuracies, and not the human-chosen intensities.

We hypothesize that better results could be obtained for the different choices of corruptions by simply having the corruption strength be proportional to the loss of accuracy, as is the case of the contrast corruption (see Figure 3).

Figure 17 confirms this. It shows us that without any calibration the ECE scores become higher when the mismatch between the $p^{\max}$ distribution of the training set $\mathcal{D}_{\text{in}}^{\text{train}}$ and the $p^{\max}$ distribution of the test set $\mathcal{D}_{\text{in}}^{\text{test}}$ increases. Here we measure the mismatch in terms of the $p^{\max}$ means, the same criteria used in the Multi-Image method. These qualitative results are confirmed by the Pearson's correlation coefficient. In addition, this correlation corroborates why detecting the $p^{\max}$ distribution shit allows us to significantly improve the calibration of the different methods: in practice our proposed methods perform the recalibration of the model based on the calibration set whose $p^{\max}$ distribution is closest to the $p^{\max}$ distribution of the test set. Moreover, one notices the higher the correlation, the bigger the calibration improvement provided by both our Single Image and Multi-Image methods. For instance, Dropout has the lowest Pearson's r score and it is also the method where we notice the least improvement. On the other hand, Vanilla has the largest improvement and also the highest Pearson's r score.

C TABLES OF ECE METRICS ACROSS PRIOR CALIBRATIONS

Table 3 and Table 4 report the ECE scores for the model across different prior calibration methods, for ImageNet and CIFAR-10, respectively. Contrast is the corruption used for to form the calibration sets.

D TABLES OF ECE METRICS ACROSS DIFFERENT CORRUPTIONS

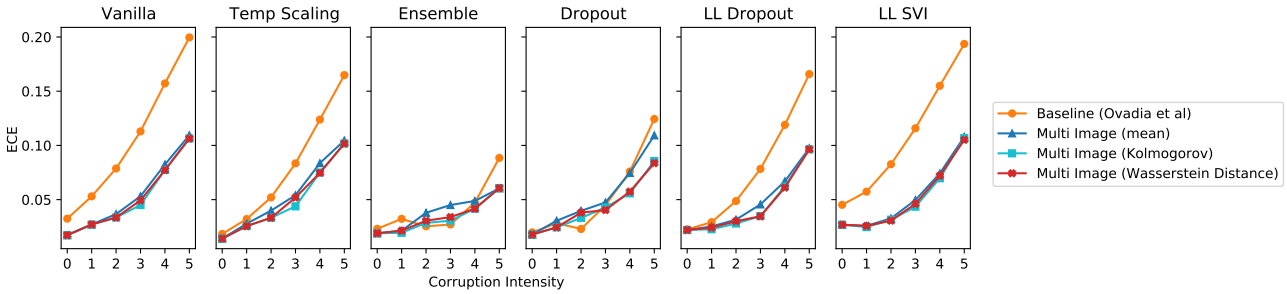
Table 5 and Table 6 report the ECE scores for the vanilla model across different corruption types and intensities ranging from 0 to 5 for ImageNet and CIFAR-10, respectively. Contrast is the corruption used for to form the calibration sets.

Table 1: Comparison on Imagenet of the benchmark implementation [Ovadia et al., 2019] versus our Single Image and Multi-Image methods. Numerical values of the means of the Brier scores across different corruptions types, for fixed corruption intensity going from 0 to 5.

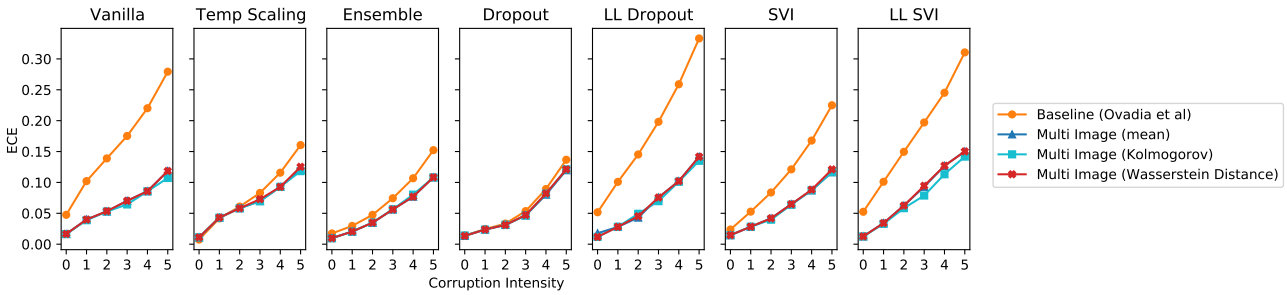
Method		Test	Corruption Intensity				
			1	2	3	4	5
Vanilla	Baseline (Ovadia et al.)	0.1222	0.1568	0.1696	0.1761	0.1794	0.1721
	Single Image (ours)	*0.1221*	**0.1541**	**0.1631**	*0.1645*	*0.1612*	*0.1475*
	Multi Image (ours)	**0.1218**	*0.1544*	*0.1634*	**0.1621**	**0.1577**	**0.1415**
Temp Scaling	Baseline (Ovadia et al.)	**0.1214**	0.1545	0.1650	0.1683	0.1674	0.1558
	Single Image (ours)	0.1219	**0.1541**	**0.1627**	*0.1633*	*0.1588*	*0.1438*
	Multi Image (ours)	*0.1215*	*0.1544*	*0.1635*	**0.1617**	**0.1569**	**0.1390**
Ensemble	Baseline (Ovadia et al.)	*0.1140*	0.1479	*0.1583*	**0.1585**	**0.1505**	*0.1307*
	Single Image (ours)	**0.1137**	*0.1475*	**0.1583**	*0.1589*	0.1515	0.1317
	Multi Image (ours)	0.1140	**0.1475**	0.1605	0.1612	*0.1512*	**0.1269**
Dropout	Baseline (Ovadia et al.)	**0.1290**	**0.1552**	**0.1613**	**0.1587**	**0.1551**	*0.1368*
	Single Image (ours)	*0.1291*	*0.1554*	*0.1617*	*0.1594*	0.1559	0.1374
	Multi Image (ours)	0.1293	0.1563	0.1639	0.1600	*0.1556*	**0.1360**
LL Dropout	Baseline (Ovadia et al.)	**0.1194**	**0.1512**	0.1613	0.1631	0.1605	0.1508
	Single Image (ours)	*0.1196*	*0.1512*	**0.1592**	*0.1578*	*0.1510*	*0.1365*
	Multi Image (ours)	0.1201	0.1514	*0.1601*	**0.1576**	**0.1485**	**0.1312**
LL SVI	Baseline (Ovadia et al.)	0.1291	0.1562	0.1647	0.1637	0.1613	0.1557
	Single Image (ours)	**0.1275**	**0.1528**	*0.1572*	*0.1512*	*0.1430*	*0.1318*
	Multi Image (ours)	*0.1282*	*0.1531*	**0.1572**	**0.1493**	**0.1388**	**0.1256**

Table 2: Comparison of CIFAR-10 of the benchmark implementation Ovadia et al. [2019] versus our Single-Image and Multi-Image methods. Numerical values of means Brier scores across different corruptions types, for fixed corruption intensity going from 0 to 5.

Method		Test	Corruption Intensity				
			1	2	3	4	5
Vanilla	Baseline (Ovadia et al.)	0.0671	0.1244	0.1623	0.1977	0.2355	0.2817
	Single Image (ours)	*0.0617*	*0.1063*	*0.1354*	*0.1609*	*0.1860*	*0.2141*
	Multi Image (ours)	**0.0616**	**0.1050**	**0.1334**	**0.1575**	**0.1777**	**0.2023**
Temp Scaling	Baseline (Ovadia et al.)	**0.0609**	*0.1056*	0.1346	0.1598	0.1849	0.2130
	Single Image (ours)	*0.0611*	0.1058	*0.1345*	*0.1594*	*0.1838*	*0.2106*
	Multi Image (ours)	0.0616	**0.1052**	**0.1335**	**0.1559**	**0.1781**	**0.2005**
Ensemble	Baseline (Ovadia et al.)	0.0434	*0.0800*	*0.1123*	*0.1420*	*0.1687*	*0.2023*
	Single Image (ours)	*0.0433*	0.0801	0.1128	0.1430	0.1693	0.2026
	Multi Image (ours)	**0.0430**	**0.0791**	**0.1105**	**0.1374**	**0.1597**	**0.1864**
Dropout	Baseline (Ovadia et al.)	**0.0634**	**0.0862**	0.1107	0.1363	0.1662	0.2010
	Single Image (ours)	0.0639	0.0864	*0.1107*	*0.1360*	*0.1657*	*0.1999*
	Multi Image (ours)	*0.0635*	*0.0863*	**0.1103**	**0.1352**	**0.1639**	**0.1960**
LL Dropout	Baseline (Ovadia et al.)	0.0746	0.1231	0.1674	0.2172	0.2706	0.3340
	Single Image (ours)	*0.0696*	*0.1047*	*0.1358*	*0.1698*	*0.2039*	*0.2428*
	Multi Image (ours)	**0.0678**	**0.1018**	**0.1330**	**0.1666**	**0.1919**	**0.2227**
SVI	Baseline (Ovadia et al.)	*0.0730*	0.1108	0.1428	0.1762	0.2138	0.2535
	Single Image (ours)	0.0737	*0.1076*	*0.1345*	*0.1617*	*0.1914*	*0.2200*
	Multi Image (ours)	**0.0726**	**0.1049**	**0.1320**	**0.1576**	**0.1826**	**0.2049**
LL SVI	Baseline (Ovadia et al.)	0.0708	0.1241	0.1714	0.2166	0.2605	0.3142
	Single Image (ours)	**0.0639**	*0.1058*	*0.1411*	*0.1746*	*0.2054*	*0.2398*
	Multi Image (ours)	*0.0641*	**0.1046**	**0.1398**	**0.1736**	**0.2023**	**0.2278**

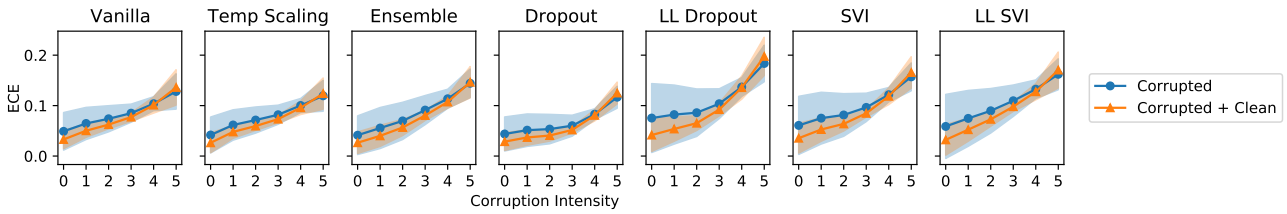


(a) ImageNet

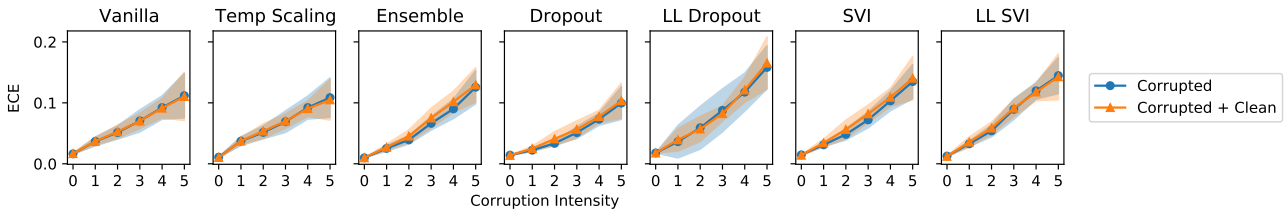


(b) CIFAR-10

Figure 8: Comparison different metrics to choose the surrogate calibration set: Mean Expected Calibration Error (ECE) (lower is better) of the benchmark implementation [Ovadia et al., 2019], versus our Multi-Image methods for ImageNet (top) and CIFAR-10 (bottom). Each box represents a different uncertainty method.

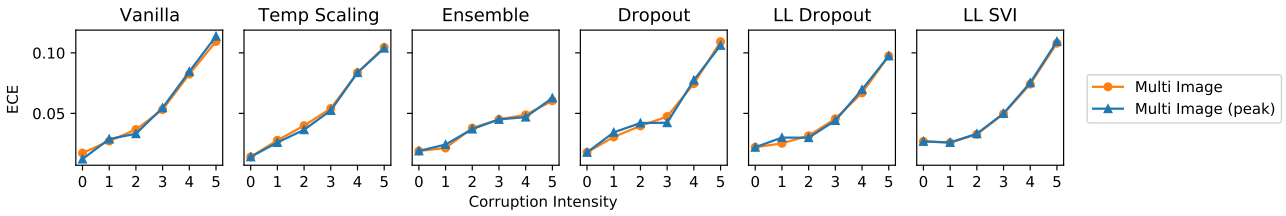


(a) Single Image

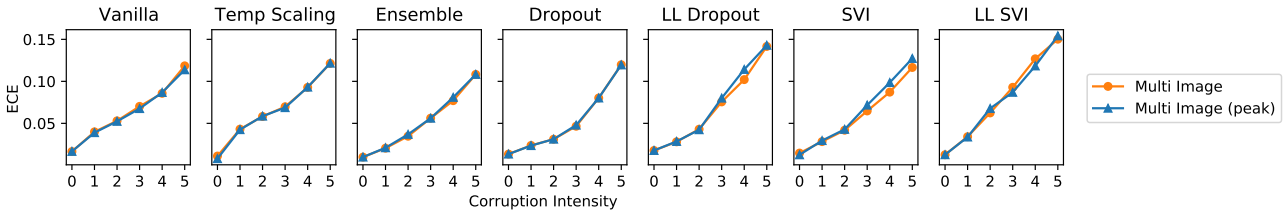


(b) Multi Image

Figure 9: Comparison of our Single Image (top) and Multi-Image (bottom) methods for CIFAR-10 with different choices of calibration sets: Corrupted+Clean refers to our choice of calibration sets as a union of clean and corrupted images, and Corrupted refers to calibration set without clean images for $j > 1$.

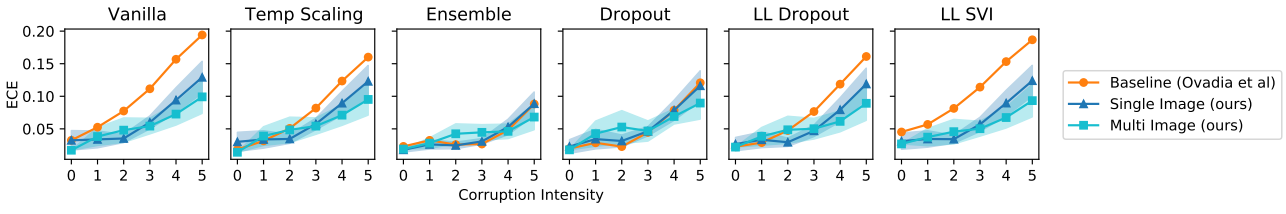


(a) ImageNet

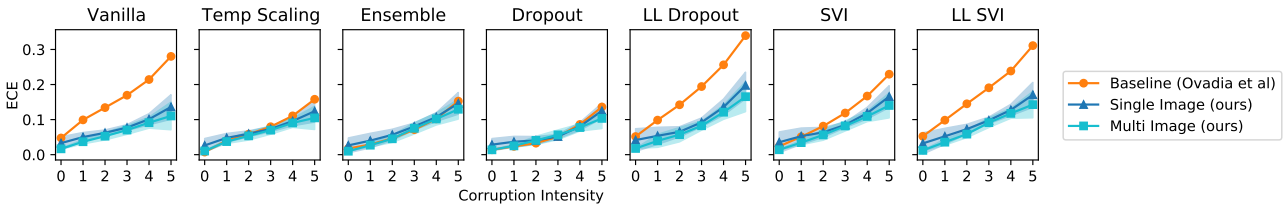


(b) CIFAR-10

Figure 10: Comparison of our Multi-Image method for ImageNet (top) and CIFAR-10 (bottom) using the full batch of test images and only 100 of them, which we refer to as peak. Mean Expected Calibration Error (ECE) across different corruptions types, for fixed corruption intensity going from 0 to 5. Each box represents a different uncertainty method.

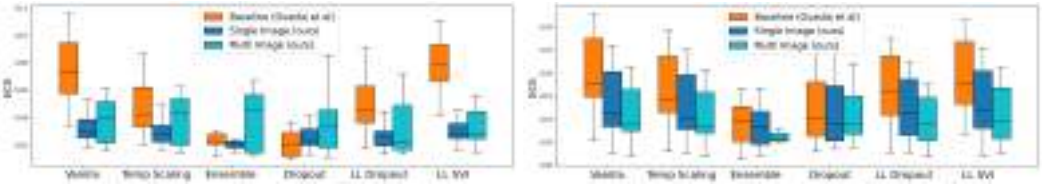


(a) ImageNet

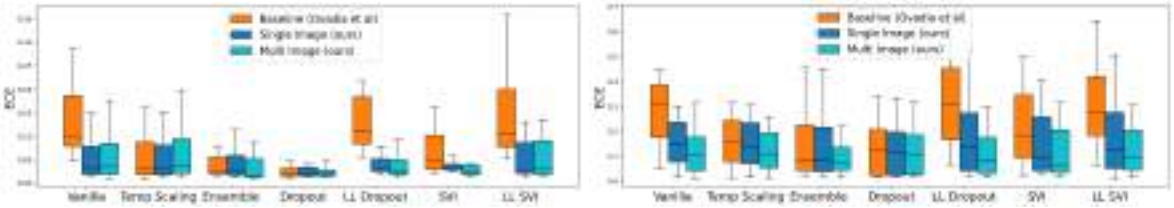


(b) CIFAR-10

Figure 11: Mean ECE (lower is better), averaged across different corruption types used in making the calibration sets. Figure 1 shows us the mean ECE using "contrast" as the calibration corruption. Here we show how those means change when different corruptions are used in the calibration set. For CIFAR-10, our proposed methods are robust to the choice of corruption used in the calibration set, while for ImageNet the choice of the corruption is more import, in particular for the Single Image method.

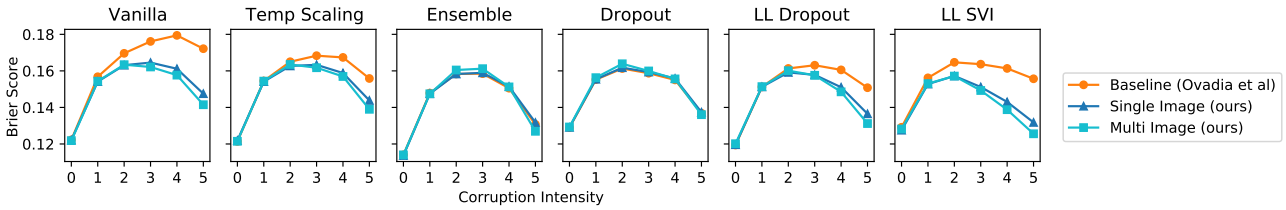


(a) ImageNet

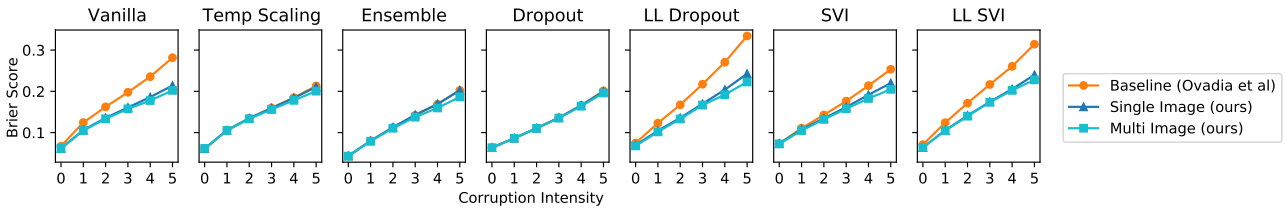


(b) CIFAR-10

Figure 12: Comparison of the ECE between the benchmark implementation [Ovadia et al., 2019] and our Single Image and Multi-Image methods for (a) ImageNet and (b) CIFAR-10, across different uncertainty methods, for fixed corruption intensity 2 (left) and 5 (right).

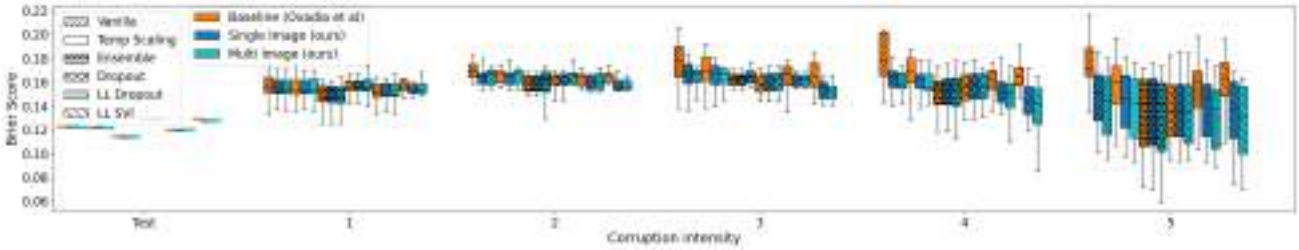


(a) ImageNet

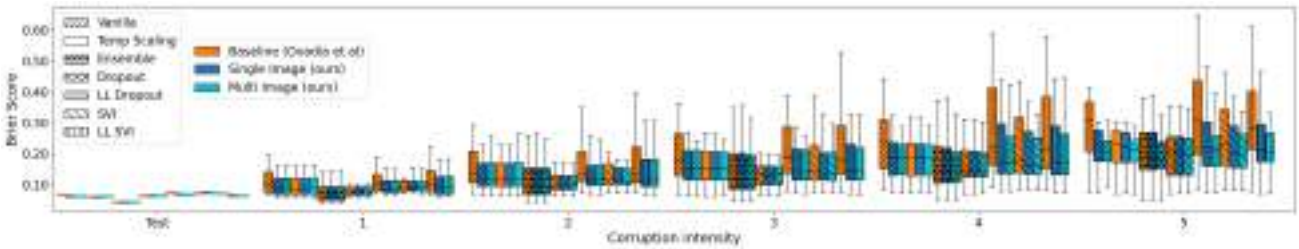


(b) CIFAR-10

Figure 13: Comparison of the benchmark implementation [Ovadia et al., 2019] versus our Single Image and Multi-Image methods for ImageNet (top) and CIFAR-10 (bottom). Mean Brier score across different corruptions types, for fixed corruption intensity going from 0 to 5. Each box represents a different uncertainty method. See Tables 1 and 2 for numerical comparisons. Notice that Brier score just for $c^{\text{Top-1}}$ Equation 5, rather than for c Equation 5.



(a) ImageNet



(b) CIFAR-10

Figure 14: Comparison of the benchmark implementation [Ovadia et al., 2019] versus our Single Image and Multi-Image methods for ImageNet (top) and CIFAR-10 (bottom). Brier score distribution across different corruptions types, for fixed corruption intensity going from 0 to 5. Each box represents a different uncertainty method. See Tables 1 and 2 in the Appendix for numerical comparisons.

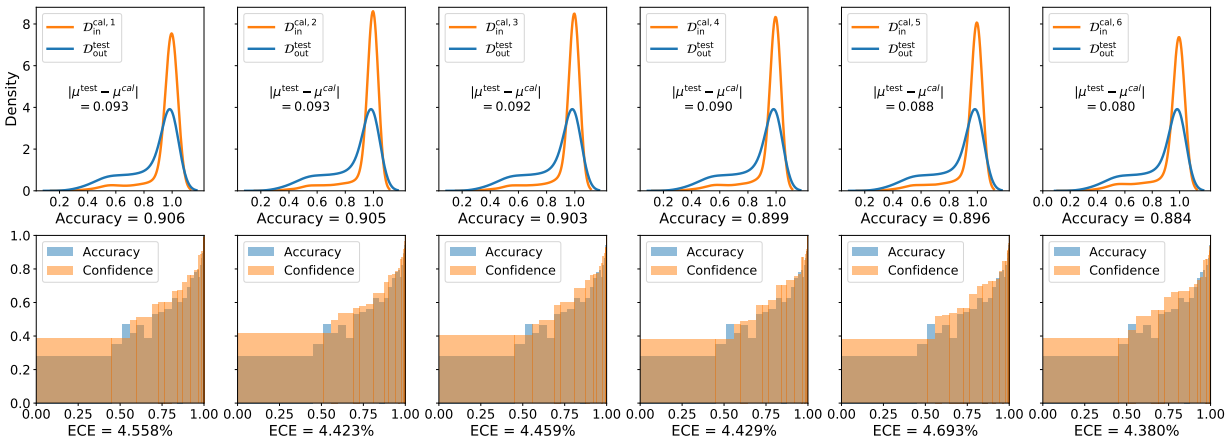


Figure 15: TOP: Probability density (using kernel density estimation) of $\mathcal{D}_{\text{out}}^{\text{test}}$ (blue) obtained by corrupting CIFAR10-test images with the "elastic transform" at intensity 4, and the $p^{\max}$ distribution of each calibration set $\mathcal{D}_{\text{in}}^{\text{cal},j}$ (orange) obtained by corrupting CIFAR10-cal images with varying intensity of a different corruption "brightness". BOTTOM: The respective accuracy and calibration confidence. The minimum calibration error is achieved precisely when the means of the distributions are the closest. The Single Image method chooses a linear combination of the calibrations based on the $p^{\max}$ of the test image. The Multi-Image method simply chooses the one with the closest corruption: $\mathcal{D}_{\text{in}}^{\text{cal},6}$ (6^{th} column).

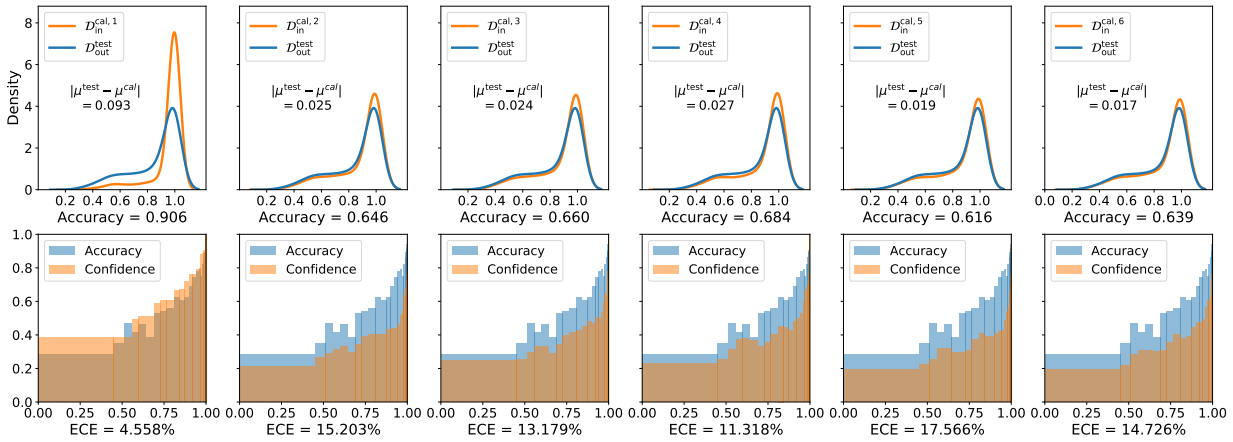
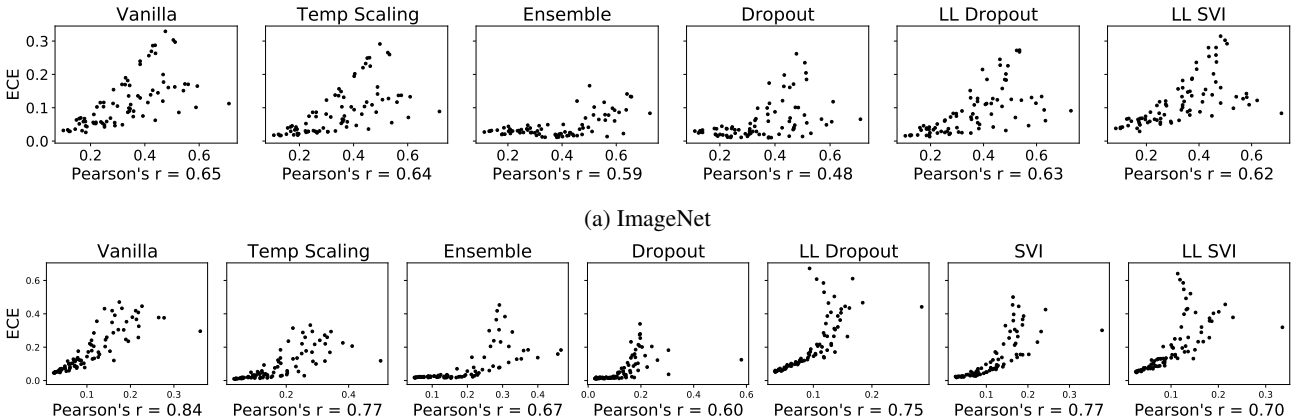


Figure 16: TOP: Probability density (using kernel density estimation) of $\mathcal{D}_{\text{out}}^{\text{test}}$ (blue) obtained by corrupting CIFAR10-test images with the "elastic transform" at intensity 4, and the $p^{\max}$ distribution of each calibration set $\mathcal{D}_{\text{in}}^{\text{cal},j}$ (orange) obtained by corrupting CIFAR10-cal images with varying intensity of a different corruption "glass blur". BOTTOM: The respective accuracy and calibration confidence. The minimum calibration error is achieved precisely when the means of the distributions are the closest. The Single Image method chooses a linear combination of the calibrations based on the $p^{\max}$ of the test image. The Multi-Image method simply chooses the one with the closest corruption: $\mathcal{D}_{\text{in}}^{\text{cal},6}$ (6^{th} column).



(a) ImageNet

(b) CIFAR-10

Figure 17: ECE (pre-calibration) versus $|\mu^{\text{test}} - \mu|$ and corresponding Pearson's r score, where μ^{test} and μ^{train} denote the $p^{\max}$ mean of the test set $\mathcal{D}_{\text{out}}^{\text{test}}$ and training set $\mathcal{D}_{\text{in}}^{\text{train}}$, respectively, for ImageNet (top) and CIFAR-10 (bottom). Each point in the plot represents a different corruption at a different level of intensity.

Table 3: Comparison on Imagenet of the benchmark implementation Ovadia et al. [2019] versus our Single Image and Multi-Image methods. Numerical values of the means ECE scores across different corruptions types, for fixed corruption intensity going from 0 to 5. The best results are indicated in bold and the second best in italic.

Method		Test	Corruption Intensity				
			1	2	3	4	5
Vanilla	Baseline (Ovadia et al.)	0.0324	0.0530	0.0787	0.1129	0.1572	0.1996
	Single Image (ours)	*0.0228*	*0.0277*	**0.0345**	*0.0633*	*0.0974*	*0.1354*
	Multi Image (ours)	**0.0172**	**0.0271**	*0.0367*	**0.0532**	**0.0824**	**0.1094**
Temp Scaling	Baseline (Ovadia et al.)	*0.0184*	0.0321	0.0521	0.0834	0.1238	0.1649
	Single Image (ours)	0.0210	**0.0271**	**0.0323**	*0.0598*	*0.0925*	*0.1292*
	Multi Image (ours)	**0.0141**	*0.0278*	*0.0399*	**0.0541**	**0.0836**	**0.1046**
Ensemble	Baseline (Ovadia et al.)	0.0231	0.0323	*0.0254*	**0.0271**	**0.0473**	0.0885
	Single Image (ours)	**0.0162**	*0.0246*	**0.0223**	*0.0285*	*0.0487*	*0.0834*
	Multi Image (ours)	*0.0189*	**0.0213**	0.0379	0.0450	0.0488	**0.0604**
Dropout	Baseline (Ovadia et al.)	0.0198	**0.0282**	**0.0229**	**0.0448**	0.0760	0.1244
	Single Image (ours)	*0.0180*	*0.0287*	*0.0268*	*0.0454*	*0.0752*	*0.1159*
	Multi Image (ours)	**0.0178**	0.0305	0.0396	0.0474	**0.0745**	**0.1092**
LL Dropout	Baseline (Ovadia et al.)	0.0225	0.0292	0.0487	0.0784	0.1189	0.1658
	Single Image (ours)	**0.0195**	*0.0258*	**0.0259**	*0.0484*	*0.0815*	*0.1231*
	Multi Image (ours)	*0.0220*	**0.0252**	*0.0315*	**0.0456**	**0.0669**	**0.0975**
LL SVI	Baseline (Ovadia et al.)	0.0452	0.0574	0.0826	0.1158	0.1550	0.1936
	Single Image (ours)	**0.0223**	**0.0258**	*0.0332*	*0.0621*	*0.0973*	*0.1360*
	Multi Image (ours)	*0.0269*	*0.0258*	**0.0327**	**0.0496**	**0.0742**	**0.1080**

Table 4: Comparison on CIFAR-10 of the benchmark implementation Ovadia et al. [2019] versus our Single Image and Multi-Image methods. Numerical values of the means ECE scores across different corruptions types, for fixed corruption intensity going from 0 to 5.

Method		Test	Corruption Intensity				
			1	2	3	4	5
Vanilla	Baseline (Ovadia et al.)	0.0476	0.1021	0.1389	0.1752	0.2201	0.2793
	Single Image (ours)	*0.0181*	*0.0418*	*0.0586*	*0.0807*	*0.1109*	*0.1540*
	Multi Image (ours)	**0.0164**	**0.0399**	**0.0530**	**0.0702**	**0.0857**	**0.1185**
Temp Scaling	Baseline (Ovadia et al.)	**0.0074**	**0.0417**	0.0609	0.0830	0.1158	0.1605
	Single Image (ours)	*0.0095*	*0.0419*	*0.0591*	*0.0794*	*0.1090*	*0.1492*
	Multi Image (ours)	0.0111	0.0431	**0.0582**	**0.0698**	**0.0928**	**0.1214**
Ensemble	Baseline (Ovadia et al.)	0.0174	0.0296	0.0475	0.0744	0.1068	0.1523
	Single Image (ours)	*0.0160*	*0.0286*	*0.0466*	*0.0736*	*0.1034*	*0.1462*
	Multi Image (ours)	**0.0098**	**0.0204**	**0.0347**	**0.0562**	**0.0770**	**0.1083**
Dropout	Baseline (Ovadia et al.)	*0.0140*	*0.0244*	0.0333	0.0535	0.0893	0.1367
	Single Image (ours)	0.0163	0.0252	*0.0333*	*0.0508*	*0.0854*	*0.1298*
	Multi Image (ours)	**0.0134**	**0.0237**	**0.0311**	**0.0465**	**0.0802**	**0.1198**
LL Dropout	Baseline (Ovadia et al.)	0.0517	0.1009	0.1453	0.1982	0.2589	0.3331
	Single Image (ours)	*0.0297*	*0.0413*	*0.0525*	*0.0836*	*0.1301*	*0.1877*
	Multi Image (ours)	**0.0177**	**0.0280**	**0.0430**	**0.0755**	**0.1021**	**0.1416**
SVI	Baseline (Ovadia et al.)	0.0237	0.0527	0.0837	0.1212	0.1678	0.2250
	Single Image (ours)	*0.0229*	*0.0430*	*0.0538*	*0.0762*	*0.1095*	*0.1536*
	Multi Image (ours)	**0.0146**	**0.0283**	**0.0418**	**0.0648**	**0.0872**	**0.1165**
LL SVI	Baseline (Ovadia et al.)	0.0525	0.1011	0.1496	0.1971	0.2452	0.3105
	Single Image (ours)	**0.0115**	*0.0385*	*0.0652*	*0.0962*	*0.1335*	*0.1834*
	Multi Image (ours)	*0.0126*	**0.0341**	**0.0624**	**0.0929**	**0.1268**	**0.1502**

Table 5: Comparison of ImageNet of the benchmark implementation Ovadia et al. [2019] versus our Single Image and Multi-Image methods for the vanilla classifier. Numerical values of ECE scores for different corruptions at different intensity levels going from 0 to 5. The contrast corruption was used to form the calibration sets as is therefore left out of the corruptions .

Corruption		Test	Corruption Intensity				
			1	2	3	4	5
Brightness	(Ovadia et al.)	0.0324	0.0319	0.0336	0.0358	0.0434	0.0531
	(Single Image)	*0.0228*	*0.0275*	*0.0281*	*0.0274*	*0.0251*	*0.0245*
	(Multi Image)	**0.0172**	**0.0174**	**0.0157**	**0.0197**	**0.0241**	**0.0199**
Defocus Blur	(Ovadia et al.)	0.0324	0.0425	*0.0489*	*0.0624*	0.0859	0.1012
	(Single Image)	*0.0228*	*0.0338*	**0.0312**	**0.0376**	**0.0422**	**0.0517**
	(Multi Image)	**0.0172**	**0.0327**	0.0580	0.0658	*0.0618*	*0.0552*
Elastic Transform	(Ovadia et al.)	0.0324	**0.0261**	0.0863	0.0587	0.1095	0.2632
	(Single Image)	*0.0228*	*0.0407*	*0.0270*	**0.0181**	**0.0432**	*0.1912*
	(Multi Image)	**0.0172**	0.0421	**0.0215**	*0.0472*	*0.0558*	**0.1545**
Fog	(Ovadia et al.)	0.0324	0.0526	0.0688	0.0969	0.1295	0.1996
	(Single Image)	*0.0228*	*0.0228*	**0.0229**	*0.0421*	**0.0647**	*0.1309*
	(Multi Image)	**0.0172**	**0.0195**	*0.0406*	**0.0283**	*0.0688*	**0.1066**
Frost	(Ovadia et al.)	0.0324	0.0531	0.0989	0.1387	0.1524	0.1770
	(Single Image)	*0.0228*	*0.0223*	*0.0373*	*0.0749*	*0.0878*	*0.1105*
	(Multi Image)	**0.0172**	**0.0209**	**0.0226**	**0.0688**	**0.0752**	**0.0876**
Gaussian Blur	(Ovadia et al.)	0.0324	*0.0311*	*0.0462*	0.0752	0.1198	0.1702
	(Single Image)	*0.0228*	0.0348	**0.0314**	**0.0412**	**0.0618**	*0.1059*
	(Multi Image)	**0.0172**	**0.0263**	0.0606	*0.0660*	*0.0697*	**0.0780**
Gaussian Noise	(Ovadia et al.)	0.0324	0.0695	0.1024	0.1699	0.2562	0.2971
	(Single Image)	*0.0228*	**0.0217**	*0.0429*	*0.1015*	*0.1845*	*0.2266*
	(Multi Image)	**0.0172**	*0.0241*	**0.0393**	**0.0611**	**0.1461**	**0.1795**
Glass Blur	(Ovadia et al.)	0.0324	0.0482	0.0734	0.1625	0.1713	0.1649
	(Single Image)	*0.0228*	*0.0259*	**0.0294**	*0.0977*	*0.1085*	*0.1045*
	(Multi Image)	**0.0172**	**0.0219**	*0.0534*	**0.0796**	**0.0849**	**0.0779**
Impulse Noise	(Ovadia et al.)	0.0324	0.1154	0.1543	0.1899	0.2693	0.3033
	(Single Image)	*0.0228*	*0.0506*	*0.0856*	*0.1198*	*0.1969*	*0.2324*
	(Multi Image)	**0.0172**	**0.0482**	**0.0455**	**0.0784**	**0.1570**	**0.1849**
Pixelate	(Ovadia et al.)	0.0324	0.0634	0.0675	0.1031	0.1354	0.1356
	(Single Image)	*0.0228*	**0.0209**	**0.0203**	*0.0432*	*0.0676*	*0.0685*
	(Multi Image)	**0.0172**	*0.0222*	*0.0239*	**0.0379**	**0.0336**	**0.0681**
Saturate	(Ovadia et al.)	0.0324	0.0515	0.0585	*0.0292*	0.0560	0.1128
	(Single Image)	*0.0228*	*0.0195*	*0.0227*	0.0332	*0.0228*	*0.0482*
	(Multi Image)	**0.0172**	**0.0187**	**0.0208**	**0.0202**	**0.0216**	**0.0237**
Shot Noise	(Ovadia et al.)	0.0324	0.0673	0.1158	0.1817	0.2874	0.3291
	(Single Image)	*0.0228*	**0.0215**	*0.0529*	*0.1119*	*0.2153*	*0.2577*
	(Multi Image)	**0.0172**	*0.0228*	**0.0500**	**0.0711**	**0.1737**	**0.2119**
Spatter	(Ovadia et al.)	0.0324	*0.0293*	0.0563	0.1050	0.1670	0.2397
	(Single Image)	*0.0228*	0.0310	*0.0186*	*0.0412*	*0.0988*	*0.1684*
	(Multi Image)	**0.0172**	**0.0201**	**0.0163**	**0.0226**	**0.0590**	**0.1399**
Speckle Noise	(Ovadia et al.)	0.0324	0.0583	0.0799	0.1694	0.2303	0.2863
	(Single Image)	*0.0228*	**0.0173**	*0.0276*	*0.1004*	*0.1585*	*0.2138*
	(Multi Image)	**0.0172**	*0.0190*	**0.0216**	**0.0600**	**0.1261**	**0.1721**
Zoom Blur	(Ovadia et al.)	0.0324	0.0553	0.0898	0.1156	0.1447	0.1606
	(Single Image)	*0.0228*	**0.0248**	**0.0397**	**0.0585**	*0.0833*	*0.0968*
	(Multi Image)	**0.0172**	*0.0510*	*0.0602*	*0.0706*	**0.0789**	**0.0812**

Table 6: Comparison of CIFAR-10 of the benchmark implementation Ovadia et al. [2019] versus our Single Image and Multi-Image methods for the vanilla classifier. Numerical values of ECE scores for different corruptions at different intensity levels going from 0 to 5. The contrast corruption was used to form the calibration sets.

Corruption		Test	Corruption Intensity				
			1	2	3	4	5
Brightness	(Ovadia et al.)	0.0476	0.0461	0.0501	0.0538	0.0617	0.0753
	(Single Image)	*0.0181*	*0.0173*	*0.0145*	*0.0190*	**0.0139**	*0.0222*
	(Multi Image)	**0.0164**	**0.0126**	**0.0108**	**0.0152**	*0.0164*	**0.0221**
Defocus Blur	(Ovadia et al.)	0.0476	0.0469	0.0528	0.0866	0.1292	0.2580
	(Single Image)	*0.0181*	*0.0128*	*0.0196*	**0.0177**	**0.0320**	*0.0995*
	(Multi Image)	**0.0164**	**0.0104**	**0.0132**	*0.0190*	*0.0363*	**0.0428**
Elastic Transform	(Ovadia et al.)	0.0476	0.0732	0.0808	0.0984	0.1330	0.1700
	(Single Image)	*0.0181*	*0.0220*	**0.0163**	*0.0267*	*0.0316*	**0.0576**
	(Multi Image)	**0.0164**	**0.0152**	*0.0172*	**0.0227**	**0.0311**	*0.0637*
Fog	(Ovadia et al.)	0.0476	0.0464	0.0486	0.0562	0.0748	0.2037
	(Single Image)	*0.0181*	*0.0165*	*0.0214*	*0.0254*	*0.0248*	**0.0859**
	(Multi Image)	**0.0164**	**0.0105**	**0.0105**	**0.0167**	**0.0199**	*0.0922*
Frost	(Ovadia et al.)	0.0476	0.0861	0.1279	0.2213	0.2414	0.3554
	(Single Image)	*0.0181*	**0.0225**	**0.0448**	**0.1148**	**0.1259**	**0.2288**
	(Multi Image)	**0.0164**	*0.0332*	*0.0464*	*0.1236*	*0.1360*	*0.2392*
Gaussian Blur	(Ovadia et al.)	0.0476	0.0470	0.0904	0.1570	0.2465	0.3777
	(Single Image)	*0.0181*	*0.0145*	**0.0154**	**0.0412**	*0.0971*	*0.2043*
	(Multi Image)	**0.0164**	**0.0116**	*0.0216*	*0.0489*	**0.0374**	**0.1126**
Gaussian Noise	(Ovadia et al.)	0.0476	0.1716	0.2864	0.3798	0.4187	0.4458
	(Single Image)	*0.0181*	**0.0655**	*0.1493*	*0.2280*	*0.2614*	*0.2849*
	(Multi Image)	**0.0164**	*0.0754*	**0.0973**	**0.1508**	**0.1803**	**0.1997**
Glass Blur	(Ovadia et al.)	0.0476	0.4188	0.3913	0.3484	0.4704	0.4329
	(Single Image)	*0.0181*	*0.2754*	*0.2464*	*0.2067*	*0.3203*	*0.2825*
	(Multi Image)	**0.0164**	**0.2028**	**0.1757**	**0.1396**	**0.2423**	**0.2050**
Impulse Noise	(Ovadia et al.)	0.0476	0.1280	0.2143	0.2594	0.3254	0.3760
	(Single Image)	*0.0181*	**0.0479**	**0.1047**	**0.1337**	*0.1715*	*0.2139*
	(Multi Image)	**0.0164**	*0.0594*	*0.1165*	*0.1456*	**0.1100**	**0.1249**
Pixelate	(Ovadia et al.)	0.0476	0.0697	0.1009	0.1365	0.2852	0.4308
	(Single Image)	*0.0181*	**0.0202**	**0.0349**	**0.0560**	**0.1724**	**0.3000**
	(Multi Image)	**0.0164**	*0.0273*	*0.0417*	*0.0600*	*0.1843*	*0.3144*
Saturate	(Ovadia et al.)	0.0476	0.0576	0.0715	0.0496	0.0672	0.1042
	(Single Image)	*0.0181*	**0.0181**	**0.0193**	*0.0144*	*0.0208*	**0.0284**
	(Multi Image)	**0.0164**	*0.0204*	*0.0201*	**0.0104**	**0.0190**	*0.0390*
Shot Noise	(Ovadia et al.)	0.0476	0.1154	0.1793	0.3048	0.3526	0.4081
	(Single Image)	*0.0181*	**0.0326**	**0.0730**	*0.1649*	*0.2054*	*0.2504*
	(Multi Image)	**0.0164**	*0.0461*	*0.0814*	**0.1016**	**0.1312**	**0.1683**
Spatter	(Ovadia et al.)	0.0476	0.0713	0.0992	0.1237	0.1163	0.1762
	(Single Image)	*0.0181*	**0.0216**	**0.0301**	**0.0452**	**0.0479**	**0.0898**
	(Multi Image)	**0.0164**	*0.0260*	*0.0379*	*0.0455*	*0.0532*	*0.0938*
Speckle Noise	(Ovadia et al.)	0.0476	0.1219	0.2055	0.2445	0.3036	0.3668
	(Single Image)	*0.0181*	**0.0378**	**0.0934**	**0.1211**	*0.1597*	*0.2135*
	(Multi Image)	**0.0164**	*0.0530*	*0.1044*	*0.1340*	**0.0910**	**0.1338**
Zoom Blur	(Ovadia et al.)	0.0476	0.0817	0.1036	0.1403	0.1758	0.2354
	(Single Image)	*0.0181*	*0.0240*	**0.0182**	**0.0273**	*0.0520*	*0.0865*
	(Multi Image)	**0.0164**	**0.0198**	*0.0196*	*0.0373*	**0.0430**	**0.0315**
Translation	(Ovadia et al.)	0.0476	0.0523	0.1200	0.1437	0.1202	0.0527
	(Single Image)	*0.0181*	*0.0194*	*0.0362*	**0.0493**	**0.0385**	*0.0162*
	(Multi Image)	**0.0164**	**0.0147**	**0.0332**	*0.0529*	*0.0403*	**0.0124**