Improving Confidence Estimates for Unfamiliar Examples

Zhizhong Li, Derek Hoiem
Department of Computer Science,
University of Illinois Urbana Champaign
{zli115,dhoiem}@illinois.edu

Abstract

Intuitively, unfamiliarity should lead to lack of confidence. In reality, current algorithms often make highly confident yet wrong predictions when faced with relevant but unfamiliar examples. A classifier we trained to recognize gender is 12 times more likely to be wrong with a 99% confident prediction if presented with a subject from a different age group than those seen during training. In this paper, we compare and evaluate several methods to improve confidence estimates for unfamiliar and familiar samples. We propose a testing methodology of splitting unfamiliar and familiar samples by attribute (age, breed, subcategory) or sampling (similar datasets collected by different people at different times). We evaluate methods including confidence calibration, ensembles, distillation, and a Bayesian model and use several metrics to analyze label, likelihood, and calibration error. While all methods reduce over-confident errors, the ensemble of calibrated models performs best overall, and T-scaling performs best among the approaches with fastest inference.

1. Introduction

In research, the i.i.d. assumption, that train and test sets are sampled from the same distribution, is convenient and easily satisfied. In practice, the training and test samples often come from different distributions, as developers often have access to a less diverse set of images than future samples observed by the deployed system. For example, the face images gathered by a company's employees may not have the racial or age diversity of the world's population. Scholars that study the impact of AI on society consider differently distributed samples to be a major risk [49]: "This is one form of epistemic uncertainty that is quite relevant to safety because training on a dataset from a different distribution can cause much harm." Indeed, high profile failures, such as a person being labeled as a gorilla [54] or a car driving through a tractor trailer [52], are due at least in part to failure to provide good confidence estimates for unfamiliar data.

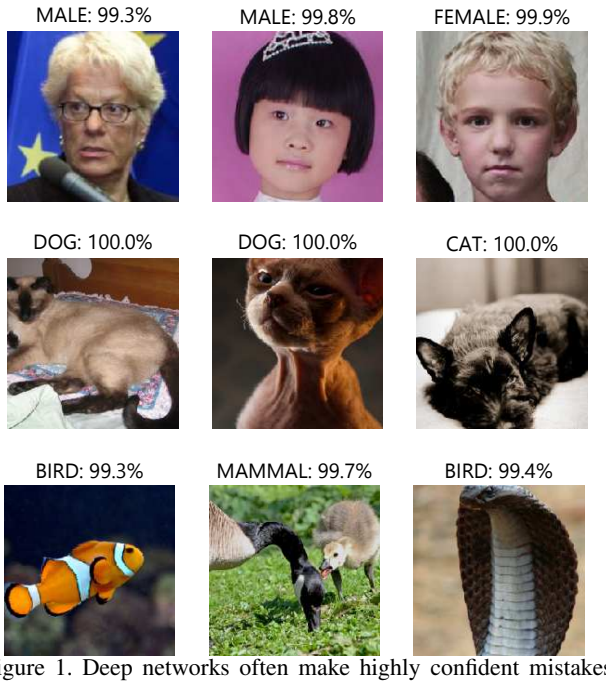


Figure 1. Deep networks often make highly confident mistakes when samples are drawn from outside the distribution observed during training. Examples shown have ages (top), breeds (middle), or species (bottom) that are not observed during training and are misclassified by a deep network model with high confidence. This paper investigates the problem of overconfidence for unfamiliar samples and evaluates several potential methods for improving reliability of prediction confidences.

In this paper, our goal is to compare and evaluate several methods for improving confidence estimates for familiar and unfamiliar samples. We consider *familiar samples* to be drawn from the same distribution as the training, as is typically done when creating training and test sets for research. We term *unfamiliar samples* as drawn from a different but still applicable distribution. For example, for cat vs. dog classification, an image of a dog from a breed seen during training is familiar, while an image from a breed not seen during training is unfamiliar. We are not concerned with non-applicable "out of domain" images such as an im-

age of a pizza for cat vs. dog classification.

We propose familiar/unfamiliar splits for four image classification datasets and evaluate by measuring accuracy of predicted labels and confidences. One would expect that classifiers would be less accurate and less confident for unfamiliar samples. Our experiments confirm that deep network classifiers have lower prediction accuracy on unfamiliar samples but also show that wildly confident wrong predictions occur much more often, due to higher calibration error. A simple explanation is that classifiers minimize a loss based on $P(y|x)$, for label y and input features x, which is unregulated and unstable wherever $P(x) \sim 0$ in training. Empirical support for this explanation comes from Novak et al. [34] who show that neural networks are more robust to perturbations of inputs near the manifold of the training data. We examine the effectiveness of calibration (we use temperature scaling [12]) for improving confidence estimates and the potential for further improvement using uncertainty-sensitive training [21], ensembles, and scaling based on novelty scores. Since calibrated ensembles perform best but are most computationally expensive, we also investigate distilling the ensemble from a mix of supervised and unsupervised data.

Our paper's **key contributions** are: (1) highlight the problem of overconfident errors in practical settings where test data may be sampled differently than training; (2) propose a methodology to evaluate performance on unfamiliar and familiar samples; (3) demonstrate the importance of confidence calibration and compare several approaches to improve confidence predictions, including new ideas for incorporating novelty prediction and mixed supervision distillation.

2. Related Work

Unreliability of prediction for unfamiliar samples: Lakshminarayanan et al. [23] show that networks are unreliable when tested on semantically unrelated or out-of-domain samples, such as applying object classification to images of digits. They also show that a using the Brier score [4] (squared error of 1 minus confidence in true label) as a loss and training an ensemble of classifier improves confidence calibration and reduces overconfident errors on out-of-domain samples. Ovadia et al. [35], in independent work concurrent to ours, also find that ensembles are most effective for skewed and out-of-domain samples, evaluating with Brier score, negative log likelihood of predictions, and expected calibration error (ECE). Our inclusion of Brier score and ECE is inspired by these methods. Our paper differs from these in the consideration of natural (not artificially distorted) samples from unfamiliar but semantically valid distributions, which is a common practical scenario when, for example, developers and users have access to different data. Roos et al. [42] distinguish between i.i.d. generalization error and off-training-set error and provide bounds based on repetition of input features. Extending their analysis to high dimensional continuous features is a worthwhile area of further study.

Methods to address epistemic uncertainty: When faced with unfamiliar samples, a model suffers from *epistemic uncertainty*, the uncertainty due to incomplete knowledge. Related works reduce this uncertainty by averaging over several models, with the intuition that different models will disagree and thus appropriately reduce certainty for parts of the feature distribution that are not well represented by the training set. Bayesian approaches [1, 2, 16] estimate a distribution of network parameters and produce a Bayesian estimate for likelihood. These methods are usually very computationally intensive [23], limiting their practical application. Gal and Ghahramani [9] propose MC-dropout as a discrete approximation of Bayesian networks, using dropout for a Monte-Carlo sample of likelihood estimates. Follow-up work by Kendall and Gal [21] proposes to estimate both aleatoric and epistemic uncertainties to increase the performance and quality of uncertainty. Lakshminarayanan et al. [23] propose a simpler method to average over predictions from multiple models with an ensemble of deep networks, an approach further validated by Ovadia et al. [35]. Multi-head deep networks [24, 31] emulate ensembles and are shown to outperform MC-dropout. Hafner et al. [13] propose a loss that encourages high uncertainty on training samples whose features are permuted or perturbed by noise. Our work differs primarily in its investigation of unfamiliar samples that are differently distributed from training but still have one of the target labels. Concurrent work by Mukhoti et al. [33] proposes combining focal loss with T-scaling. We evaluate T-scaling calibration [12], Kendall and Gal [21], ensembles, and a proposed novelty-sensitive T-scaling approach.

Calibration methods aim to improve confidence estimates, by learning a mapping from prediction scores to a well-calibrated probability. We use *T-scaling*, short for temperature scaling, in which the logit score of a classifier is divided by a scalar T as a special case of Platt calibration [38]. In a broad evaluation of calibration methods, Guo et al. [12] found T-scaling to be the simplest and most effective. Note that T-scaling has no effect on the rank-order of predictions, so it affects only the Brier error, negative log likelihood, and expected calibration error, not label error. Calibration parameters are fit to the validation set which is i.i.d. with training. Thus, calibration does not explicitly deal with unfamiliar samples, but our experiments show that calibration is an essential part of the solution for producing accurate confidence estimates on both familiar and unfamiliar samples.

Distillation [17] regresses the confidence predictions of a network to match those of another model, such as a larger network or ensemble. Radosavovic *et al*. [41] obtain soft labels on transformed unlabeled data and use them to dis-

till for unsupervised learning. Li and Hoiem [27] extend models to new tasks without retaining old task data, using the new-task examples as unsupervised examples for the old tasks with a distillation loss. Distillation has also been used to reduce sensitivity to adversarial examples that are similar to training examples [36]. We investigate whether distilling an ensemble into a single model can preserve the benefits of the ensemble on familiar and unfamiliar data, when using the training set and an additional unsupervised dataset to distill.

Other: The remainder of this section describes works and problem domains that are less directly related. *Domain adaptation* (e.g., [40]) aims to train on a source domain and improve performance on a slightly different target domain, either through unsupervised data or a small amount of supervised data in the target domain. *Domain generalization* [32, 26, 45] aims to build models that generalize well on a previously unspecified domain, whose distribution can be different from all training domains. These models generally build a domain-invariant feature space [32] or a domain-invariant model [45], or factor models into domain-invariant and domain-specific parts [26]. Attribute-based approaches, such as Farhadi *et al*. [7], attempt to learn features or attributes that are more likely to be consistent between familiar and unfamiliar samples. These methods require multiple training domains to learn invariant representations, with the intent to improve robustness to variations in the target domain. *One-shot learning* (e.g. [50]) and *zero-shot learning* (e.g. [53]) aim to build a classifier through one sample or only metadata of the class. Many methods more broadly attempt to *improve generalization*, such as data augmentation or jittering [39], dropout [46], batch normalization [20], and weight decay. Hoffer *et al*. [18] propose better hyperparameter selection strategies for better generalization. Bagging [3], ensembles, and other model averaging techniques are also used prior to deep learning.

Other methods aim to reduce confident errors by detecting failure [8, 10, 55, 51, 44], for example by looking at how close samples are to decision boundaries or estimating whether a test sample comes from the same distribution as training [5, 25, 28, 48, 22]. Typically, the motivation of these methods is to avoid making any prediction on suspect samples, while the goal of our work is to understand and improve performance of classifier predictions on both familiar and unfamiliar samples that have applicable labels.

3. Problem Setup and Methods

In many commercial settings, the developers of an algorithm have access to data that may be limited by geography, demographics, or challenges of sampling in diverse environments, while the intended users, in aggregate, have much broader access. For example, developers of a face attribute classification algorithm may undersample children, elderly, or Inuits, due to their own demographics. Someone training a plant recognition algorithm may have difficulty collecting samples of species not locally native. A recognizer of construction equipment may be applied to vehicle models that came out after release of the classification model. To study and improve the robustness of classifiers in these settings we explore:

- How to organize data to simulate the familiar and unfamiliar test sets (Sec. 3.1)
- How to evaluate the quality of predictions (Sec. 3.2)
- What methods are good candidates to improve prediction quality on unfamiliar samples (Sec. 3.3)

3.1. Datasets and Familiar/Unfamiliar Split

We choose four classification tasks for evaluation, detailed below and shown in Figure 2. For each of the first three tasks, a dataset is first split into "familiar" and "unfamiliar" subsets according to an attribute or subcategory, simulating the case of training data not containing the full diversity of potential inputs. In the fourth task (object presence classification), two similar datasets are used for the same object categories, simulating similar sources but sampled at different times. The "familiar" samples $(\mathbf{x}_\mathcal{F}, y_\mathcal{F}) \sim \mathcal{F}$ are further split into training F_{tr}, validation F_{vl}, and test F_{ts} sets, while the "unfamiliar" samples $(\mathbf{x}_\mathcal{U}, y_\mathcal{U}) \sim \mathcal{U}$ are used only for testing. The inputs $\mathbf{x}_\mathcal{F}$ and $\mathbf{x}_\mathcal{U}$ may occupy different portions of the feature space, with little to no overlap, but where they do overlap $P(y|\mathbf{x})$ is the same for $\mathcal{F}$ and $\mathcal{U}$. No sample from $\mathcal{U}$ is ever used in pre-training, training, or validation (parameter selection). In some cases, we use a dataset's standard validation set for testing (and not parameter tuning) so that we can compute additional metrics, as ground truth is not publicly available for some test sets.

Gender recognition: The extended Labeled Faces in the Wild (LFW+) dataset [14] with 15,699 faces is used. Samples are split into familiar $\mathcal{F}$ and unfamiliar $\mathcal{U}$ based on age annotations provided by Han et al. [15], with familiar ages 18-59 years and unfamiliar ages outside that range. The dataset comes with five preset folds; we use the first two for training, the third fold for validation, and the last two for testing.

Cat vs. dog recognition: Using the Pets dataset [37], the first 20 dog breeds and first 9 cat breeds are familiar, and the other 5 dog and 3 cat breeds are unfamiliar. The standard train/test splits are used (with training samples from $\mathcal{U}$ excluded).

Animal categorization: Four animal superclasses (mammals, birds, herptiles, and fishes) are derived from ImageNet [43], and different subclasses are used for familiar and unfamiliar sets. After sorting object classes within each superclass by their indices, the first half of classes are familiar $\mathcal{F}$, and the second half are unfamiliar $\mathcal{U}$. The data is also subsampled, so there are 800 training and 200 validation

Figure 2. Familiar and unfamiliar samples from each dataset. To study how classifier performance varies with novelty, we create splits of unfamiliar and familiar samples that are task-relevant, where the split is defined by age, breed, species, or date of sampling. The first three represent cases where the training distribution does not fully cover the test cases. The last represents a case of the minimal novelty achievable without independently sampling from the same image set.

examples drawn from the ImageNet training set per superclass, and 400 examples drawn from the ImageNet validation set for each of the unfamiliar and familiar test sets.

Object presence classification: The PASCAL VOC 2012 dataset [6] is used as familiar, with the similar 20 classes in MS COCO [29] used as unfamiliar. `tvmonitor` is mapped to `tv`. Test samples are drawn from the VOC PASCAL and MS COCO validation sets. The familiar and unfamiliar samples in this task are more similar to each other since they vary, not by attribute or subclass, but by when and by whom the images were collected.

3.2. Evaluation Metrics

We use several error metrics to assess the quality of classifier predictions. We denote $P_m(y_i|\mathbf{x}_i)$ as the assigned confidence in the correct label for the i^{th} of N samples by a model m. In all metrics, lower is better.

NLL: Negative log likelihood (NLL) $\frac{1}{N}\sum_i -\log P_m(y_i|\mathbf{x}_i)$ is a natural measure of prediction quality and commonly used as a loss for training classification models (often called "cross-entropy"), as it corresponds to the joint probability of predictions on independently drawn samples. The main drawback is that NLL is unbounded as confidence in the correct class approaches 0. To help remedy this, we clip the softmax probability estimates to $[0.001, 0.999]$ for all models before computing NLL.

Brier: The Brier score [4] measures the root mean squared difference between one and the confidence in the correct label: $\left(\frac{1}{N}\sum_i (1 - P_m(y_i|\mathbf{x}_i))^2\right)^{1/2}$. Similar to NLL, Brier is smallest when the correct label is predicted with high confidence, but the penalty for highly confident errors is bounded at 1, avoiding too much emphasis on a few large errors. We use RMS (root mean squared) instead of mean squared, as in the original, because we find it easier to interpret, and we call it "Brier error", since it should be

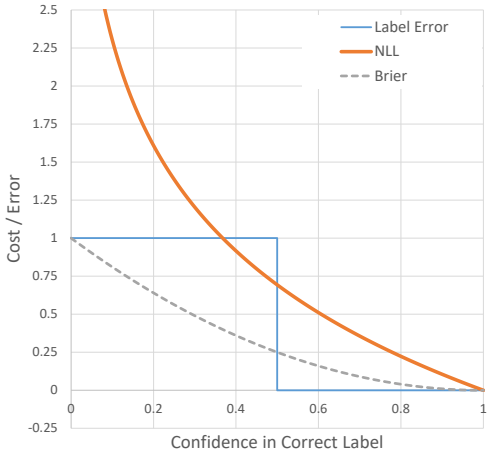


Figure 3. **Prediction quality metrics**: plot of error vs. confidence in correct label for 0-1 classification. NLL (negative log likelihood) strongly penalizes confidently wrong predictions, while Brier error penalties are constrained. Label error does not assess confidence beyond which label is most likely.

minimized.

Label Error: Label error is measured as the percent of incorrect most likely labels, or 1 minus average precision. We use percent incorrect for all tasks except object presence classification, for which we use mean average precision, in accordance with community norms for reporting performance on these tasks.

ECE: Expected calibration error (ECE) measures whether the classifier "knows what it knows". Following the notation of [12], ECE is computed as $\sum_{j=1}^{J} \frac{|B_j|}{N} |\mathrm{acc}(B_j) - \mathrm{conf}(B_j)|$ where B_j is a set of predictions binned by confidence quantile, $\mathrm{acc}(B_j)$ is the average accuracy of the B_j, and $\mathrm{conf}(B_j)$ is the average confidence in the most likely label. We use 10 quantiles for binning.

E99: E99 is the error rate among the subset of samples that have at least 99% confidence in any label. If the classifier is well-calibrated, E99 should be less than 1%. We created E99 to directly measure a model's tendency to generate highly confident errors.

3.3. Compared Methods

Deep network classifiers are often overconfident, even on familiar samples [12]. On unfamiliar samples, the predictions are less accurate and even more overconfident, as our experiments show. We consider several tools to improve predictions: calibration, novelty detection, ensembles, and loss functions that account for uncertainty. Some methods provide better confidence calibration, while others (e.g. ensembles and Bayesian models) can also provide more confidently accurate predictions.

T-scaling: Calibration aims to improve confidence estimates so that a classifier's expected accuracy matches its confidence. Among these, we use the temperature-scaling method described in Guo et al. [12]. At test time, all softmax logits are divided by temperature T. With $T > 1$, prediction confidence is decreased. T is a single parameter set to minimize NLL on the validation set. We then use this T on a network retrained on both training and validation sets.

Novelty-weighted scaling: We also consider novelty-weighted scaling, with the intuition that confidence should be lower for novel (i.e. unfamiliar) samples than for those well represented in training. We use the ODIN [28] model-free novelty detector. Since the novelty scores $novelty(\mathbf{x})$ often have a small range, we normalize them by linearly scaling the 5^{th} and 95^{th} percentile on training data to be 0 and 1 and clipping values outside $[0, 1]$. We then modify temperature scaling to set $T(\mathbf{x}) = T_0 + T_1 \cdot novelty(\mathbf{x})$, with T_0 and T_1 set by grid search on the validation set, so that temperature depends on novelty.

Ensemble methods consider both model parameter and data uncertainty by averaging over predictions. In areas of the feature space that are not well represented by training data, members of the ensemble may vary in their predictions, reducing confidence appropriately. In our experiments, members of the ensemble are trained with all training samples and differ due to varying initialization and stochastic optimization. We found this simple averaging approach to outperform bagging and bootstrapping. In prediction, the member confidences in a label y_i are averaged to yield the ensemble confidence: $P_m(y_i|\mathbf{x}_i) = \frac{1}{M} \sum_j^M P_{m_j}(y_i|\mathbf{x}_i)$, where M is the number of ensembles. $M = 10$ in our experiments.

Distillation: Our experiments show the ensemble is highly effective, but it is also M times more expensive for inference. We, thus, consider whether we can retain most of the benefit of the ensemble at lower compute cost using distillation [17]. After training the ensemble, the distilled model is trained by minimizing a weighted distillation loss (minimizing temperature-scaled cross-entropy of the ensemble's soft predictions with the distilled model's predictions) and a classification loss:

$$\mathcal{L} = \frac{1}{|F_{tr}|} \sum_{(\mathbf{x}_\mathcal{F}, y_\mathcal{F}) \in F_{tr}} \Big(\lambda_{cls} \mathcal{L}_{cls}(y_\mathcal{F}, f_{dis}(\mathbf{x}_\mathcal{F})) + \mathcal{L}_{dis}(f_{ens}(\mathbf{x}_\mathcal{F}), f_{dis}(\mathbf{x}_\mathcal{F})) \Big) \quad (1)$$

where $\mathcal{L}_{cls}$ is the classification loss over the distilled model's soft predictions $f_{dis}(\mathbf{x}_\mathcal{F})$, $\mathcal{L}_{dis}$ is the distillation loss over the soft predictions of the distilled model and ensemble $f_{dis}(\mathbf{x}_\mathcal{F})$, and λ_{cls} is a weighting to balance classification and distillation losses (λ_{cls} = 0.5, as recommended in [17]).

G-distillation: Under the standard distillation, the distilled model is guided to make similar predictions to the ensemble for the familiar distribution $\mathcal{F}$, but its predictions are still unconstrained for unfamiliar samples, potentially losing the benefit of the ensemble's averaging for samples from $\mathcal{U}$. Therefore, we propose G-distillation, a generalized distillation where the distillation loss is also computed over samples from an unsupervised distribution $\mathcal{G}$. In our experiments, we choose $\mathcal{G}$ to be related to the task, but make sure there is no overlap between specific examples in $\mathcal{G}$ and F_{ts} or $\mathcal{U}$. We use the following unsupervised datasets for $\mathcal{G}$ in our experiments: Gender, CelebA [30]; Broad animal, COCO [29]; Cat-dog, ILSVRC12 [43]; and Object presence, Places365-standard [56]. The images from $\mathcal{G}$ are disjoint with the datasets used to draw $\mathcal{F}$ and $\mathcal{U}$ for each respective task.

Bayesian model: Finally, we consider the Bayesian method of Kendall et al. [21], which accounts for uncertainty in model parameters (epistemic) and observations (aleatoric). To account for model parameter uncertainty, multiple predictions are made with Monte Carlo Dropout, and predictions are averaged. In this way, dropout is used to simulate an ensemble within a single network. In our implementation, we apply dropout to the second-to-last network layer with a rate of 0.2. Observation uncertainty is modeled

with a training loss that includes a prediction of logit variance. The logits can then be sampled based on both dropout and logit variance, and samples are averaged to produce the final confidence. See [21] for details.

4. Experiments

When comparing these methods, we aim to answer the following experimental questions:

- Do T-scaling calibration parameters learned from $\mathcal{F}$ also improve confidence estimates on $\mathcal{U}$?
- Does novelty-weighted scaling outperform the data agnostic T-scaling?
- Do ensembles learned on $\mathcal{F}$ also improve predictions on $\mathcal{U}$?
- Is distillation able to preserve ensemble performance on $\mathcal{F}$ and $\mathcal{U}$?
- Does adding the unsupervised set for distillation in G-distillation lead to better preservation?
- Does the Bayesian model that is specifically designed to manage model and observational uncertainty outperform more general alternatives?

(Spoiler alert: answers in order are yes, no, yes, no, yes, partially.)

4.1. Training and Testing Details

Training: For all experiments we use PyTorch with a ResNet-18 architecture and Adam gradient descent optimization with a momentum of 0.9. We initialize the final layer of our pre-trained network using Glorot initialization [11]. We perform hyper-parameter tuning for the learning rate and the number of epochs using a manual search on a validation split of the training data. When the performance on validation plateaus, we reduce the learning rate by a factor of 10 and run 1/3 as many additional epochs as completed up to that point. After fitting hyperparameters on the validation data, the models are retrained using both train and val sets. For data augmentation, we use a random crop and mirroring similar to Inception [47]. Places365-standard [56] dataset is used to pretrain the network, and the network is fine tuned separately for each task. When training G-distillation, we sample the image from G to be roughly $\frac{1}{4}$ the size of F_{tr}. We also verified that using a different architecture (DenseNet161 [19]) yields the same experimental conclusions.

Testing: At test time we evaluate on the center crop of the image. Due to the relatively high variance of NLL on U_{ts}, we run our experiments 10 times to ensure statistical significance (unpaired two-tail t-test with p=0.95 on model performance), but we run the ensemble method only once (variance estimated using ensemble member performance variance). Our 10 runs of the distillation methods use the same ensemble run.

	NLL		Brier		Label Error		ECE	
Gender	**fam.**	**unf.**	**fam.**	**unf.**	**fam.**	**unf.**	**fam.**	**unf.**
Baseline	0.083	0.542	0.147	0.352	0.028	**0.147**	0.013	0.109
T-scaling	12%	26%	2%	**4%**	0%	**0%**	**73%**	20%
Ensemble	**24%**	33%	**10%**	**6%**	**22%**	**0%**	36%	**29%**
Distill	8%	33%	3%	4%	3%	-7%	41%	21%
G-distill	13%	**38%**	5%	**6%**	9%	-5%	31%	**31%**
Bayesian	17%	26%	5%	4%	6%	**0%**	**77%**	19%
Cat vs. Dog								
Baseline	0.053	0.423	0.112	0.290	0.016	0.095	0.010	0.078
T-scaling	23%	30%	4%	5%	0%	0%	64%	23%
Ensemble	**40%**	**46%**	**17%**	**12%**	**22%**	**8%**	**79%**	**46%**
Distill	-13%	22%	-9%	1%	-18%	-4%	55%	26%
G-distill	-18%	27%	-14%	1%	-33%	-8%	41%	31%
Bayesian	17%	26%	3%	5%	0%	3%	42%	21%
Animals								
Baseline	0.326	1.128	0.199	0.341	0.104	0.291	0.048	0.187
T-scaling	13%	23%	3%	5%	0%	0%	**75%**	37%
Ensemble	**22%**	**32%**	**9%**	**8%**	**11%**	**6%**	50%	**57%**
Distill	7%	24%	1%	5%	-1%	0%	66%	45%
G-distill	14%	26%	5%	7%	7%	2%	56%	49%
Bayesian	16%	24%	5%	5%	4%	1%	**74%**	39%
Objects								
Baseline	0.086	0.128	0.154	0.186	0.195	0.455	0.005	0.010
T-scaling	0%	0%	0%	0%	0%	0%	2%	2%
Ensemble	**4%**	4%	**2%**	2%	**6%**	**3%**	**3%**	7%
Distill	-1%	**5%**	0%	**2%**	1%	0%	-31%	**10%**
G-distill	-2%	5%	-1%	2%	-2%	-1%	-41%	7%
Bayesian	0%	0%	0%	0%	0%	0%	3%	1%

Table 1. Performance of baseline (single model) for several metrics and percent reduction in error for other methods. All methods except baseline use T-scaling calibration. "T-scaling" is a single calibrated model.

4.2. Results

Our main table of results is shown in Table 1. The **baseline** is a single uncalibrated ResNet-18 network. The others correspond to the methods described in Sec. 3. For the baseline, we show the absolute error, and for the other methods, we show the percent reduction in error compared to the baseline (e.g. a drop from 0.10 to 0.09 is a 10% reduction) to facilitate comparison. The complete table with absolute error is included in the supplemental material. All methods except baseline use calibration.

Familiar vs. Unfamiliar Performance: Looking at baseline performance in Table 1, we see much higher error rates for unfamiliar samples, compared to familiar, for all tasks. The label error and calibration error are both higher, leading to much higher NLL and Brier error. *This means the baseline classifier is less accurate and has poor ability to detect its own inaccuracy on unfamiliar samples* — it does not know what it does not know. For example, in gender recognition, the label error increases from 2.8% for unfamiliar to 14.7%; the calibration error ECE increases from 0.013 to 0.109; and the NLL increases from 0.083 to 0.542. Figure 4 underscores the prevalence of confident er-

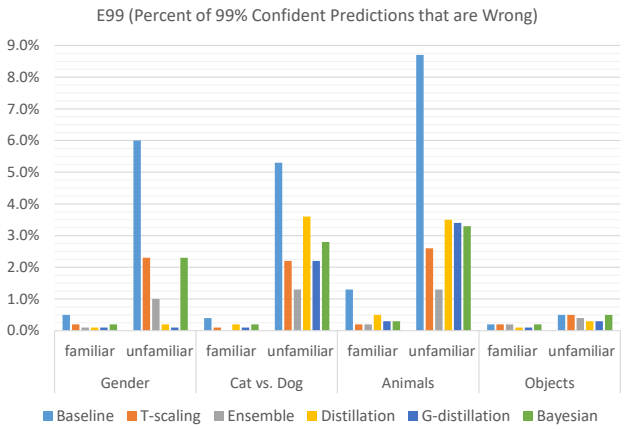


Figure 4. Classifiers are much more prone to confident errors when faced with unfamiliar samples. T-scaling calibration, among other methods, reduces the overconfidence, with ensembles of calibrated models providing consistent further improvement.

	NLL		Brier		ECE	
Gender	**fam.**	**unf.**	**fam.**	**unf.**	**fam.**	**unf.**
Single	0.083	0.542	0.148	0.352	0.013	0.109
Sin. T-scale	0.073	0.400	0.145	0.338	0.004	0.087
Ensemble	0.062	0.455	0.130	0.344	0.003	0.093
Ens. T-scale	0.063	0.363	0.130	0.333	0.009	0.077
Cat vs. Dog						
Single	0.053	0.423	0.110	0.290	0.010	0.078
Sin. T-scale	0.041	0.295	0.105	0.276	0.004	0.060
Ensemble	0.033	0.286	0.095	0.263	0.002	0.055
Ens. T-scale	0.032	0.229	0.095	0.255	0.002	0.042
Animals						
Single	0.326	1.128	0.200	0.341	0.048	0.187
Single T-scale	0.284	0.866	0.195	0.324	0.012	0.118
Ensemble	0.256	0.930	0.182	0.322	0.022	0.138
Ens. T-scale	0.254	0.772	0.182	0.311	0.024	0.080

Table 2. T-scaling calibration effectively reduces likelihood error (NLL, Brier) and calibration error (ECE) for many models across tasks for familiar and unfamiliar samples. Without calibration, using an ensemble reduces these errors, but an ensemble of calibrated models ("Ens. T-scale") performs best. Applying T-scaling to an ensemble of uncalibrated classifiers, and creating an ensemble of calibrated classifiers produces nearly identical results.

rors, which are several times more common for unfamiliar samples than familiar.

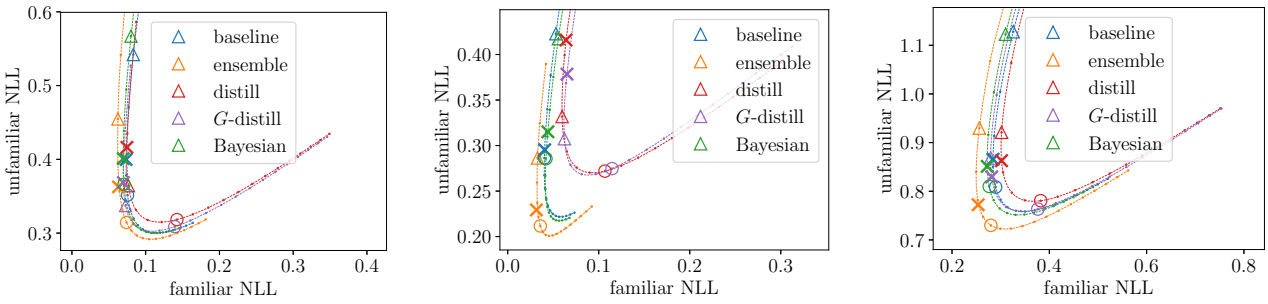
The differences between unfamiliar and familiar for object presence classification are substantial but smaller than other tasks, as expected, since VOC (familiar) and COCO (unfamiliar) images were both sampled from Flickr using similar methodologies [29]. The larger differences in mean AP (label error) may be due to lower frequency for a given object category in COCO.

Importance of calibration: Table 2 compares performance of the baseline and ensemble methods, both without and with T-scale calibration. Calibrated single models outperform uncalibrated models, and ensembles of calibrated models outperform ensembles of uncalibrated models. For example, in cat vs. dog recognition, the baseline NLL drops from 0.423 to 0.295, a 30% reduction; and the ensemble NLL drops from 0.286 to 0.229, a 20% reduction. Though not shown, a calibrated ensemble of uncalibrated models performs very similarly to an ensemble of calibrated models. For the object presence task, there is little effect of calibration because the classifier trained on the training samples was already well-calibrated for the familiar validation samples. We also found calibration to improve the Bayesian method [21]. Calibration has little effect for distillation and G-distillation, likely because distillation's fitting to soft labels makes it less confident. For those methods, we used calibration only when $T >= 1$, as setting $T < 1$ always made classifiers more over-confident. In Table 1, "T-scaling" refers to the T-scaled baseline, and T-scaling is used for all other non-baseline methods as well.

Given the benefits of T-scaling, we expected that novelty-weighted scaling, in which samples predicted to be unfamiliar have a greater temperature (reducing confidence more), would further improve results. However, we found the novelty weight T_1 was usually set to zero in validation, and, in any case, the novelty-weighted scaling performed similarly to T-scaling. The problem could be that the validation set does not have enough novelty to determine the correct weights. If we "cheat" and use samples drawn from the unfamiliar distribution to set the two weights T_0 and T_1, the method performs quite well. For example, when tuning parameters on a mix of familiar and unfamiliar samples for Gender recognition, novelty-weighted scaling performed best with 0.297 NLL compared to 0.328 for T-scaling and 0.313 for ensemble of calibrated classifiers that are tuned on the same data.

In Figure 5, we plot calibration curves of single networks, ensembles, distillation, G-distillation, and the Bayesian method with varying T. These curves allow us to peek at the best possible performance, if we were able to tune calibration parameters on unfamiliar and familiar test data. These curves allow a clearer view of which methods perform best. They also show that calibration on the familiar samples ('X' marks) leads to lower T values than is optimal for the unfamiliar samples. Generally, increasing T further would reduce likelihood error for unfamiliar samples without much adverse impact on likelihood error for familiar samples. On the object presence task (curve not shown), all models are well-calibrated (without T-scaling) for both unfamiliar and familiar categories.

Comparison of methods: Finally, considering Table 1, we see that the ensemble of T-scaled models dominates, consistently achieving the lowest label error, calibration error, NLL, and Brier error. The downside of the ensemble is higher training and inference computational cost, 10x in our



(a) Gender recognition (b) Cats vs. Dogs (c) Animal Categorization

Figure 5. Familiar and unfamiliar NLL error while varying the T calibration parameter. Triangles mark the uncalibrated models; 'X' marks models calibrated on the validation set. Circles mark $T = 2$, with each rightward dot increasing by 0.25. Without calibration, classifiers are often overconfident even for familiar samples, so calibration reduces confidence to improve NLL for familiar and unfamiliar. Ensembles dominate the other methods, always achieving lower NLL for some T.

case since we test with an ensemble of 10 classifiers. Distillation and G-distillation offered hope of preserving some of the gains of ensembles without the cost, and we expected the performance of G-distillation at least to fall between T-scaling and the ensemble. However, while G-distillation, which uses unsupervised samples to better mimic ensemble behavior in the broader feature domain, slightly outperforms distillation, neither method consistently outperforms T-scaling — no pain, no gain.

The method of Kendall et al. [21], which we call "Bayesian", performs second best to the ensemble, with small reductions in label error and comparable calibration improvements to all methods except ensemble. The Bayesian method also requires generating multiple predictions via MC-Dropout at test time, so also incurs significant additional computational cost.

The supplemental material compares prediction entropy to label cross-entropy (NLL), showing that calibration eliminates overconfidence for familiar samples but calibrated ensembles further reduce overconfidence on unfamiliar samples by increasing prediction uncertainty and improving accuracy.

4.3. Findings

We summarize our findings:

- Unfamiliar samples lead to much higher calibration error and label error, which can make their behavior unreliable in applications for which inputs are sampled differently in training and deployment.
- T-scaling is effective in reducing likelihood and calibration error on familiar and unfamiliar samples.
- The simple ensemble, when applied to T-scaled models, is the best method overall, reducing all types of error for both unfamiliar and familiar samples. The method of Kendall et al. [21] is the only other tested method to consistently reduce labeling error.
- T-scaling, distillation, and G-distillation all perform much better than the baseline.

Our recommendation: developers of any application that relies on prediction confidences (e.g. deciding whether to return a label, or to sound an alarm) should calibrate their models or, better yet, use calibrated ensembles. Ensembles achieve higher accuracy and better calibration, but at additional computational expense. We suspect that ensembles of shallower networks may outperform single deeper networks with similar computation costs, though we leave confirmation to future work. Tuning calibration on a validation set that is i.i.d with training leads to overestimates of confidence for unfamiliar samples, so to minimize likelihood error for both unfamiliar and familiar samples, it may be best to obtain a small differently-sampled validation set.

5. Conclusion

We show that modern deep network classifiers are prone to overconfident errors, especially for unfamiliar but valid samples. We show that ensembles of T-scaled models are best able to reduce all kinds of prediction error. Our work is complementary to recent works on calibration of i.i.d. data (e.g., Guo et al. [12]) and artificially distorted data [35]. More work is needed to improve prediction reliability with a single model in the unfamiliar setting and to consolidate learnings from the multiple recent studies of calibration and generalization. Data augmentation and representation learning are other important ways to improve generalization, and it would be interesting to evaluate their effect on prediction for both familiar and unfamiliar samples.

Acknowledgments: This work is supported in part by NSF Award IIS-1421521 and by Office of Naval Research grant ONR MURI N00014-16-1-2007.

References

[1] Christopher M Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995. 2

[2] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*, 2015. 2

[3] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, Aug 1996. 3

[4] Glenn W Brier. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950. 2, 4

[5] Pandu R Devarakota, Bruno Mirbach, and Björn Ottersten. Confidence estimation in classification decision: A method for detecting unseen patterns. In *Advances In Pattern Recognition*, pages 290–294. World Scientific, 2007. 3

[6] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015. 4

[7] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785. IEEE, 2009. 3

[8] Giorgio Fumera and Fabio Roli. Support vector machines with embedded reject option. In *Pattern recognition with support vector machines*, pages 68–82. Springer, 2002. 3

[9] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016. 2

[10] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in neural information processing systems*, pages 4885–4894, 2017. 3

[11] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010. 6

[12] 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, 2017. 2, 5, 8

[13] Danijar Hafner, Dustin Tran, Alex Irpan, Timothy Lillicrap, and James Davidson. Reliable uncertainty estimates in deep neural networks using noise contrastive priors. *arXiv preprint arXiv:1807.09289*, 2018. 2

[14] H. Han and A. K. Jain. Age, gender and race estimation from unconstrained face images. Technical Report MSU-CSE-14-5, Michigan State University, 2014. 3

[15] H. Han, A. K. Jain, S. Shan, and X. Chen. Heterogeneous face attribute estimation: A deep multi-task learning approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2017. 3

[16] José Miguel Hernández-Lobato, Yingzhen Li, Mark Rowland, Daniel Hernández-Lobato, Thang D Bui, and Richard E Turner. Black-box α-divergence minimization. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning-Volume 48*, pages 1511–1520. JMLR. org, 2016. 2

[17] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2, 5

[18] Elad Hoffer, Itay Hubara, and Daniel Soudry. Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In *Advances in Neural Information Processing Systems*, pages 1729–1739, 2017. 3

[19] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, volume 1 (2), page 3, 2017. 6

[20] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456, 2015. 3

[21] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems*, pages 5580–5590, 2017. 2, 5, 6, 7, 8

[22] Shehroz S Khan and Michael G Madden. A survey of recent trends in one class classification. In *Irish Conference on Artificial Intelligence and Cognitive Science*, pages 188–197. Springer, 2009. 3

[23] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, pages 6405–6416, 2017. 2

[24] Stefan Lee, Senthil Purushwalkam, Michael Cogswell, David Crandall, and Dhruv Batra. Why m heads are better than one: Training a diverse ensemble of deep networks. *arXiv preprint arXiv:1511.06314*, 2015. 2

[25] Yong Jae Lee and Kristen Grauman. Object-graphs for context-aware visual category discovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(2):346–358, 2012. 3

[26] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 5543–5551. IEEE, 2017. 3

[27] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 3

[28] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018. 3, 5

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 4, 5, 7

[30] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 12 2015. 5

[31] Osama Makansi, Eddy Ilg, Ozgun Cicek, and Thomas Brox. Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7144–7153, 2019. 2

[32] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18, 2013. 3

[33] Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip Torr, and Puneet Dokania. The intriguing effects of focal loss on the calibration of deep neural networks, 2020. 2

[34] Roman Novak, Yasaman Bahri, Daniel A Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Sensitivity and generalization in neural networks: an empirical study. *arXiv preprint arXiv:1802.08760*, 2018. 2

[35] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *NIPS*, 2019. 2, 8

[36] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *Security and Privacy (SP), 2016 IEEE Symposium on*, pages 582–597. IEEE, 2016. 3

[37] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar. Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 3

[38] 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. 2

[39] Dean Pomerleau. Neural network vision for robot driving. In *Intelligent Unmanned Ground Vehicles*, pages 53–72. Springer, 1997. 3

[40] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset shift in machine learning*. The MIT Press, 2009. 3

[41] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: Towards omni-supervised learning. *arXiv preprint arXiv:1712.04440*, 2017. 2

[42] Teemu Roos, Peter Grünwald, Petri Myllymäki, and Henry Tirri. Generalization to unseen cases. In *Advances in neural information processing systems*, pages 1129–1136, 2006. 2

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 3, 5

[44] Walter J Scheirer, Anderson Rocha, Ross J Micheals, and Terrance E Boult. Meta-recognition: The theory and practice of recognition score analysis. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1689–1695, 2011. 3

[45] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 3

[46] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014. 3

[47] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, Jen-Hao Rick Chang, et al. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 6

[48] DMJ Tax. *One-class classification*. PhD thesis, TU Delft, Delft University of Technology, 2001. 3

[49] Kush R Varshney and Homa Alemzadeh. On the safety of machine learning: Cyber-physical systems, decision sciences, and data products. *Big data*, 5(3):246–255, 2017. 1

[50] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638, 2016. 3

[51] Pei Wang and Nuno Vasconcelos. Towards realistic predictors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 36–51, 2018. 3

[52] Danny Yadron and Dan Tynan. Tesla driver dies in first fatal crash while using autopilot mode. *The Guardian*, 2016. news article. 1

[53] Oliver Zendel, Katrin Honauer, Markus Murschitz, Martin Humenberger, and Gustavo Fernandez Dominguez. Analyzing computer vision data - the good, the bad and the ugly. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3

[54] M. Zhang. Google photos tags two african-americans as gorillas through facial recognition software. *Forbes*, 2015. news article. 1

[55] Peng Zhang, Jiuling Wang, Ali Farhadi, Martial Hebert, and Devi Parikh. Predicting failures of vision systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3566–3573, 2014. 3

[56] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 5, 6